

DEVELOP 10 TIME FASTER

Tutorial

# XIII

*Express*

---

## Trial Version

---

This document cannot be released without PC SOFT written permission.

Remember to visit our Web site ([www.windex.com](http://www.windex.com)) on a regular basis to find out whether upgraded versions are available.

This documentation is not contractually binding. PC Soft reserves the right to modify or delete any topic dealt with in this document.

All product names or other trademarks mentioned in this publication are registered trademarks of their respective owners.  
© PC SOFT 2008: This publication may not be reproduced in part or in whole in any form without the express permission of PC SOFT.

# LICENSE AGREEMENT

## Software license agreement

---

*Caution: By installing and using WinDev Mobile Express Version, you have agreed with the terms of license agreement :*

SOFTWARE shall mean the right to use WinDev Mobile Express Version (also called Trial Version, Evaluation Version or Limited Version).

TRIAL USER shall mean the original customer (either individual or company) who installs and/or uses the software.

### 1. Express Version

The SOFTWARE is proposed in an "evaluation" version.

This trial version is intended for evaluation purpose only. It can also be used by the students of any educational establishment recognized by the relevant authorities during the duration of their studies. Any commercial use, in part or in full, is strictly forbidden.

Several features differ from the ones available in the "Commercial" version.

This version cannot be used to create commercial applications. To do so, a "Commercial" version must be purchased.

Creating applications, in part or in full, is strictly prohibited except for evaluation purpose or for practical work in the course of studies.

Applications can be created for evaluation purpose during the authorized period of time.

Unauthorized use of this software and/or use beyond the authorized period of time will be prosecuted to the maximum extend possible under the law.

The software (even in this evaluation version) shall not be duplicated or released without PC SOFT written agreement.

### 2. License

This SOFTWARE shall be used on a single computer by a single person at a time. The SOFTWARE is considered used as soon as it is loaded in a "computer's memory" (also called RAM).

This SOFTWARE can be used during the authorized period of time only.

### 3. SOFTWARE ownership

This software is the sole and exclusive property of PC SOFT. The TRIAL USER has only purchased the right to use the SOFTWARE according to these conditions.

### 4. Documentation

Reproducing the documentation, in part or in full, is strictly prohibited.

## 5. Use

The TRIAL USER agrees to use the SOFTWARE only for the purpose for which it is intended. The CUSTOMER shall not modify, or try to modify, nor disassemble, or try to disassemble this SOFTWARE. The CUSTOMER shall not remove (or try to remove) the copyright mentions that may be displayed and/or included in the SOFTWARE.

The SOFTWARE shall not be rented, leased or shared.

WinDev Mobile "Express" shall not be used to create and distribute all or part of a development tool, software for application creation and program building, or any other software of a type similar to WinDev Mobile. Contact us if you have any questions.

## 6. No responsibility

SOFTWARE and documentation are sold WITHOUT ANY WARRANTY as to performance, merchantability or fitness for any particular purpose. PC SOFT shall not be liable for damages of any nature whatsoever, including data loss or deterioration, financial or operational loss, discrepancies between information supplied in the documentation and SOFTWARE behavior, in case the SOFTWARE behaves differently than expected. The entire risk as to results and performance of programs created with the SOFTWARE is assumed by the TRIAL USER.

### Acceptance of the license :

By installing and using this software, you have agreed with the terms of license agreement.

This license shall not be modified by the representations of anyone unless a written amendment has been signed by a corporate officer of PC SOFT.

This license agreement is governed by French law; any disputes which may arise are to be settled within the jurisdiction of the courts where PC SOFT has its head office.

PC SOFT,  
3, rue de Puech Villa  
BP 44 408  
34197 Montpellier Cedex 5  
France

# SUMMARY

## License Agreement

Software license agreement .....	3
----------------------------------	---

## Introduction

Preliminary talks .....	13
Overview of the tutorial .....	13
How do I access the online help? .....	14
The guide .....	14
Legend for the symbols .....	14
If you are familiar with WinDev Mobile 11 ... ..	15
What is WinDev Mobile used for? .....	15
The new features of WinDev Mobile 12 .....	16
Work environment .....	16
Programming.....	17
WinDev 12 "Express" version .....	17

## PART 1 - DISCOVERING WINDEV MOBILE

### Before we start...

WinDev Mobile: what is it used for? .....	22
Overview of WinDev Mobile .....	23
The environment of WinDev Mobile .....	24
The window editor.....	26
The code editor.....	27
The data model editor .....	29
The query editor .....	30
The report editor .....	31
The setup editor.....	32
To conclude.....	32

### New features in WinDev Mobile 12

Overview .....	34
New features found in the example .....	34
Assisted input in the edit controls.....	34
Web Camera control.....	34
Using fDirSize and LengthToString.....	35
Sorts in memory list boxes and tables .....	35
New syntax for SWITCH .....	35
Higher speed for FOR loops .....	35
Procedure with variable number of parameters .....	36
New features in the editors .....	37
Go on global procedures .....	37
Automated tests with iterations .....	38
Running an automated test on a window.....	39



Running the automated tests of the executable found on the Pocket.....	39
Trace window .....	39
Control template .....	40
<b>Conclusion .....</b>	<b>40</b>

## PART 2 - FIRST APPLICATIONS

---

### Lesson 2.1. Your 1st application

---

<b>Now down to work! .....</b>	<b>44</b>
Starting WinDev Mobile.....	45
How do I create a project? .....	46
Creating your first window.....	50
Creating the controls found in the window .....	53
<b>Running the test of a WinDev Mobile window .....</b>	<b>58</b>
Test in simulation mode.....	58
Running test and debugging on Pocket PC .....	60
Direct test on the Pocket PC.....	62
<b>Characteristics of a WinDev Mobile window .....</b>	<b>63</b>
Changing the type of the window .....	64
.....	64
Specific process when closing the window.....	64
<b>Creating the executable program .....</b>	<b>65</b>
Creating the executable program.....	65
Running the test of the executable on the Pocket PC connected to the development computer .....	67
<b>Distributing your application .....</b>	<b>68</b>

### Lesson 2.2. Application on Smartphone

---

<b>Overview .....</b>	<b>70</b>
<b>How do I handle an SMS? .....</b>	<b>70</b>
<b>Creating the application .....</b>	<b>71</b>
Creating the project.....	71
Creating the window for sending SMSs .....	72
Creating the controls found in the window .....	73
Creating the window for reading SMSs.....	76
Creating the controls found in the window .....	77
Creating the selection window.....	79
Creating the controls found in the window .....	80
<b>Running the test of this application .....</b>	<b>80</b>
<b>Improving the application: managing the SIM card .....</b>	<b>82</b>
Adding the necessary elements .....	82
Running the test of SIM card management.....	85
<b>Creating the executable program and distributing the application .....</b>	<b>85</b>

**PART 3 - DATABASE**

**Lesson 3.1. Introduction**

<b>Format of the databases .....</b>	<b>90</b>
Hyper File Mobile .....	90
CEDB.....	91
Oracle Lite, SQL Server CE and AS/400 .....	92

**Lesson 3.2. Hyper File Mobile files**

<b>Overview .....</b>	<b>94</b>
<b>Generating a full application .....</b>	<b>94</b>
<b>Managing the data files .....</b>	<b>96</b>
Generated data files.....	96
Copying the data files.....	96
Synchronizing the data files.....	96
WDMaP and Pocket Map .....	97

**Lesson 3.3. CEDB files**

<b>Overview .....</b>	<b>99</b>
<b>Handling a record .....</b>	<b>99</b>
Adding a record.....	99
Modifying a record.....	100
Deleting a record .....	101
<b>Browsing a file .....</b>	<b>101</b>
<b>Searching for a record .....</b>	<b>102</b>
<b>Viewing the records .....</b>	<b>102</b>
Viewing the record items (Form mode).....	102
Viewing all the records (Table mode) .....	103

**PART 4 - SPECIFIC FEATURES OF POCKET PC**

**Lesson 4.1. Specific Formats**

<b>Managing the character strings .....</b>	<b>108</b>
What is the UNICODE format?.....	108
What is the ANSI format?.....	108
Using character strings in UNICODE format in Pocket PC .....	108
AnsiToUnicode and UnicodeToAnsi .....	108
The "Buffer" type.....	108
Handling external files.....	109
Transmission between two computers that use different formats of character strings.....	110
<b>Managing the directories in Windows for Pocket PC .....</b>	<b>110</b>
Handling a file by programming.....	110
Current directory .....	111
File picker.....	111
Storage card.....	115
<b>Memory space and speed of Pocket PC .....</b>	<b>116</b>
<b>Platform where the project is run .....</b>	<b>116</b>



## Lesson 4.2. Interactions between applications

---

<b>Sharing data between two applications</b> .....	<b>118</b>
Handling the same data files.....	119
Copying the data files to the Pocket PC.....	120
<b>Accessing the Pocket PC</b> .....	<b>120</b>

## Lesson 4.3. Sharing WinDev elements

---

<b>Importing a standard WinDev window</b> .....	<b>122</b>
How do I import a WinDev window? .....	122
Operations performed during the import.....	123
<b>Sharing source code</b> .....	<b>123</b>
Input of multi-product code.....	124
InPocketMode.....	124

## Lesson 4.4. Input mode

---

<b>Entering information on a Pocket PC</b> .....	<b>126</b>
<b>Entering information on a Smartphone</b> .....	<b>127</b>

## PART 5 - COMMUNICATION

---

### Lesson 5.1. Introduction

---

<b>Communicate with WinDev Mobile</b> .....	<b>132</b>
<b>Transferring files by FTP</b> .....	<b>132</b>
<b>Remote access (RPC on Hyper File Mobile)</b> .....	<b>133</b>
<b>Managing sockets</b> .....	<b>133</b>
<b>Web services (SOAP, J2EE, .NET)</b> .....	<b>134</b>
<b>SMS</b> .....	<b>135</b>
<b>Summary</b> .....	<b>135</b>

### Lesson 5.2. Managing emails

---

<b>Overview</b> .....	<b>137</b>
<b>Managing emails via the POP3/SMTP protocol</b> .....	<b>137</b>
<b>Managing emails via "CEMAPI"</b> .....	<b>138</b>
User account .....	138

## PART 6 - HOW DO I PROCEED?

---

<b>Controls, windows</b> .....	<b>144</b>
How do I change the type of a window?.....	144
How do I modify the type of the "OK/Close" button displayed in the title bar?.....	144
How do I display the keyboard on the Pocket PC? .....	145
How do I display all the drop-down menus of a window? .....	145
How do I duplicate a control found in a window by programming? .....	145
How do I delete a control found in a window by programming? .....	145
How do I manage the planes of a window? .....	145
How do I make a button invisible?.....	146
How do I modify the color of a static? .....	146
How do I display the progress of a process? .....	147

How do I link a window to an option of my main menu? .....	147
How do I create a popup menu? .....	148
How do I pass parameters to a window? .....	148
How do I change a check box into a radio button? .....	149
How do I retrieve the parameters passed in command line to an executable? .....	149
How do I group the controls in order to modify their properties by programming? .....	149
How do I align the controls? .....	150
How do I give the same size to the buttons? .....	150
How do I add a background image to a window? .....	150
<b>Environment .....</b>	<b>151</b>
How do I display or hide the panes? .....	151
How do I view the element to which the current process belongs? .....	151
How do I print the source code? .....	151
How do I print the analysis documentation? .....	151
How do I print the full documentation of my project? .....	151
How do I create a skin template? .....	151
How do I find and/or replace a variable in the code? .....	152
How do I manage a task list in my project? .....	152
How do I find out the list of project elements? .....	152
How do I view and change the tab order of the controls in a window? .....	153
How do I enable or disable the automatic data preview (Live Data)? .....	153
How do I add a language to my project? .....	153
How do I modify the options of WinDev Mobile? .....	153
<b>Various .....</b>	<b>153</b>
How do I perform a "screen shot"? .....	153
What are the processors on which the WinDev Mobile applications operate? .....	154
How do I read and write into a .INI file? .....	154
What are the image formats supported by WinDev Mobile? .....	154
I want to compress data, can I do this with WinDev Mobile? .....	154
How do I read and write in the registry? .....	155
How do I uninstall an application created with WinDev Mobile? .....	155
How do I create an executable? .....	155
How do I install an application? .....	155
How do I associate an icon with my executable? .....	156
How do I detect the elements not used by my application? .....	156
<b>Managing files and disks .....</b>	<b>156</b>
How do I manage the files found on the Pocket PC from a standard WinDev application? .....	156
How do I list all the files found in a directory? .....	156
How do I copy files? .....	157
How do I create a directory? .....	157
How do I read a text file? .....	157
<b>Tables .....</b>	<b>158</b>
How do I modify the search key in a table linked to a file? .....	158
How do I modify the stored item of a table linked to a file? .....	158
<b>Hyper File Mobile .....</b>	<b>159</b>
Is the format of the Hyper File files compatible with the format of the Hyper File Mobile files? .....	159
How do I disable an integrity constraint? .....	159
How do I manage the NULL value? .....	159
How do I manage a duplicate error when writing into a file? .....	160
How do I manage an integrity error when writing into a file or when deleting from a file? .....	160



How do I fill a TreeView control from a file?.....	161
How do I manage a composite key during a search? .....	161
<b>CEDB .....</b>	<b>162</b>
How do I handle the information included in the applications found by default on my Pocket PC (contacts, notes, and so on)? .....	162
How do I create a CEDB database? .....	162
Why should I use a CEDB database? .....	162
<b>Queries .....</b>	<b>163</b>
How do I optimize the speed of a query?.....	163
How do I add or modify a condition in a query? .....	163
How do I add or modify a sort in a query? .....	163
<b>Printout .....</b>	<b>164</b>
How do I print from a WinDev Mobile application? .....	164
What is the PCL standard?.....	164
Why does the font on the printed page differ from the font in my report? .....	164
<b>Ports .....</b>	<b>164</b>
How do I read a bar code? .....	164
How do I read the data sent by a reader of magnetic card? .....	165
How do I manage a serial port?.....	165
How do I manage a parallel port?.....	165
How do I manage an infrared port?.....	165

## Conclusion

---

## Appendix

---

# INTRODUCTION

## Preliminary talks

---

**Caution:** This manual is a tutorial. We advise you to refer to the online help when you use WinDev Mobile.

The aim of the tutorial is to help you discover WinDev Mobile, become familiar with the editors and teach you the concepts of WinDev Mobile. This manual does not cover all the features of WinDev Mobile.

This manual is intended for the developers who are already familiar with standard WinDev and who know how to handle a Pocket PC. This manual presents the main concepts required to develop an application for Pocket PC.

If you are not familiar with standard WinDev, we recommend that you to read the tutorial of standard WinDev beforehand.

Note: To receive the tutorial of standard WinDev, get in touch with the sales department of PC SOFT. You should plan on spending a few hours to follow this course and to learn WinDev Mobile: you'll find it was worth it!

If you try to develop an application before practising, you will lose time, and much more than a couple of days.

This course was designed so you can approach it in two different ways:

- either you follow all the detailed exercises in each lesson (recommended method).
- or, if you are in a hurry and already experienced, you can read through it without doing the exercises, as all the exercises have screen shots. However, in order to quickly assimilate the main concepts, we recommend that you follow the course step by step.

WinDev Mobile evolves all the time, so the screen shots found in this course may differ from the screen shots found in your product.

The language aspect is only one of the many aspects of development. Programming is a lot easier if all the aspects of development are taken into account.

## Overview of the tutorial

---

The tutorial has been designed to progressively teach you how to use WinDev Mobile. By following this course:

- you will discover the main concepts explained here informally; these are the concepts you need to learn and understand.
- you will also be asked to perform tasks that demonstrate the concepts just explained.

A glossary that summarizes the terms used is available in the Appendices (page 168).


As you progress through the tutorial, if you want to take a closer look at a concept or if you want to get more details about a programming function, see the online help (directly accessible from the editors or from the guide).

The size of a lesson is not necessarily proportional to its relevance ...

And don't forget to take a look at the examples supplied with WinDev Mobile: they are very instructive!


## How do I access the online help?

---

1. In the code editor, a specific help is available for each function via the [F1] key.
2. The button  accessible from each window
3. In the editors, by pressing the [F1] key.
4. In the editors, the help menu (symbolized by "?") enables you to display the help summary or to search for specific information.

## The guide

---

The guide contains over 150 wizards and several examples. Accessible via the "saucer" (  ), it enables you to get familiar with the features of WinDev Mobile and even to discover some features you knew little or nothing about!

## Legend for the symbols

---



This symbol indicates the duration of the lesson. Please note that the actual time may change according to your level of experience



EXAMPLE

An example is available to complement the lesson. The examples supplied with WinDev Mobile can be accessed from the "Wizards, Examples and Components" pane.



This symbol introduces a "Tip", we advise you to read the associated text.



This symbol introduces a "Warning", reading the associated text is extremely important.



This symbol introduces a "Note", we advise you to read the associated text.



This symbol gives the result of a "Test", we advise you to read the associated text.

## If you are familiar with WinDev Mobile 11 ...

---

If you are familiar with WinDev Mobile 11 and if you want to find out more about the new features of WinDev Mobile 12, we advise you to read and to follow the chapter named 'New features of WinDev Mobile 12' (page 33).

But following the entire tutorial can do no harm: it is a good occasion for "revising" the features of WinDev Mobile!

## What is WinDev Mobile used for?

---

WinDev Mobile is an IDE (Integrated Development Environment). It enables you to develop applications in many fields:

- Management of stocks
- Inventories, traceability of goods
- Adjustment and monitoring of machines on an assembly line
- Taking orders for fast processing in a temporary outlet (fairs, schools, stand, and so on)
- Customer forms
- Help with making snap decisions on a cell phone
- Checking the identity of visitors at an event: trade fair, presentation of products, and so on.
- Doctors or vets away from their surgeries
- Taking information in a temporary outlet: trade fair, street poll, stadium, and so on.
- Restoring leased heavy equipment (tools, vehicles, and so on) to a parking lot
- ...

WinDev Mobile is a development environment that includes all the tools required for developing an application.

Unlike some other programming languages, you don't need to find and add modules to be able to design, test and install an application.

The 5GL (5th Generation Language) of WinDev Mobile, the WLanguage, will surprise you by its simplicity: a few hours are all you need to get the hang of it, a week is usually all it takes to fully master its potential!

No more programming hassle, the WLanguage is available in English and in French!

**Note:** In this book, "Pocket PC" represents all the possible runtime platforms (Pocket PC, Smartphone, Psion, and so on). For special cases, the name of the relevant platform is specifically mentioned.

## The new features of WinDev Mobile 12

---

The non-exhaustive list of the new features that you will discover in WinDev Mobile 12 is as follows:

### Work environment

The new features of the work environment:

- **Project Editor:**
  - New RAD (Rapid Application Development)
  - Internal components
  
- **Data Model Editor:**
  - Multilingual name for items and files
  - The analysis groups can be handled by programming
  
- **Window Editor:**
  - New catalog of images.
  - Refactoring a selection of controls into a control template.
  - Templates of control templates.
  - Templates of window templates.
  - Style inheritance.
  - New sort options in the list boxes and tables.
  - Memory column and 'check box' column in browsing tables.
  - Table: 'progress bar' column, cell merge.
  - Browsing table loaded in memory.
  
- **Code Editor:**
  - Go of global procedure.
  - 'Trace' pane of the debugger to display the result of the trace during the last test.
  - Automated test of a window.
  - Iterations in automated tests.
  - Test robot to run tests on the Pocket PC.
  
- **Hyper File Client/Server:**
  - Hot backup.
  - Programmed task on the server
  - Server triggers.
  
- **Source Code Manager:**
  - Window automatically checked out when it is modified.
  - Management of rights.
  - Management of physical change of computer.
  - Masking the analysis files.
  - Ability to process the tasks and the bugs during the checkout.

## Programming

The new programming features:

- Hyper File: unicode items.
- Emulating cameras and video recording
- New functions:
  - Functions for managing the video resolution
  - ArrayToString and StringToArray
  - SNMP functions
  - Functions for managing services
  - Functions for managing the recycle bin
  - ...

## WinDev 12 "Express" version

---

Caution: Some features presented or quoted in this tutorial may not operate in this trial version.



DEVELOP 10 TIMES FASTER

**PART 1**

**Discovering  
WinDev Mobile**

**XIII**  
**Express**



# BEFORE WE START...

**This lesson will explain the following concepts...**

---

- WinDev Mobile: what is it used for?
- Overview of the features of WinDev Mobile.

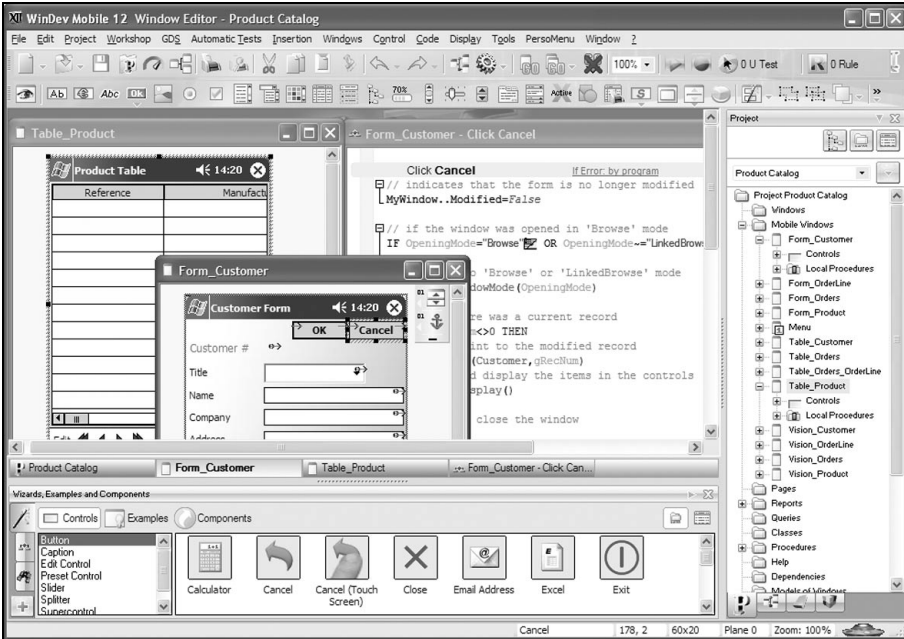


Estimated time: 1h

## WinDev Mobile: what is it used for?

WinDev Mobile 12 enables you to manage the development cycle of an application for Pocket PC.

WinDev Mobile enables you to develop all the applications you've been dreaming of. The environment of WinDev Mobile is as follows:



WinDev Mobile 12 enables you to create applications for managing data. The WinDev Mobile applications access most of the available databases. WinDev Mobile 12 includes Hyper File Mobile, a powerful relational database already used by hundreds of sites!

WinDev Mobile 12 proposes a powerful work environment. Your teams will easily create breathtaking applications.

The window editor of WinDev Mobile 12 is 100% WYSIWYG ("What You See Is What You Get"). It enables you to easily create great windows linked to the data.

## Overview of WinDev Mobile

This lesson teaches you to perform your first operations (don't worry, nothing too difficult!) in the environment of WinDev Mobile 12.

- ▶ Start WinDev Mobile 12:
  - click the desktop icon:

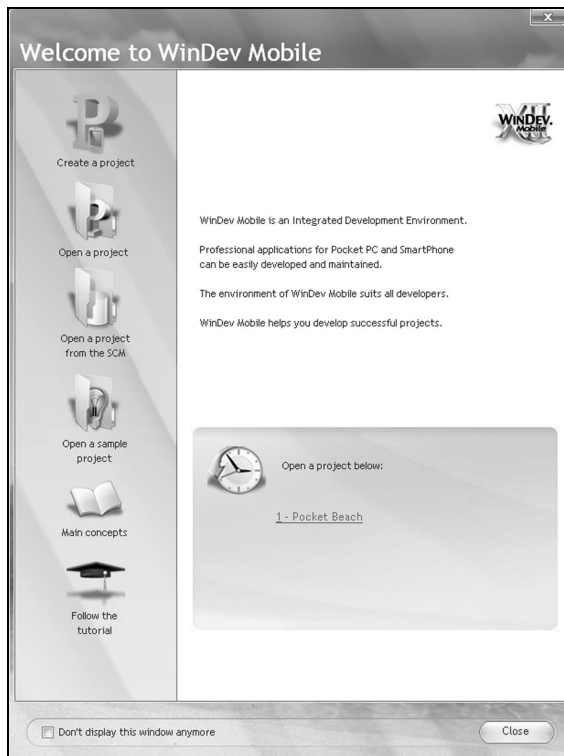


- select "Start .. Programs .. WinDev Mobile 12 .. WinDev Mobile 12".

WinDev Mobile starts.

If it is run for the first time, WinDev Mobile proposes to customize your work environment. Customize your environment and answer the questions.

The following window is displayed:



- ▶ Select "Open a sample project".
  1. Select the "Mobile Windows\Beach Booking\Pocket Beach" directory.
  2. Select the "Pocket Beach.WPP" file.

3. Click "Open" to validate. The "Pocket Beach" application starts.




## The environment of WinDev Mobile

The WYSIWYG ("What You See Is What You Get") environment of WinDev Mobile 12 is conducive to productivity.

- ① The project dashboard gives an overall view of the progress of a project. It can also be used to start the different project elements.
- ② The document bar gives you the ability to re-display the elements that were previously opened.
- ③ The "wizards, Examples and Components" pane proposes a set of elements: components (elements that can be easily reused in all your projects), preset controls, examples, and so on. An invaluable time saver!
- ④ The "Project" pane returns the list of project elements: a double click performed on an element opens this element in the relevant editor. The content can be displayed in a list box or in a "TreeView" (our example).


The environment of WinDev Mobile 12 is highly intuitive. The different panes can be displayed at any time via "Display .. Toolbars".

- ▶ The dashboard enables you to easily find out whether bugs have occurred in the project, whether automated tests have been created, whether the project can be optimized. It also gives you the ability to find project elements.
- ▶ You will now display the graph of the "Pocket Beach" project: in the left part, click the vertical "Pocket Beach Project" tab or click .  
The project graph enables you to view the sequences between the different project elements.



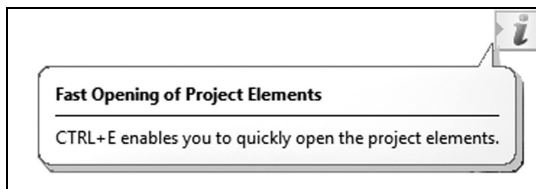
Note

**FishEye**

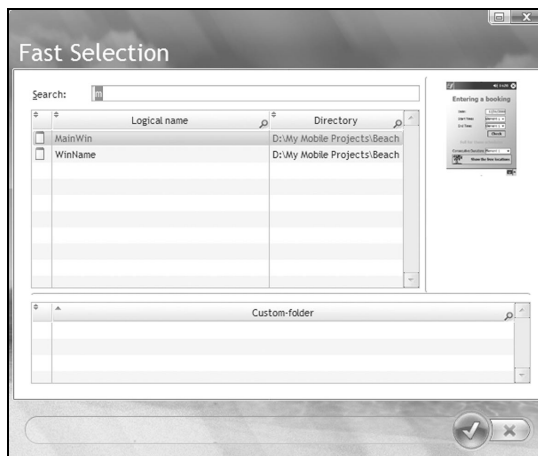
To enlarge the section of the project graph hovered by the mouse cursor, use the FishEye effect (  ).

- ▶ You will now open a window found in the "Pocket Beach" example: type the name of the window to open: "WIN\_Main".  
This window is the first window of the project: it is the first window that will be displayed when the application is started. This window is displayed with an orange background in the project graph.

You have probably noticed that information was displayed at the top right of the project graph. When this information is hovered by the mouse cursor, AAD (Aided Application Development) indicates another method for opening the project elements.



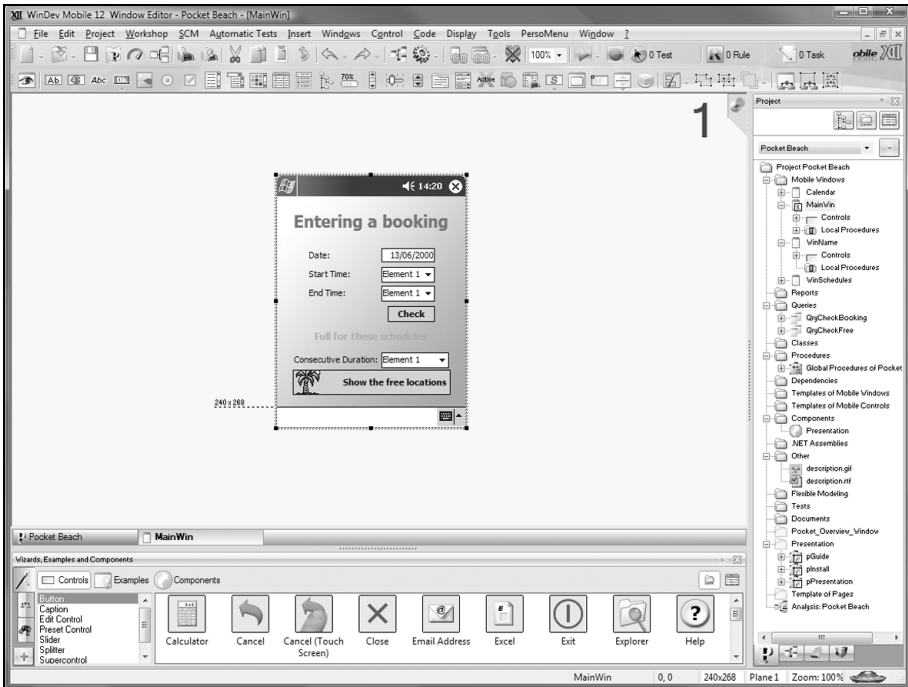
This AAD proposes to find elements via [CTRL]+[E]. Enter the letters found in the sought element and the list of corresponding elements is displayed:



- ▶ Double-click the selected window to open it.

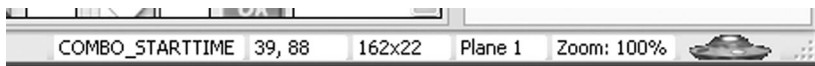
## The window editor

The window editor of WinDev Mobile enables you to easily create great windows linked to the data. The window you just opened is displayed in the window editor of WinDev Mobile. Several controls are found in this window.



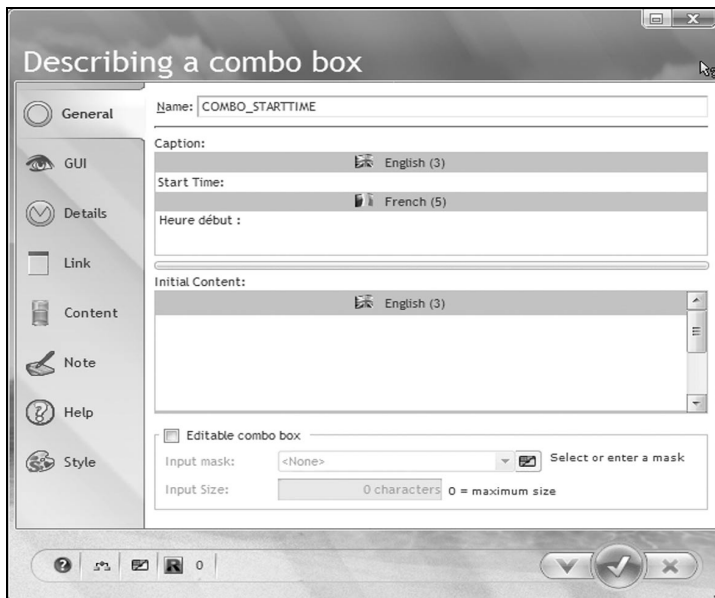
- Use the [Page up] and [Page down] keys to change the plane displayed in the window. Move your mouse in the window, above a text or an image. Click with the left mouse button. The control is selected.

You will notice that information regarding the control is displayed in the status bar.



This information corresponds to the name of the selected control, its position, its size as well as the current display zoom in the window editor.

A double click performed on the control gives access to all the characteristics of the control: name, caption, link to data file, and so on.



To close the description window, click the green button to validate.

- ▶ Select the "Date" control to view the code that is associated with it. To do so:
  1. Click the "Date" control. This control is named "EDT\_Date".
  2. Press [F2] on your keyboard. The code associated with this control is displayed in the code editor.

## The code editor

The WLanguage (the language of WinDev Mobile) enables you to describe all the requested processes. The WLanguage is a 5GL (5th generation language) that greatly simplifies the programming of your Windows applications.

The code editor is a major component of the power and efficiency of WinDev Mobile. Coding is intuitive, fast and done directly in the relevant control or window.

All the events are supported: click, double click, key down, mouse rollover, and so on.

To make coding easier and more readable, each word is colored according to its type.

For example, in the code window you just opened:

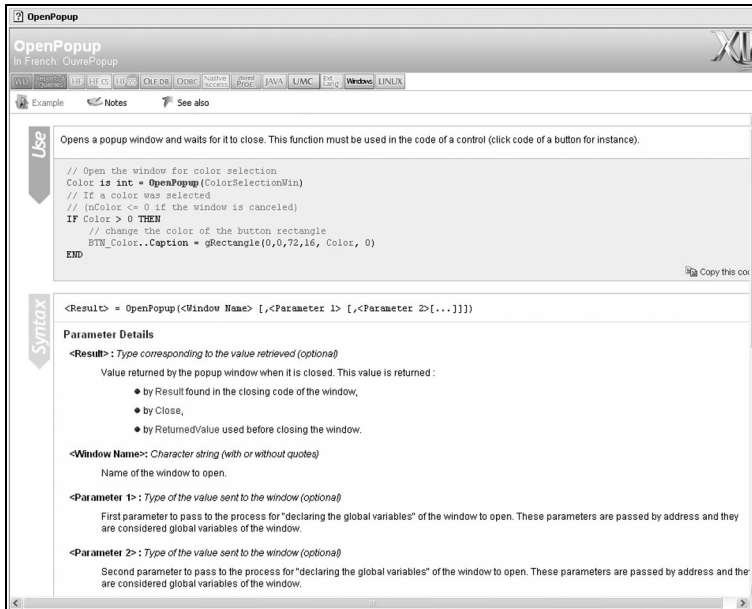
- The "OpenPopup" word is colored in blue: it is a WLanguage function.
- The "Calendar" word is colored in cyan: it is a project element.
- The "sdate" word is colored in green: it is a variable.

An assisted mode is also available for entering the source code: when typing the name of a function, the type of parameter expected by this function is displayed in a tooltip and in the status bar of

WinDev Mobile. The auto-complete feature is also available for the names of the variables or functions.

An online help is available for each control, for each editor and for each WLanguage function or property.

- ▶ You will now display the help for **OpenPopup**. To do so:
  1. Position the mouse cursor over "OpenPopup".
  2. Press [F1] on your keyboard. The help page associated with the function is displayed.



**OpenPopup**  
In French: OuvrirPopup

Example Notes See also

**Use**

Opens a popup window and waits for it to close. This function must be used in the code of a control (click code of a button for instance).

```
// Open the window for color selection
Color is int = OpenPopup(ColorSelectionWin)
// If a color was selected
// (nColor <= 0 if the window is canceled)
IF Color > 0 THEN
  // change the color of the button rectangle
  BTN_Color..Caption = gRectangle(0,0,72,16, Color, 0)
END
```

**Syntax**

<Result> = OpenPopup(<Window Name> [,<Parameter 1> [,<Parameter 2>{...}]])

**Parameter Details**

**<Result>**: Type corresponding to the value retrieved (optional)  
Value returned by the popup window when it is closed. This value is returned :

- by Result found in the closing code of the window,
- by Close,
- by ReturnedValue used before closing the window.

**<Window Name>**: Character string (with or without quotes)  
Name of the window to open.

**<Parameter 1>**: Type of the value sent to the window (optional)  
First parameter to pass to the process for "declaring the global variables" of the window to open. These parameters are passed by address and they are considered global variables of the window.

**<Parameter 2>**: Type of the value sent to the window (optional)  
Second parameter to pass to the process for "declaring the global variables" of the window to open. These parameters are passed by address and they are considered global variables of the window.

This help page displays the function documentation for all the available products, languages and platforms. To display the documentation for a single product, language or platform, select this product or platform in the "Display" list.

## The data model editor

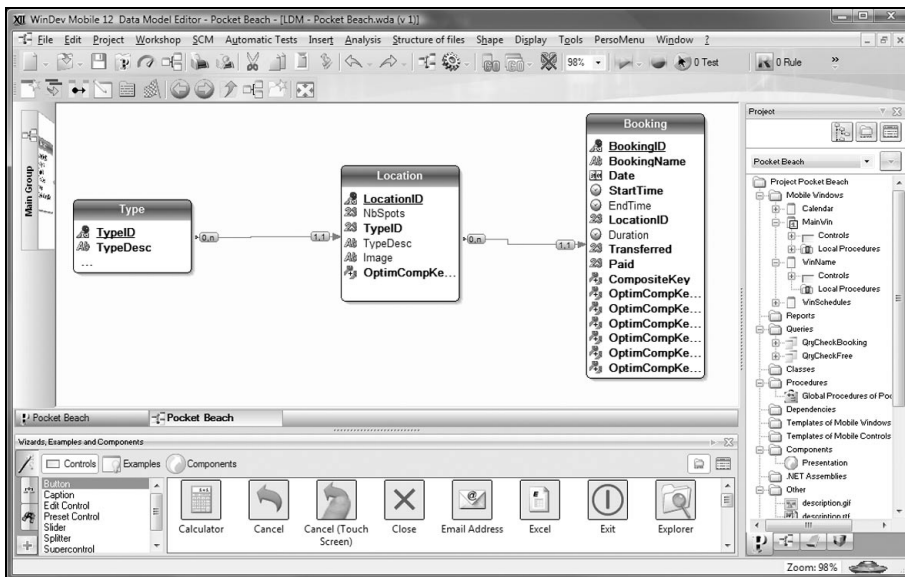
The analysis is an important phase when describing an application. The analysis is used to describe the structure of the data used by the application.

WinDev Mobile enables you to easily define the analyses, based on new or existing data. It is all done very intuitively.

The data model editor enables you to create your database, the data files and the links between these data files.

- ▶ You will now open the analysis found in the "Pocket Beach" example. To do so, select "Project .. Load the analysis".

The data model editor opens.



This analysis contains several data files linked together.

WinDev Mobile 12 enables you to create applications linked to the data via RAD (Rapid Application Development, "Workshop .. Full application RAD").

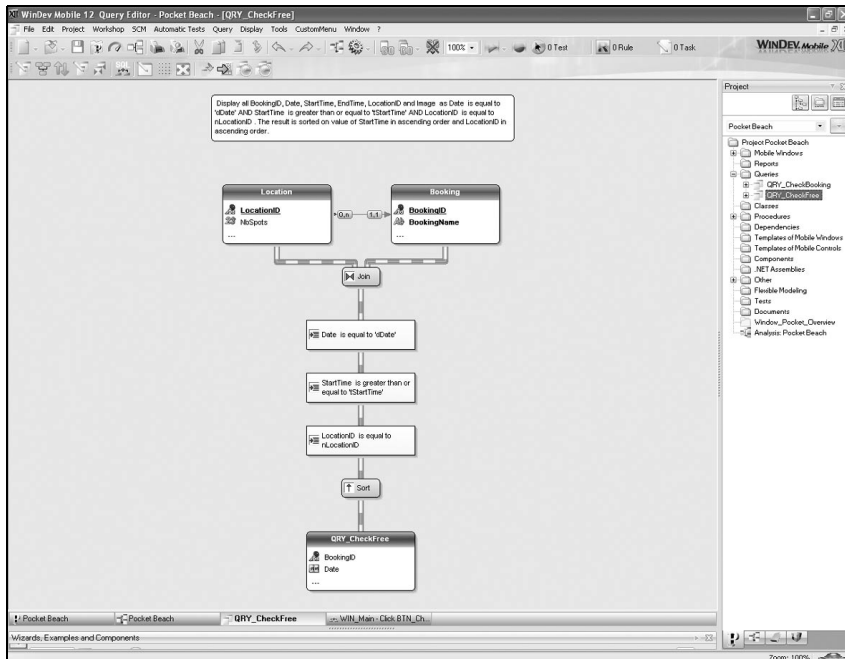
## The query editor

The query editor enables you to automatically create queries based on data files. This simplifies programming: windows, tables, combo boxes, reports, ... can be based on queries.

► You will now open a query found in the "Pocket Beach" example. To do so:

1. Display the dashboard ( ).
2. In the "Fast Selection" tab (middle right), type "QRY\_CheckFree".
3. The "QRY\_CheckFree" query is found. Double-click its name to open it.

The query editor is displayed:



This query is used to display the free spots.

The data source of a query can be a Hyper File Mobile database or an external database: Oracle Lite, SQL Server CE, AS/400, and so on.

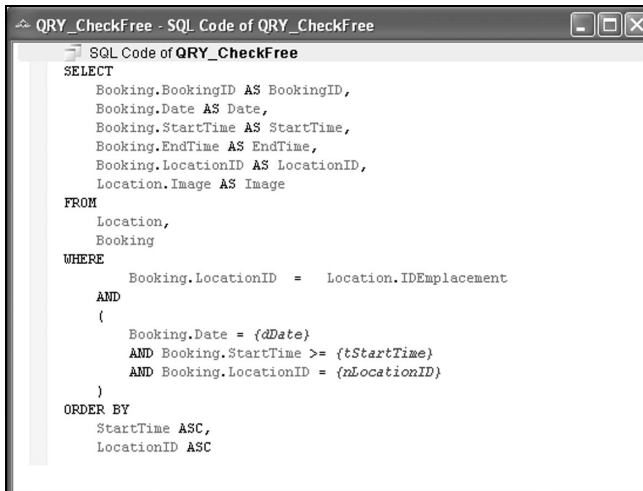
Choose the items to include, enter the selection conditions via the wizard for query creation, ... and the query is automatically created.

The queries are generated in everyday language, which makes future modifications a lot easier!

No need to know the SQL language to create powerful queries: the SQL code is automatically generated by WinDev Mobile.

- ▶ Select "Query.. SQL code".

The SQL code generated by WebDev Mobile is displayed in the query editor:



```
SQL Code of QRY_CheckFree
SELECT
  Booking.BookingID AS BookingID,
  Booking.Date AS Date,
  Booking.StartTime AS StartTime,
  Booking.EndTime AS EndTime,
  Booking.LocationID AS LocationID,
  Location.Image AS Image
FROM
  Location,
  Booking
WHERE
  Booking.LocationID = Location.IDEmplacement
AND
  (
    Booking.Date = {dDate}
    AND Booking.StartTime >= {tStartTime}
    AND Booking.LocationID = {nLocationID}
  )
ORDER BY
  StartTime ASC,
  LocationID ASC
```

## The report editor

The report editor enables you to create reports that can be directly printed from your applications. A wizard is automatically proposed: it asks questions so you won't forget anything!

### Creating a report is as easy as creating a window!

The principle for printing a report in Pocket PC is straightforward: a PCL file is created when the report is printed and this file is sent to the destination printer. The content of the PCL file is specific to the printer used.

WLanguage code can be entered in all the report elements: the most specific processes can be performed with no trouble.

### The setup editor

WinDev Mobile enables you to create the programs required to easily install your applications on the Pocket PCs of the end users.

Several methods can be used to install a WinDev Mobile application on a Pocket PC:

- setup in CAB format. This setup program is run on a Pocket PC.
- setup by direct copy of the executable from a PC to a connected Pocket PC.
- setup performed via a setup program. This setup program is run on a PC under Windows connected to a Pocket PC.

### To conclude

You now had a quick look at parts of the environment of WinDev Mobile 12.

You will be able to explore the main features of WinDev Mobile in details by following this tutorial.

The online help is available at any time via the [F1] key.

You also have the ability to contact our free Technical Support from our Web site (<http://www.windev.com>).

Don't forget to visit our Web site (<http://www.windev.com>) on a regular basis to get updates of the examples supplied with WinDev Mobile.

# NEW FEATURES IN WINDEV MOBILE 12

**This lesson will explain the following concepts...**

---

- Overview of the new features of WinDev Mobile.



Estimated time: 1h



EXAMPLE

The "WM12News.WPP" project corresponds to the full project with the answers to this lesson. This project is found in "\\Tutorial\Answers\WM12News".



To open this project, select "? .. Tutorial .. New features of this version".

## Overview

---

Several new features are included in the new version of WinDev Mobile.

This lesson presents some of these new features. This lesson is intended for the developers who are already familiar with the earlier versions of WinDev Mobile and who are experienced in handling the editors of WinDev Mobile.

- ▶ We are going to open an existing project.
  1. Select "File .. Open a project".
  2. Select the "Tutorial\Answers\WM12News" sub-directory of the setup directory of WinDev Mobile 12.
  3. Select the "WM12News.WPP" project.
  4. Run the test of the project (  to run the test in simulator mode or  to run the test on the Pocket PC connected to the current computer).

We shall first present the new features displayed when the project is run, then we shall present the new features found in the different editors.

## New features found in the example

---

### Assisted input in the edit controls

An information text can now be entered in the edit controls; this text informs the user of the data that must be entered. This information text is displayed if the control is empty and it disappears as soon as a character is entered in the control.

The advantage of this information text (compared to tooltips for instance) is that this text remains visible as long as no value is entered in the control.

- ▶ Click "New features of GUI" and click "Information Text".  
The different controls found in the window include an information text.  
Click "Change the information text" to modify the text displayed in the control via **..Hint**.
- ▶ Close the window.

In the editor, the information text is entered in the "Content" tab of the description window of the edit control.

### Web Camera control

The Web Camera control is available in WinDev Mobile. Your applications can now use the camera supplied with the Smartphone or with the Mobile.

The VideoXXX functions are used to manage the camera and to record a video sequence in "ASP" format.

- ▶ In the window named "New features of GUI", click "Web Camera control".  
The different buttons found in the window are used to view the broadcast and to save it in a "test.asp" file.

We shall now take a look at the new features of WLanguage, the language of WinDev Mobile.

- ▶ Close (if necessary) the windows named "Web Camera control" and "New features of GUI", then click "New features of WLanguage".

### Using fDirSize and LengthToString

**fDirSize** is used to find out the size of a directory. This function returns the size in bytes. In most cases, the size in bytes is not the value that we want to display and several calculations must be performed to get a size in another unit (MB, KB, and so on). **LengthToString** is used to automatically convert a size in bytes into another unit.

Let's see how these two functions operate.

- ▶ Click the "fDirSize/LengthToString" button.
- ▶ Select a file and click the "fDirSize" button. The size of the directory containing the file is specified in bytes. Click the "LengthToString" button.

### Sorts in memory list boxes and tables

From now on, you have the ability to choose the sort performed on the memory list boxes and tables. You have the ability to perform a sort while taking into account:

- the uppercase characters,
- the space characters,
- the accented characters,
- the lexicographic order,
- the numeric order.

This choice can be performed:

- in the editor, in the "Details" tab of the control description.
- by programming with **ListSort** or **TableSort**.
- by programming with **..SortOption**.

The WM12News example uses the WLanguage functions to sort the list of first names.

- ▶ Close (if necessary) the "WIN\_WL\_Size" window, then click "Sorts in list boxes".
- ▶ Check the different cases.

### New syntax for SWITCH

The SWITCH statement is used to choose the action that will be run according to the value of an expression. A new syntax is available.

- ▶ Close (if necessary) the "Sorted list box" window, then click "New SWITCH syntax".

### Higher speed for FOR loops

You are already familiar with the FOR statement. This statement is used to repeat a block of code lines by modifying a control variable and by comparing this variable to a maximum value. The block of code lines is then run for each one of the values successively taken by the control variable.

The maximum value is automatically recalculated for each iteration. From now on, the maximum value can be calculated once only by using the syntax:

```
FOR I = <Lower Bound> _TO_ <Upper Bound>
...
END
```

This syntax is very useful when the iterations concern all the rows found in list boxes, tables, and so on. For example:

```
FOR I = 1 _TO_ List1..Occurrence
...
END
```

This new syntax evaluates "List1..Occurrence" once only, which speeds up the process if the table or the list box contains several rows.

- ▶ Close (if necessary) the "SWITCH" window, then click "FOR acceleration".

### Procedure with variable number of parameters

WinDev Mobile allows you to create procedures with a variable number of parameters. This type of procedure can be very useful to calculate the maximum value in a list of numbers or to move a group of controls for instance.

- ▶ Close (if necessary) the "FOR" window, then click "Procedure with variable number of parameters".
- ▶ The "Maximum Proc" tab uses a procedure to calculate the maximum value in a list of numbers. This procedure is as follows:

```
PROCEDURE Maximum(*)
nMax is int = -Infinite
FOR nSub=1 _TO_ MyParameters..Occurrence
  IF TypeVar(MyParameters[nSub]) = wlInt THEN
    IF nMax<MyParameters[nSub] THEN
      nMax = MyParameters[nSub]
    END
  ELSE
    Error("Not numeric")
    RESULT 0
  END
END
RESULT nMax
```

In this code, the MyParameters keyword is used to handle the parameters that will be passed to the procedure.

To call this procedure, use the code:

```
Maximum(1, 2, -5, 100, 29)
```

- ▶ The "Move Proc" tab uses a procedure to move a group of controls:

```
PROCEDURE Move(nPixel, *)

FOR nSub=2 _TO_ MyParameters..Occurrence
  MyParameters[nSub]..X += nPixel
  MyParameters[nSub]..Y += nPixel
END
```


## New features in the editors

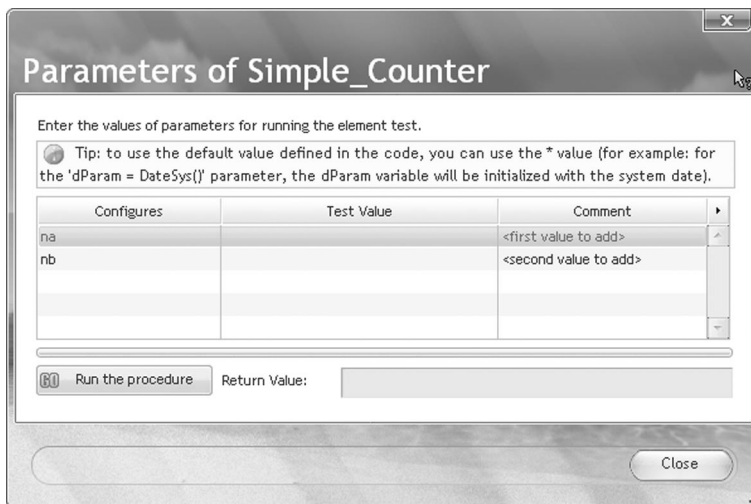
Let's now take a look at the new features found in the editors.

### Go on global procedures

From now on, you have the ability to run the test of a global procedure via GO.

The WM12News project contains a global procedure whose test will now be run.


- ▶ In the Code pane, select the global procedure named "Simple\_Counter".
  1. Double-click the name of the procedure: the code of the procedure is displayed in the code editor. This procedure adds the value of two integers and returns the result.
  2. Run the test of the procedure (  ). The following window is displayed:

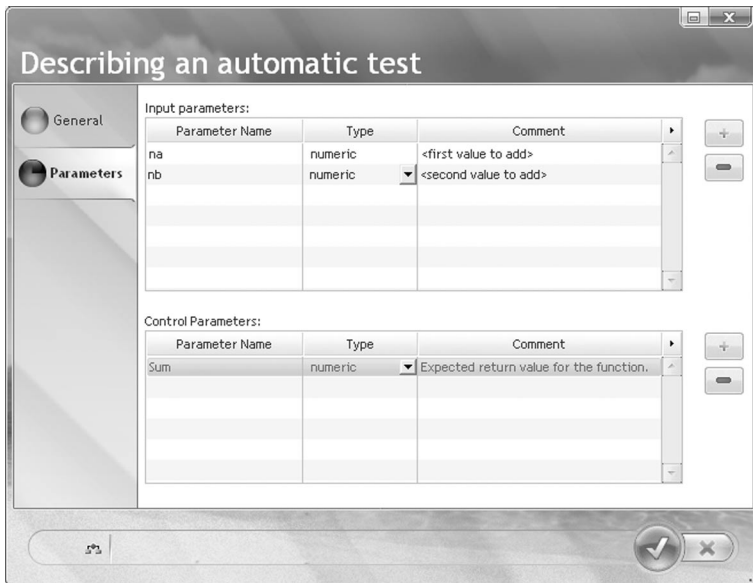


3. Enter 3 and 5 for instance, then click "Run the procedure test".
4. The test is run and the result is displayed in "Return Value".
5. Close the test window.

## Automated tests with iterations

Now that we've run the test of this procedure, why not run an automated test?

- ▶ To create an automated test on the global procedure:
  1. Select the global procedure in the "Code" pane.
  2. Select "Create a unit test" from the popup menu of the "Simple\_Counter" procedure.
  3. The test editor is displayed.
  4. Display the "Test Data" tab (if necessary).
  5. Display the test description (  ).
  6. Enter the following information:



na and nb correspond to the parameters of the procedure.

Sum corresponds to the value of the test result.

Validate.

Caution: Don't forget to replace "Controller1" by "Sum" in the code of the test.

7. Enter the following data in the "Test Data" tab:

na	nb	Sum
2	2	4
3	3	6
4	4	8

8. Go back to the "Results" tab and click "Click here the make the test available", then click "Click here to run the test".

9. The test was successfully run.

## Running an automated test on a window

From now on, an automated test can be run on a WinDev Mobile window. The test script is recorded from the actions performed in the simulator. The automated test can then be run on the Pocket PC.

- ▶ We shall now create an automated test on the window named "WIN\_Help\_Edit":
  1. Open the "WIN\_Help\_Edit" window in the editor.
  2. Select "Automatic tests .. Save a new test".
  3. Click "Start recording".
  4. Validate (if necessary) the test in the simulator.
  5. Enter the last and first names in the window and click.
  6. Close the simulator.
  7. The test editor is displayed, with the code corresponding to the window test.
  8. Make the test available. We shall now run it directly on the Pocket.

## Running the automated tests of the executable found on the Pocket

To run the automated tests directly on the Pocket, you must:

- generate the test library. This test library will be installed along with the executable on the Pocket.
- generate the executable and install it on the Pocket.
- start the test robot.

- ▶ To run the automated tests on the Pocket:
  1. Select "Automatic tests .. Generate the test library".
  2. If you did not run the test of the WM12News application on the Pocket, create the application executable ("Workshop .. Generate the executable"). Validate the default options proposed by the wizard and validate the copy of the executable onto the Pocket.
  3. Select "Automatic tests .. Run the test robot".
  4. Validate (if necessary) the login and password for connecting to the test robot (the login and password to specify are those of WinDev Mobile). This connection is required to see and enter the test results in the database of Control Centers.
  5. The test robot is displayed and it indicates the tests that can be run.
  6. Click "Run the tests".
- ▶ Once the tests have been run, close the test robot as well as the test editor.

## Trace window

Let's now talk about another new feature of the editor: the ability to display a trace window. This trace window is available in the WinDev Mobile editor, even when the test is over.

This trace window contains:

- the information displayed by **Trace**.
- the WLanguage errors that occurred in the application.

A double click performed on the information displayed in the trace window enables you to point to the corresponding code.

To display the trace window, select "Display .. Toolbars .. Trace of debugger".

► To check this feature, some statements will be added to the project:

- In the initialization code of the WIN\_Menu window, specify:

```
Trace ("Global Menu")
```

- In the initialization code of the WIN\_NewGUIFeatures, specify:

```
Trace ("GUI Menu")
```

- In the initialization code of the WIN\_NewWLFeatures, specify:

```
Trace ("WL Menu")
```

- Debug the application on the Pocket via "Project .. Test Mode .. Debug the project (Pocket)". Handle the application and close it.  
When going back to the editor, the trace window is displayed.

### Control template

WinDev Mobile enables you to define control templates. A control template is a set of controls performing a specific action (a file picker for instance).

When creating a window, control templates can be used via the "Control Template" control.

When a template is modified, WinDev Mobile proposes to modify all the windows that use this control template. Each window can be dissociated from the template, or from part of the template (some controls only), or from some parameters (the position of some controls for instance). The controls found in the control template can be moved anywhere in the host window: the control template is overloaded.

You want to create a control template from several controls found in your window? All you have to do is select "Control .. Refactoring .. Create a control template from the selection".

### Conclusion

---

We have reached the end of our short look at the new features in WinDev Mobile 12. But this is by no means all of the new features found in the product.

See the online help to discover all these new features.

DEVELOP 10 TIMES FASTER

**PART 2**

**First applications**

**XIII**  
**Express**



## LESSON 2.1. YOUR 1ST APPLICATION

### This lesson will teach you...

---

- Creating your first window that can be used on Pocket PC.
- Running the test of this window.
- Creating the executable and the setup program for this application.
- Installing this window on a Pocket PC.



Estimated time: 1h



EXAMPLE

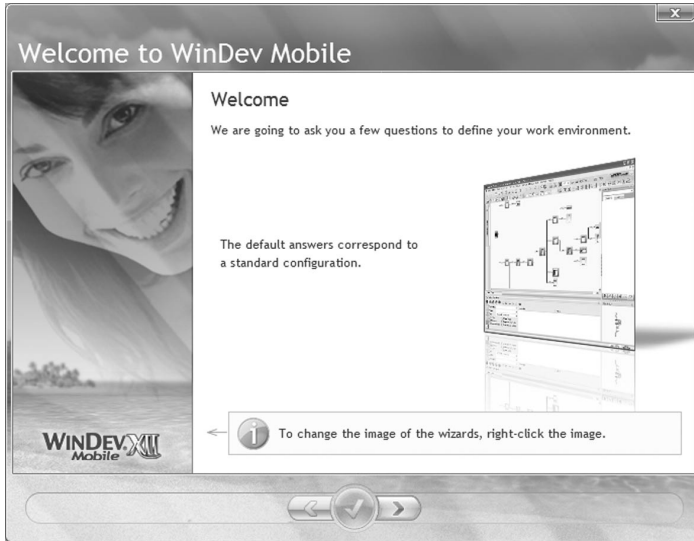
The "Get Familiar.WPP" project corresponds to the full project with the answers to this lesson. This project is found in "\\Tutorial\\Answers\\Get Familiar".

To open this project, select "? .. Tutorial .. Get Familiar".

You can follow this lesson without opening the project.

## Now down to work!

The following wizard is displayed when WinDev Mobile is started for the first time:



This wizard helps you to configure the environment of WinDev Mobile.

- Answer the questions.

All the wizards of WinDev Mobile can be customized. Your favorite image can be displayed in the wizard windows: fixed image, animated image, image chosen in the catalog or image imported by yourself (snapshot of your kids for instance).  
To customize the wizards, right-click the image of a wizard and select the image to use.

For example:



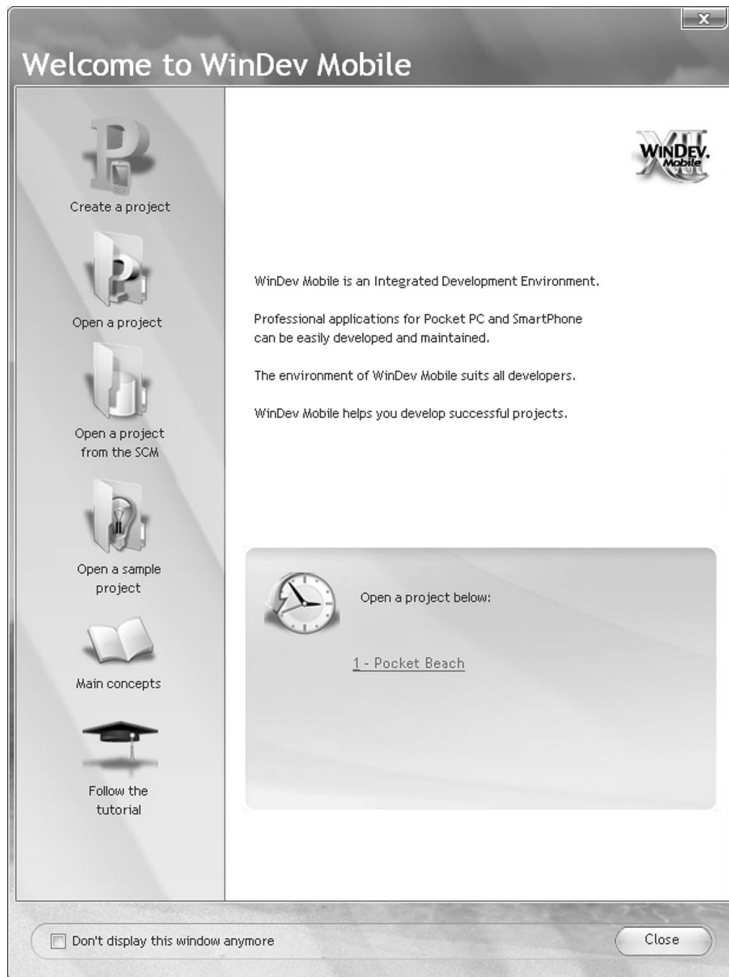
Tip



See the online help (keyword: "Wizard") for more details.

## Starting WinDev Mobile

The following window is displayed when WinDev Mobile is opened:

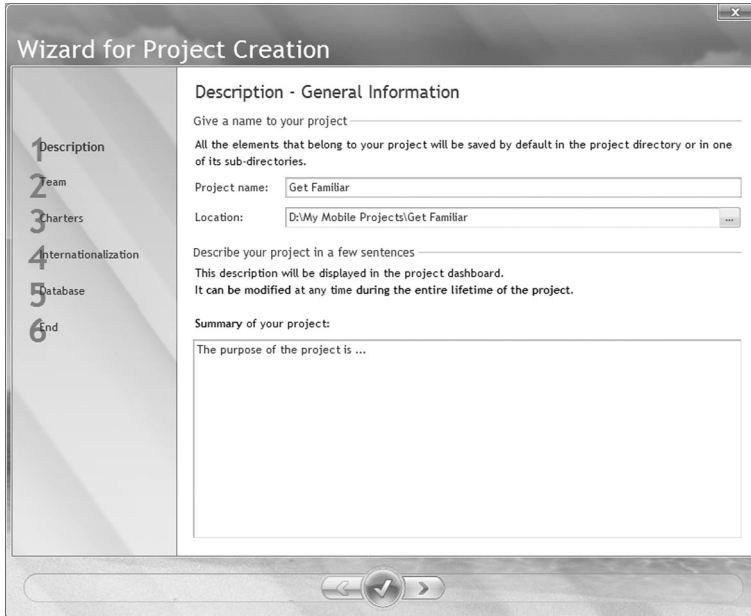


This window gives access the most common features when WinDev Mobile is opened. For our first application, we are now going to create a new project.

## How do I create a project?

To create our first project:

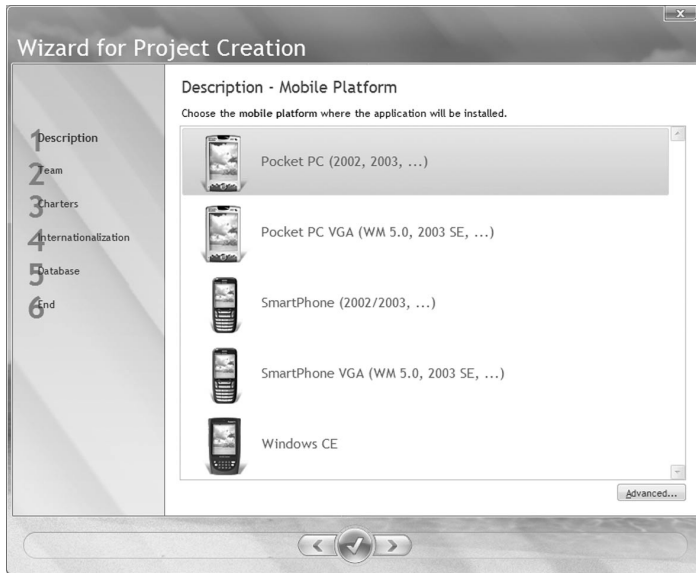
- ▶ Select "Create a project". The wizard for project creation starts.  
Note: To start this wizard, you can also select "File .. New .. Project".
- ▶ Enter the name of the project ("Get Familiar"). The name of the project is the name used to save the project file, with the "WPP" extension.  
The project directory is initialized by default ("My Mobile Projects\Get Familiar").



- ▶ Validate and go to the next screen of the wizard.  
Note: The browse buttons in the wizards are used to go to the previous plane, to go to the next plane or to validate the entire wizard with the default options.
- ▶ Validate the wizard screens until you reach the plane named "Mobile Platform".

► Select the runtime platform your application.

The runtime platform of your project will be automatically detected if a Pocket PC is currently connected to the current computer. Otherwise, the runtime platform must be specified in the wizard for project creation.



WinDev Mobile enables you to develop applications for different families of products: Pocket PC, Smartphone, Psion, and so on.

The following elements can be configured for each family:

- the dimensions of windows.
- the position of menus.
- the size of the title bar.
- the image used in test mode (simulator image). This images enables you to run the test of your application in an environment corresponding to the environment of the users.

All these characteristics define the runtime platform.



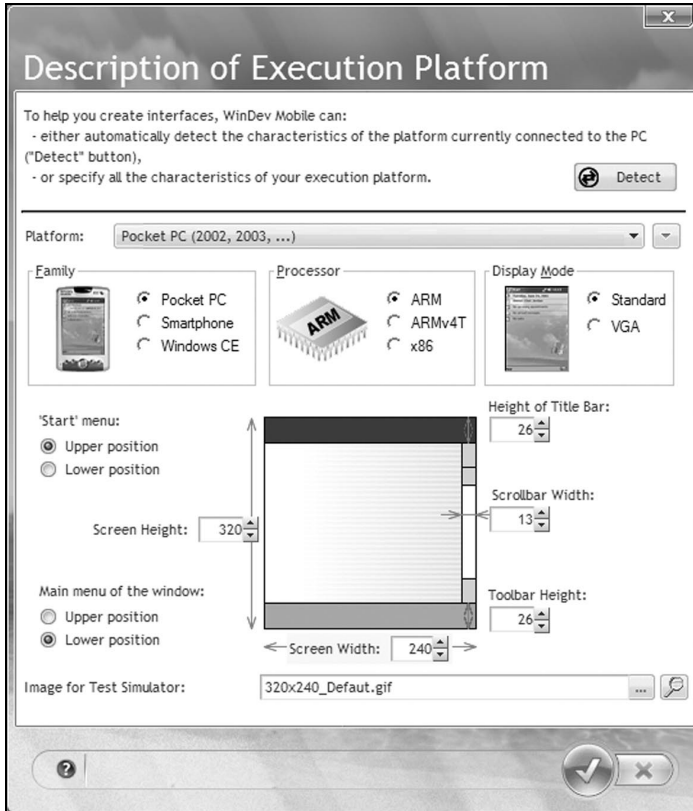
Note

The same application can be run on different platforms (Pocket PC and Smartphone for instance).

However, we advise you to create a project configuration per runtime platform. Therefore, each configuration will group all the elements specific to a runtime platform. Indeed, some characteristics can differ according to the platform used (size of the screens, management of menus, management of keyboard keys, and so on).

See 'Application on Smartphone' (page 69) for more details.

- Optional step: click the "Advanced" button. The following window is displayed:



To configure the characteristics of the platform where the project will be run, you can:

- automatically detect the characteristics of the platform for the mobile device currently connected to the current computer: click "Detect".
- specify all the characteristics of the platform manually.

Note: These characteristics can be modified later ("Platform" button in the description window of the project, "Project .. Project Description").

- Close this window, keep "Pocket PC" checked and click "Team" in the left part of the wizard.
- You will be working on your own. Click "No, I work on my own". Display the next screen.
- We shall not be using the Source Code Manager in this project. Keep "No, do not use SCM" checked. Display the next screen.
- You have the ability to use a programming charter to automatically prefix the variables. You will use the one supplied with WinDev Mobile. Display the next screen.
- You will now choose the style of your application by selecting a style book. Select the "Media Center" theme for instance.

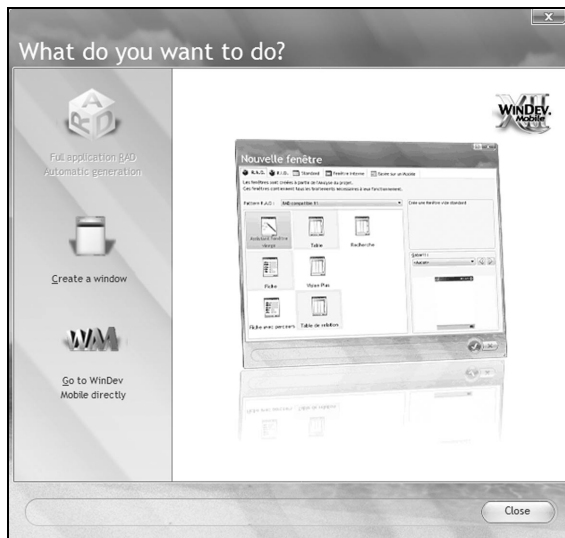
WinDev Mobile proposes several style books (or skin templates). The skin templates are used to give a specific "style" to the interface of an application.

Too often, applications built for Pocket PC all look alike: a square window, a white background, a blue title bar, gray buttons, ... a little dull, which is a pity because this is your 1st contact with the user!

WinDev Mobile enables you to select the skin template of your choice from dozens of models. Your windows, your buttons, ... are all "dressed up" in a few seconds!

You also have the ability to create your own skin templates. This is an advanced feature of WinDev Mobile. See the online help (keyword: "Skin template") for more details.

- ▶ Click the "Database" link on the left side of the wizard. This project will be linked to no analysis. Select "No, do not use a database".
- ▶ Click the validation button to validate (green button)
- ▶ The following window is displayed:



## Creating your first window

You are now going to create the following window:




This window is a stopwatch.

You may think this window is too simple, too basic, ... but we recommend that you create this window. You may well be surprised by how intuitive and easy it is to use the editor of WinDev Mobile. Furthermore, this window will teach you some principles that are fundamental for the rest of this tutorial.

► To create the window:

1. Select "Create a window". The window for creating a new window is displayed.

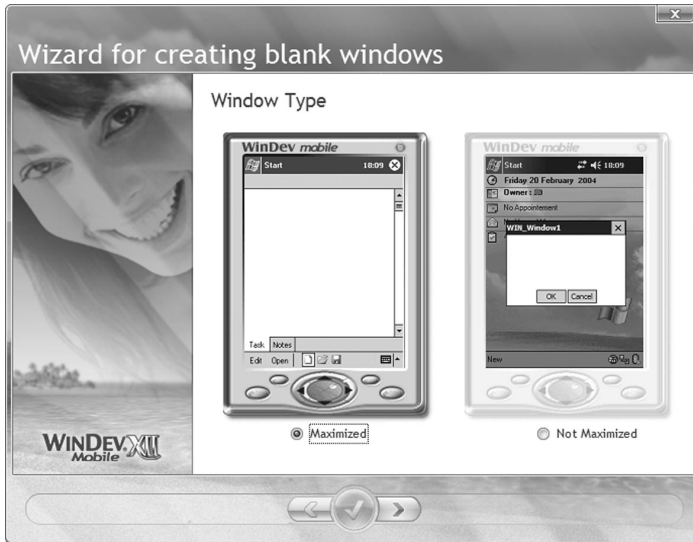
Note: To start this wizard, you also have the ability to click  in the toolbar of WinDev Mobile and to select "Window".

2. Select "'Blank window' Wizard".

By default, this window will use the skin template selected when the project was created.

3. Click "OK". The wizard for creating a blank window starts. We are now going to enter information about the window (type, name, title, and so on).

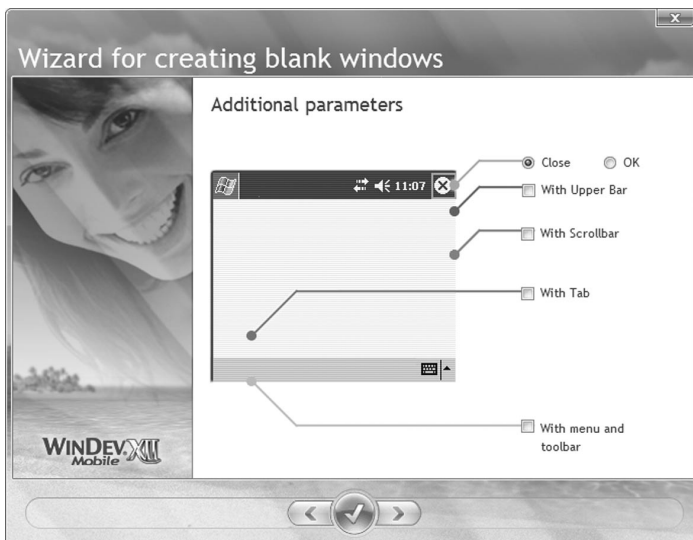
- ▶ Select the type of the window.



Two types of windows can be created in WinDev Mobile:

- Maximized window: Window that occupies the entire screen of Pocket PC.
- Non-maximized window: Window that can be resized by the user and that occupies part of the screen only.

- ▶ Select "Maximized". Go to the next screen.
- ▶ Select the elements found in your window.



Different elements can be found in a maximized window:

- **Close/OK button:** used to close or to validate the window.  
This button is associated with a process used to customize the window closing.
- **Upper bar:** used to display information, buttons, and so on.
- **Vertical scrollbar:** automatically displayed if the size of a window is greater than the resolution of the screen on the Pocket PC used.
- **Tabs:** used to organize the information on different panes. The user only has to choose the requested tab.
- **Menu and toolbar:** allow the users to easily access the application features. In an application for Pocket PC, this menu is located at the bottom of the windows.

- ▶ Keep the default options ("Close" button) and click "Next".




Note

#### Non-maximized window

The following elements can be displayed in a non-maximized window:

- **Title bar:** used to display the title of the window as well as the Close and OK buttons.
- **Close/OK button:** used to close or to validate the window.  
This button is associated with a process used to customize the window closing.
- **Window border.**

A non-maximized window can be moved by the background and resized.  
See the online help (keyword: "Window") for more details.

- ▶ Enter the name of the window: "WIN\_Stopwatch".  
The name of the window is used to handle the window in programming. This name also corresponds to the name used to save the window on disk (with the "WPW" extension).  
Enter the title of the window: "Stopwatch".
- ▶ Validate the creation wizard ("Done" button). The created window is displayed in the window editor.  
Note: All the window characteristics specified in this wizard can be modified later in the description window ("Description" from the popup menu).
- ▶ Save the window ("File .. Save" or )



Note

#### Automatic prefixing

WinDev Mobile enables you to automatically prefix all the elements that can be handled in the code editor (windows, controls, variables, and so on).

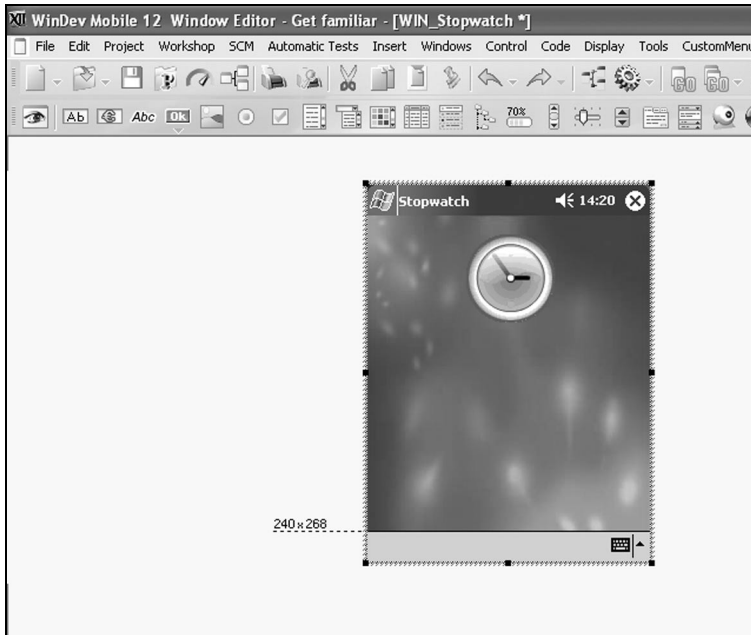
This feature enables you to easily identify the element that will be used in your programs.

See the online help (keyword: "Automatic prefixing") for more details.



## Creating the controls found in the window

### Creating the image control

To give this window a cheerful look, we are going to insert an image chosen in the image catalog of WinDev Mobile. This image represents a stopwatch and it will be animated during the timing.



► To create the image control:

1. Click  and drag the control into the window.
2. Right-click the control and select "Description".
3. Enter the name of the control: "IMG\_AnimatedImage".
4. Associate an image with this control:
  - click the "Catalog" button.
  - select "Gif animations".
  - click the animation that represents a stopwatch and validate.
  - validate the window asking for the name of the image to use.
 This image will be automatically animated.
5. Select the "Centered" display mode ("Display Mode" combo box).
6. Click  and enter the following code line in the initialization code:

```
IMG_AnimatedImage..Animation = False
```

Disable the animation of the image

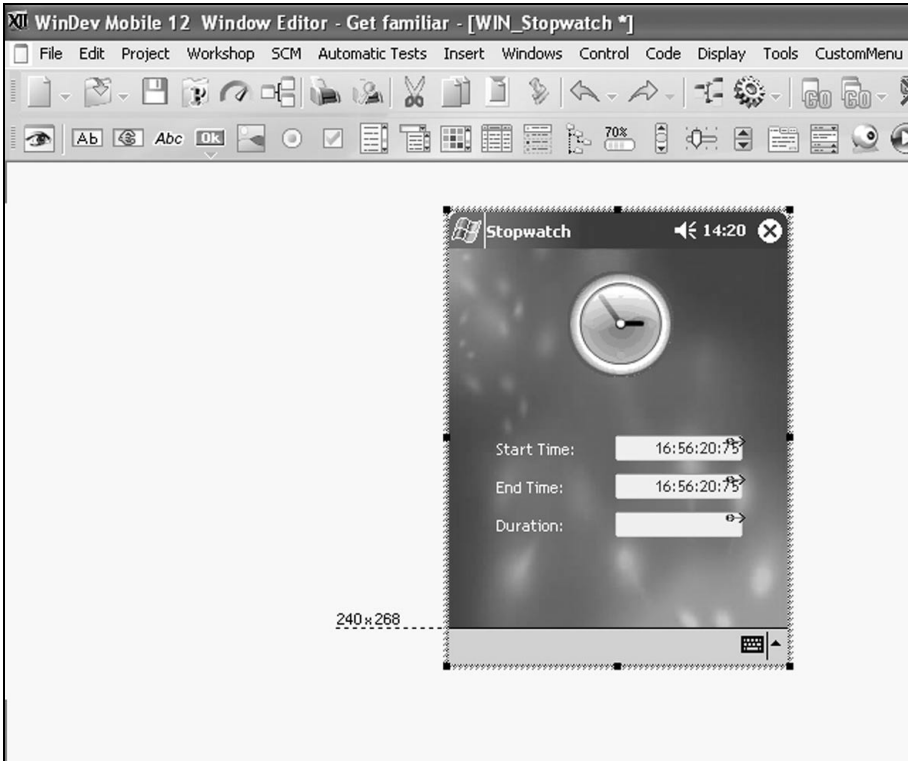


Note

In WinDev Mobile, the authorized image formats are: BMP, JPEG, GIF and ICO.

### Creating edit controls

Let's create three edit controls used to display the stopwatch information (start time, end time and duration).



► To create the first edit control:



1. Click **Ab** and drag the control into the window.
2. Right-click the control and select "Description".
3. Enter the name of the control: "EDT\_StartTime".
4. Enter the caption: "Start Time:".
5. Select the "Time" type. Indeed, this control will be used to display the start time of the stopwatch.
6. Select the input mask ("HH:MM:SS:CC") and the returned value ("HHMMSSCC"). This format will be used to calculate the duration by programming.



#### Notes

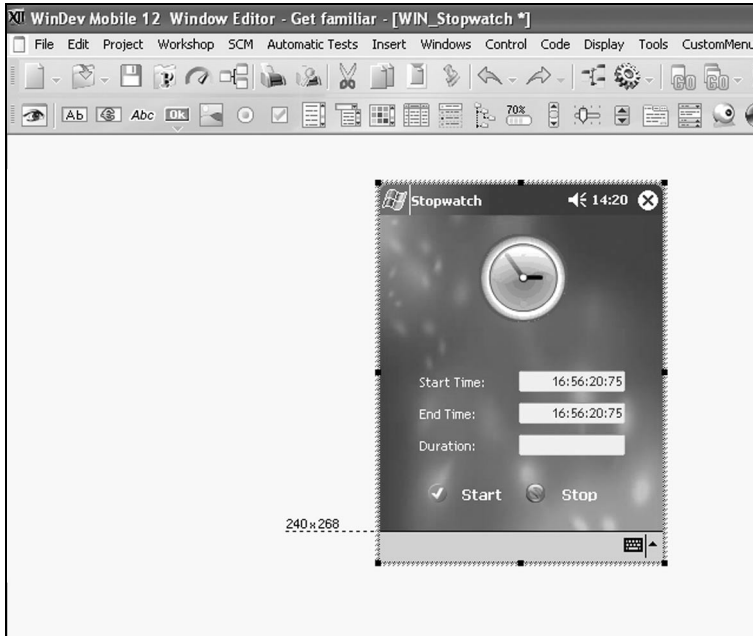
By default, the format (input mask and display mask) of the control corresponds to the numeric mask defined by the project ("Project .. Project Description", "Language" tab). Therefore, the same mask is automatically used in all the numeric controls of the application.


This feature is also very useful for the multilingual applications.

7. Select the "GUI" tab.
  8. Select "Read-only". No data can be entered by the user in this control.
  9. Validate.
- ▶ The second edit control will be created by "Copy/Paste":
1. Select the control that was just created.
  2. Press [CTRL] + [C], then [CTRL] + [V]: a new control is automatically created. Modify the name and caption of this new control:
    - its name: "EDT\_EndTime".
    - its caption: "End Time:".
- ▶ To create the third edit control:
1. Click  and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the name of the control: "EDT\_Duration".
  4. Enter the caption: "Duration:".
  5. Select the "Duration" type. Indeed, this control will be used to display the duration.
  6. Keep the input mask defined by the project.
  7. Select the "GUI" tab.
  8. Select "Read-only". No data can be entered by the user in this control.
  9. Validate.
- ▶ Save the window ("File .. Save" or .

## Creating the buttons

We are now going to create the buttons used to start and to stop the stopwatch.




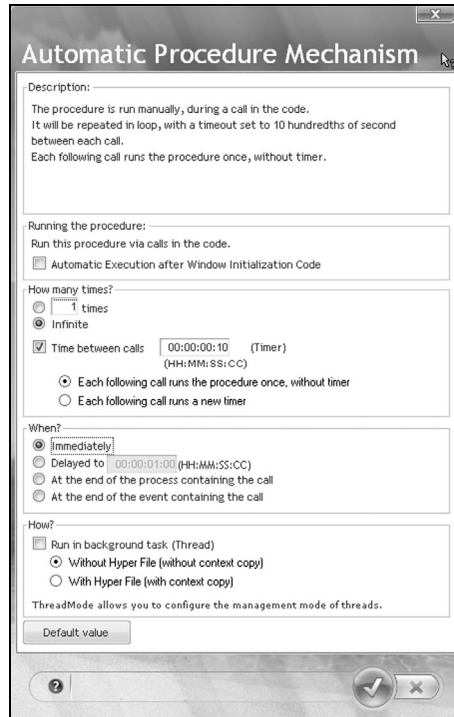
- ▶ To create the button used to start the stopwatch:
  1. Click **Ok** and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the caption: "Start".
  4. Click  and enter the following code lines:

<code>EDT_StartTime = Now()</code>	Display the start time
<code>IMG_AnimatedImage..Animation = True</code>	Animate the animated image
<code>ChronoStart()</code>	Start the stopwatch
<code>Stopwatch()</code>	Call a procedure used to calculate the time passed



- ▶ To create the procedure used to calculate the duration:
  1. Click "Stopwatch" in the code editor and select "Insert .. New local procedure" (or press [F4]).
  2. The name of the procedure ("Stopwatch") is automatically proposed. Validate this name.
  3. Enter the following code lines:

<code>PROCEDURE Stopwatch()</code>	
<code>EDT_EndTime = Now()</code>	Display the end time
<code>EDT_Duration = ChronoValue()</code>	Display the duration

- ▶ This procedure will be called every 10 hundredths of a second in order to calculate and to display the time passed:
  1. Click  found to the right of the code window of the procedure.
 The following window is displayed:



2. Select "Infinite" and "Time between calls".
3. Specify the time between the calls to the procedure: "00:00:00:10".
4. Select "Immediately".
5. Validate.

- ▶ To create the button used to stop the stopwatch:
  1. Click  and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the caption: "Stop".
  4. Click  and enter the following code lines:

```
EndAutomatedProcedure (Stopwatch)


ChronoEnd ()

IMG_AnimatedImage..Animation = False
```

Stop the automatic call to the procedure

Stop the stopwatch

Disable the animation of the image

- ▶ Save the window ("File .. Save" or ).

The development of this window is now completed. We are now going to run its test.

## Running the test of a WinDev Mobile window

WinDev Mobile proposes several types of test:

- run the test on the development computer (in simulation mode). This test simulates a Pocket PC on the development computer. This test is useful when no Pocket PC can be used by the developer. The debugger can be used.
- run the test and debug on the Pocket PC connected to the development computer. This test allows the use of the debugger while being directly run on the Pocket PC.
- run the test on the Pocket PC connected to the development computer. This test generates the executable of the application, copies it and runs it on the Pocket PC. The debugger is not available.

**GO**

Test


The result of some features depends on the platform used (operating system, format of character strings, and so on).

Differences may occur when running the project test or the window test in simulation mode and when running the application test on a Pocket PC.

See the online help (keyword: "Test, Differences between a test on Pocket PC and a simulator test") for more details.

### Test in simulation mode

Let's now test the window in simulation mode.

- ▶ Click the "GO" icon  (or press [F9]). The test of the window is run. This test is performed via a simulator.



- ▶ Run the test of the different buttons and watch the changes that occur in the window.

Any developer knows that running a program test can be a long and tiresome job. WinDev Mobile enables you to run the test of your window in ONE CLICK. This is both simple and fast!

WinDev Mobile enables you to customize the simulator used for your tests in GO mode. This enables you to choose the shape of the Pocket PC for your tests. To customize the simulator, right-click the image of the simulator and select the image to use. For example:



Tip

See the online help (keyword: "Simulator") for more details.

- Close the window ("X" button in the title bar). The editor of WinDev Mobile is redisplayed.

## Running test and debugging on Pocket PC

Let's now run the test of the window on the Pocket PC. To run this test, a Pocket PC must be currently connected to the current computer.

Before connecting a Pocket PC to a computer, we advise you to install "ActiveSync" on the PC. This software is used to synchronize the data between a PC and a Pocket PC.

In most cases, ActiveSync is supplied with the Pocket PC. ActiveSync can also be downloaded from the Internet.

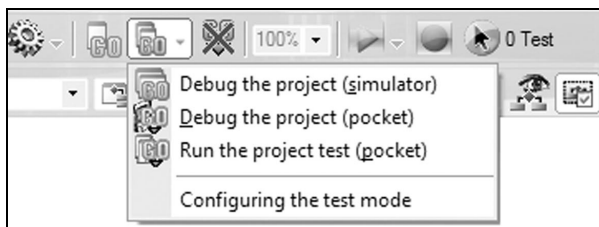
ActiveSync automatically starts when the connection is established between the Pocket PC and the PC.



Note



- ▶ First, we will debug our window on the Pocket PC. To do so, a breakpoint must be added to the code in order to start the debugger:
  1. Display the code of the "Start" button ("Code" from the popup menu).
  2. Click with the mouse in front of the first code line. A red dot is displayed: it's a breakpoint. In debug mode, the debugger will automatically start when the code line is run.
- ▶ To debug the stopwatch on the Pocket PC directly, click the arrow located to the right of the "GO" icon. A drop-down menu is displayed:



- ▶ Select "Debug the project (pocket)".
- ▶ As the executable was not created, the editor informs us that its creation must be configured in order to run the test on the Pocket PC. Answer "Yes".
  1. Define (if necessary) the window that will be displayed first.  
There is only one in our example!  
This window will be called "First project window".



Select the "WIN\_Stopwatch" window and validate.

2. The wizard for executable creation starts automatically.
3. The different options for creating the executable will be presented later. For this first test, keep the options selected by default: validate the wizard directly (green button).  
The screen named "Copying the files to Pocket PC" is displayed. Validate the wizard once again.



Note

Copying the executable to the Pocket PC may take a few minutes (indeed, the PC SOFT Framework is copied when a WinDev Mobile application is installed for the first time). The next copy of the executable to the Pocket PC will be faster.

If an antivirus is installed on the development computer, the real-time protection of the file system may significantly slow down the copy of the executable. To shorten this duration, disable this protection when copying the executable.

The window is automatically opened on the Pocket PC:




- ▶ Click the "Start" button. The debugger is automatically displayed on the PC. The different code lines can be run step by step.



Note

On-the-fly corrections cannot be made during this test.

- ▶ Stop the debugger (  ). The application is automatically stopped on the Pocket PC.

### Direct test on the Pocket PC

Let's now run the final test, without debugger.

- ▶ To run the test of the stopwatch on the Pocket PC, click the arrow located to the right of the "GO" icon. A drop-down menu is displayed.
- ▶ Click "Run the project test (pocket)".
- ▶ As the executable was already created when running the test with debugger, WinDev Mobile recreates the executable with the same options. The window is automatically opened on the Pocket PC.
- ▶ Run the test of the different buttons and watch the changes that occur in the window. Although the breakpoint is still found in the code, a click performed on the "Start" button does not trigger the debugger.
- ▶ Close the window ("X" button in the title bar).



Note

When the test is run on the Pocket PC, you have the ability to disconnect the Pocket PC from the PC and to continue to use the application.

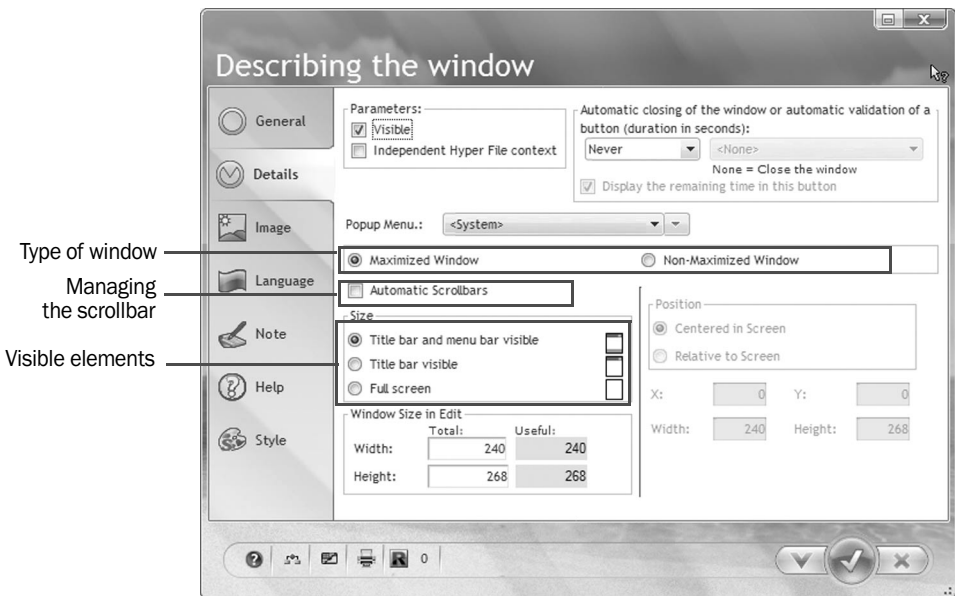
However, if you close the application (or the window), the test can only be run if the Pocket PC is reconnected to the PC.

To start the application on the Pocket PC while the Pocket PC is not connected to the PC, you must create the application executable (see 'Creating the executable', page 65).

## Characteristics of a WinDev Mobile window

All the characteristics of a WinDev Mobile window defined during its creation can be modified in the description of this window.

For example:



Reminder: To open this window, select "Description" from the popup menu of the WinDev Mobile window (or double-click the WinDev Mobile window).


## Changing the type of the window

Now let's see what happens when the window type changes.

- ▶ Select "Description" from then popup menu of the WinDev Mobile window. The description of this window is displayed.
- ▶ Display the "Details" tab and/or the "Style" tab and modify the characteristics of the window (non-maximized window, title bar, "OK" button, status bar, and so on).
- ▶ Apply the modifications and view the modified window directly.



Tip

To get help for a specific option, click  then click the requested option.



Note

The controls can be anchored in WinDev Mobile. Anchoring enables you to use the same window on different runtime platforms.

## Specific process when closing the window

The "OK/Close" button found in the title bar of the window is associated with a specific process. This process is used to customize the closing or the validation of the window.

By default, this button automatically closes the window.

Let's display a dialog box for when closing the window.

- ▶ To display a dialog box when closing the window:
  1. Make sure that the "Close" button is found in the title bar of the window ("Style" tab of the description window).
  2. In the editor, display the popup menu of the "Close" button (right mouse click) and select "Code of Button". The "Before closing with OK/Close button" process is displayed.
  3. Enter the following code lines:

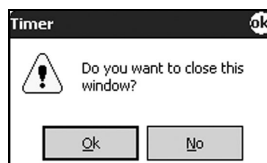
```
IF YesNo("Do you want to close this window?")...
    = False THEN

    ReturnToCapture ()
END
```

Display a dialog box

No: the window remains opened  
Yes: the window is automatically closed

4. Run the test the window and close it with the "X" button.





Tip


In the "Before closing with OK/Close button" process, *ReturnToCapture* is used to cancel the window closing.

## Creating the executable program

### Creating the executable program

Your window is now created. You had the ability to run its test while it was developed. But how do I create the executable?

► To create the executable from the environment of WinDev Mobile:

1. Select "Workshop .. Generate the executable" or click .

The wizard for executable creation starts.

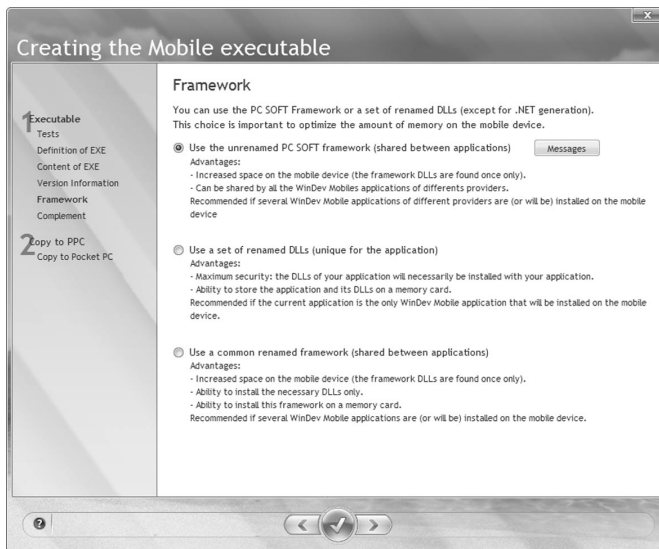
2. A screen inform us that no automated test was created. Go to the next screen.

By default, the executable program will be created with the same name as the project.

3. Click the "Catalog" button to select the icon associated with the executable.

4. Choose an image among the proposed ones (a stopwatch for instance). Double-click the image and validate.

5. Click the "Framework" link located to the left of the wizard. **WinDev 12 "Express" version : This feature is not available in this trial version.**



Each WLanguage function is associated with a WinDev Mobile library (".DLL" file). In order for the executable to operate properly, all the necessary libraries must be installed with the executable. The full set of the WinDev Mobile libraries is also called Framework.

To install the Framework, WinDev Mobile proposes:

- **to use the PC SOFT Framework:** the WinDev Mobile Framework will be installed in the "Windows\PC SOFT\WD12.0" directory of the Pocket PC. This Framework is copied to the Pocket PC when the application is installed for the first time or when the application is updated. This option is used to limit the amount of memory occupied on the Pocket PC: all the WinDev Mobile applications use the same Framework.
- **to use a renamed Framework:** the renamed Framework will contain the WinDev Mobile DLLs used by the application. The location of this Framework is chosen by the developer. This Framework is renamed to avoid conflicts with the other Frameworks. This Framework is copied to the Pocket PC when the application is installed for the first time or when the application is updated. This option is used to limit the amount of memory occupied on the Pocket PC: several WinDev Mobile applications use the same renamed Framework.
- **to use a renamed common Framework:** each WinDev Mobile DLL used by the application must be renamed (next plane of the wizard). These DLLs will be installed in a shared directory in order to be used by several applications.

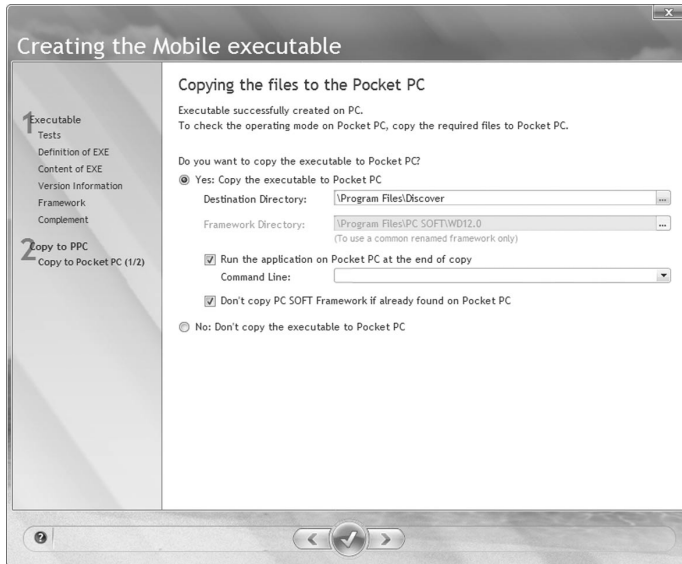
6. Validate the creation of the executable (green button). The default parameters will be automatically taken into account.

See the online help (keyword: "Executable") for more details.

Congratulations! You now can create stand-alone executable programs. This program is compatible with all the versions of Windows for Pocket PC!

## Running the test of the executable on the Pocket PC connected to the development computer

Once the executable is created, the wizard proposes to copy the executable to the Pocket PC connected to the current computer.



- ▶ To copy the executable to the Pocket PC:
  1. Select "Yes: Copy the executable to the Pocket PC".
  2. Specify the destination directory of the executable on the Pocket PC.
  3. Select "Run the application on Pocket PC at the end of copy".
  4. Select "Don't copy PC SOFT Framework if already found on Pocket PC".
  5. Validate (green button).

As soon as the executable is copied, the application is automatically started on the Pocket PC. The application can be started from the Pocket PC directly without having to connect the Pocket PC to the PC.

## Distributing your application

---

WinDev Mobile simplifies the distribution of your applications.

A wizard helps you create the setup procedure ("Workshop .. Create the setup procedure").

The setup program (in CAB or MSI format) will be generated in a directory in order to be copied onto different media.

To install your application:

- in CAB format: copy the setup program (".CAB" file) to the Pocket PC and run it.
- in MSI format: run the setup program (".MSI" file) on a PC:
  - if this PC is connected to a Pocket PC, the application will be immediately installed on the Pocket PC.
  - if no Pocket PC is connected, the setup will be performed the next time the PC and the Pocket PC are synchronized.

The files required by the application are automatically installed in the specified setup directory.



Note

When creating the setup program of an application, you have the ability to configure the autorun of this application.

The application can be started:

- when the Pocket PC is started (after reset for instance).
- once the Pocket PC is synchronized (via ActiveSync).
- at the end of the standby mode ("On/Off" button of Pocket PC).

You can also configure the autorun of the application by programming with **AutoRunAdd** and **AutoRunDelete**.

## LESSON 2.2. APPLICATION ON SMARTPHONE

### This lesson will teach you...

---

- Creating your first window that can be used on Smartphone.
- Running the test of this window.
- Sending and reading SMSs.
- Managing the SIM card.



Estimated time: 40 min



EXAMPLE

The "SMS.WPP" project corresponds to the full project with the answers to this lesson. This project is found in "\\Tutorial\\Answers\\SMS". To open this project, select "? .. Tutorial .. SMS".

You can follow this lesson without opening the project.

## Overview

We are now going to create an application used to send SMSs and to read incoming SMSs.

An SMS (Short Message Service) corresponds to a text message (up to 160 characters) sent on a cell phone.

To be able to use the SMS functions, the application must be installed:

- on a Pocket PC with phone access (GSM).
- and/or on a Smartphone.



EXAMPLE

Example

The "Sending SMS" example supplied with WinDev Mobile contains a project that can be used on Pocket PC and a project that can be used on PC. These two examples are used to send SMSs.

These examples can be accessed from the "Wizards, Examples and Components" pane.

## How do I handle an SMS?

To send or read an SMS with WinDev Mobile, all you have to do is enter the description of the SMS: its message, the recipient number, and so on. This information is stored in the SMS structure.

The SMS structure is as follows:

ReceiveDate	Date and time when the SMS was received. Note: On Pocket PC 2002, this member contains the date and time when the SMS was read.
Retry	Boolean (True by default) Indicates whether the message must be regularly sent if wrong reception.
Subscript	Integer corresponding to the subscript of the incoming SMS.
Message	Character string containing the outgoing message or the incoming message (up to 160 characters).
Number	Character string containing the phone number of the caller or sender.
CountryPrefix	Character string containing the international prefix (33 by default for France). If the recipient number starts with "0" and if a national prefix is specified, "0" will be replaced by this prefix. If no national prefix is specified, use a number in international format. For example, 33612345678.

NumberType	Indicates the type of number used: <ul style="list-style-type: none"> <li>• smsInternationalNumber (default value): these numbers can be accessed anywhere and are in 06.xx.xx.xx.xx. format</li> <li>• smsNationalNumber: short numbers, accessible within the country only</li> </ul>
------------	---

Once the SMS is described, all you have to do is use:

- **SMSSend** to send the SMS.
- **SMSFirst** and **SMSNext** to read the incoming SMSs.

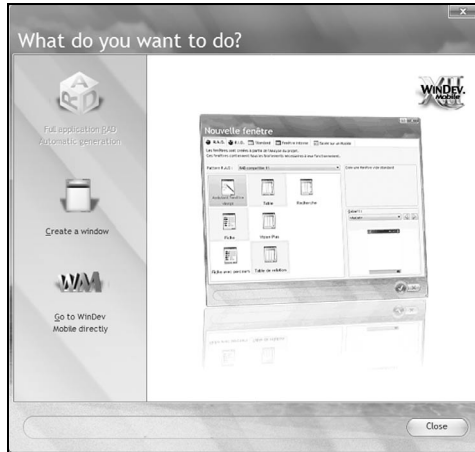
## Creating the application

We are going to create an application containing two windows. A window will be used to enter the text of the SMS as well as the recipient number. In this window, a check box will be used to define whether the SMS must be regularly re-sent in case of non-reception. A second window will be used to list the incoming SMSs.

### Creating the project

- ▶ To create our application:
  1. Select "File .. New .. Project .. WM Smartphone". The wizard for project creation starts.
  2. Specify the name and the summary of the project: "SMS" and "Project for sending and reading SMSs".
  3. Click the "Team" link on the left side of the wizard.
  4. You will be working on your own. Select "No, I work on my own" and go to the next screen.
  5. We shall not be using the Source Code Manager in this project. Keep "No, do not use SCM" checked. Go to the next screen.
  6. You have the ability to use a programming charter to automatically prefix the variables. You will use the one supplied with WinDev Mobile. Go to the next screen.
  7. Choose the style of your application by selecting a skin template. Select the theme of skin template: "Media Center" for instance.
  8. Click the "Database" link to validate. This project will be linked to no analysis. Select "No, do not use a database".
  9. Go to the next screen and close the wizard (green button).

The following window is displayed:



Let's now create the window used to send SMSs.


### Creating the window for sending SMSs


You are now going to create the following window:



Compared to the "WIN\_Stopwatch" window created in the previous lesson, you can see straight-taway that the Smartphone windows are really small.

- ▶ To create the window for entering SMSs:
  1. Select "Create a window". The wizard for creating a new window is displayed.

Note: To start this wizard, you also have the ability to click  in the toolbar of WinDev Mobile and to select "Window".

2. Select "'Blank Window' Wizard".  
By default, this window will use the skin template selected when the project was created.
3. Validate. The wizard for creating a blank window starts.
4. Keep the "Maximized" mode and go to the next screen.
5. Keep "With menu" checked and to the next screen.
6. Enter the name and the title of the window: "WIN\_SendSMS" and "Sending SMSs".
7. Validate (green button). The created window is displayed in the window editor.
8. Save the window ("File .. Save" or .



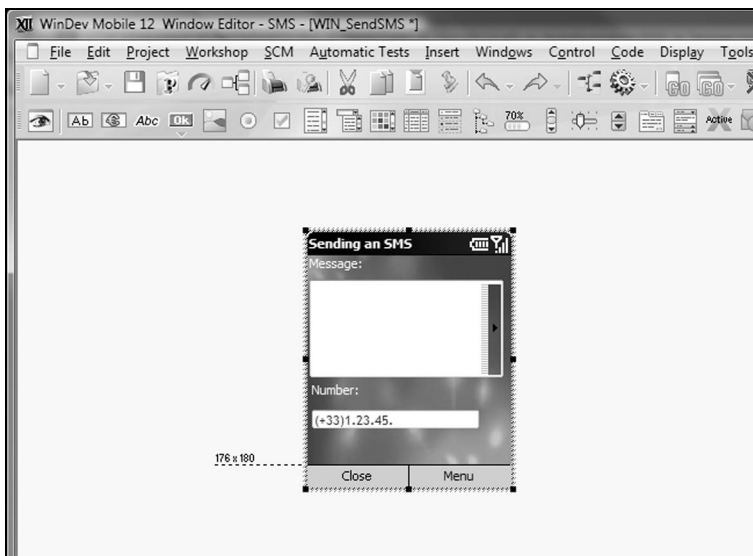
Note


The characteristics of a window for Smartphone are practically identical to the characteristics of a window for Pocket PC. These characteristics were presented in the previous lesson.

## Creating the controls found in the window

### Creating edit controls

We are going to create two edit controls used to enter the message and the recipient number.



- ▶ To create the first edit control:
  1. Click  and drag the control into the window.

2. Right-click the control and select "Description".
3. Enter the name: "EDT\_SMSMessage".
4. Enter the caption: "Message:".
5. Select the input size: "160 characters". Indeed, the maximum size for an SMS cannot exceed 160 characters.
6. Select "Multiline Text".
7. Select the "Details" tab and select "Vertical Scrollbar".



Note

#### Smartphone mode

You will notice that "Smartphone Mode" is checked. This option is used to automatically manage the display of the control in a Smartphone.

For example, when typing in a multiline control:


- on a Pocket PC: the size of the control is not changed.
- on a Smartphone: the size of the control is increased to occupy all the available space.

8. Select the "Style" tab and select a style in which the control caption is "overlapping".
9. Validate.



Note

The Smartphone keys enable you to enter the text of the message.  
See 'Input mode' (page 125) for more details.


- To create the second edit control:
1. Click  and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the name: "EDT\_SMSNumber".
  4. Enter the caption: "Number:".
  5. Select the input mask: "Phone number".



Note

When entering a numeric edit control or an edit control that uses an input mask containing digits only, the input mode automatically changes to "123" mode: only the digits associated with the keys can be displayed.

See 'Input mode' (page 125) for more details.

6. Select the input size: "13 characters". Indeed, the maximum size for a phone number cannot exceed 13 characters.
  7. Select the "Style" tab and select a style in which the control caption is "overlapping".
  8. Validate.
- Save the window ("File .. Save" or .

### Creating the check box

We are now going to create the check box used to define whether the SMS must be regularly re-sent in case of non-reception.

- ▶ To create this check box:
  1. Click  and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the name: "CBOX\_Resend".
  4. Enter the caption of the option: "Resend if not received".-
  5. Click the "Style" tab and choose a style "Without caption".
  6. Validate.

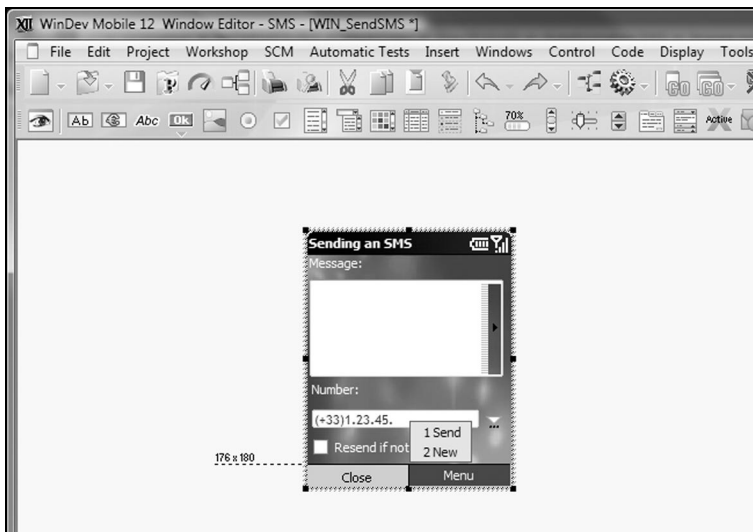


Note

To check an option of a check box, give focus to the option (with the "joystick") and press the "joystick" to validate.  
See 'Input mode' (page 125) for more details.

### Describing the menu options

By default, two menus are automatically created at the bottom of the window. The "Done" menu is used to close the application. We are going to add sub-options used to send the SMS and to describe a new one.



- ▶ To add the sub-options:
  1. Click the menu to the right and press [Enter] to add a sub-option.
  2. In this new sub-option, enter the caption: "Send".
  3. Press [Enter] again to validate the addition of this sub-option.
  4. Press [Enter] again to add another sub-option.
  5. Enter the caption ("New") and press [Enter] again to validate the addition of this second sub-

option.

6. Select "Send" and press [F2].

7. Enter the following code lines:

<code>SMS.Message = EDT_SMSMessage</code>	Initialize the SMS structure
<code>SMS.Number = EDT_SMSNumber</code>	
<code>SMS.Retry = CBOX_Resend</code>	
<code>bResSend is boolean = SMSSend()</code>	Send the SMS
<code>If bResSend = False THEN</code>	Error occurred ?
<code>Error(ErrorInfo(errMessage))</code>	
<code>ELSE</code>	
<code>Info("SMS sent")</code>	
<code>END</code>	

8. Go back to the window editor, select "New" and press [F2]. Enter the following code line:

<code>SMSReset()</code>	Reset the SMS structure
<code>RESET()</code>	Reset the window controls



Note

When sub-items are added to the menu, the numbers "1" and "2" are automatically added beside the caption of each sub-item. These numbers give direct access to the requested sub-item via the Smartphone keys.

Only two menus can be displayed at the same time on Smartphone. These menus can be accessed via the buttons found below the Smartphone screen.

The menu options can be accessed via the numeric keys and/or via the "joystick" of the Smartphone.

We are now going to create a window used to read the incoming SMSs.

## Creating the window for reading SMSs

► To create the window:

1. Select "File .. New .. Window". The wizard for creating a new window is displayed.

2. Select "'Blank window' Wizard".

By default, this window will use the skin template selected when the project was created.

3. Validate. The wizard for creating a blank window starts.

4. Keep the "Maximized" mode and go to the next screen.

5. Keep "With menu" checked and to the next screen.

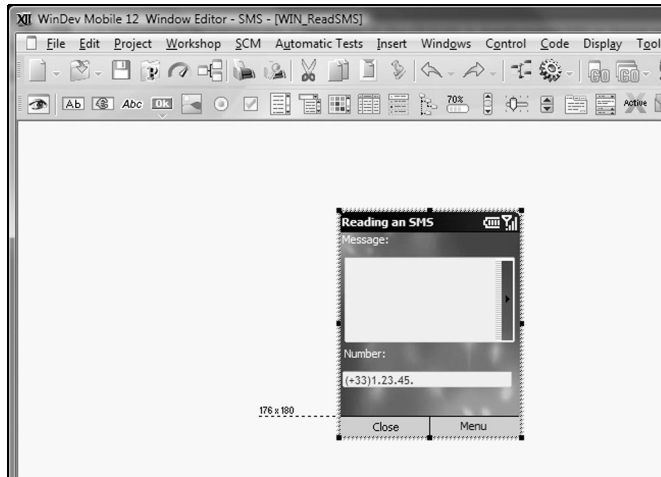
6. Enter the name and title of the window: "WIN\_ReadSMS" and "Reading SMSs".

7. Validate (green button). The created window is displayed in the window editor.

## Creating the controls found in the window


### Creating edit controls

We are going to create two edit controls used to view the message and the sender number.



- ▶ To create these two edit controls, we will copy the "EDT\_SMSMessage" and "EDT\_SMSNumber" controls described in the "WIN\_SendSMS" window:
  1. Open (if necessary) the "WIN\_SendSMS" window (via the bar of opened documents or via the "Project" pane).
  2. Select the "EDT\_SMSMessage" and "EDT\_SMSNumber" controls.  
Reminder: To select several controls at a time, keep the [Ctrl] key down while you select the requested controls with the mouse.
  3. Right-click the controls and select "Copy".
  4. Display the "WIN\_ReadSMS" window, right-click the window and select "Paste".
  5. Position the controls in the window.

When the window is opened, these controls will display the message and the sender number for the first incoming SMS.

- ▶ Save the window ("File .. Save" or .
- ▶ To display the characteristics of the first incoming SMS when the window is opened:
  1. Right-click the window and select "Code".
  2. Enter the following code lines in the process for declaring the global variables:

```
gnSubscript is int
```

Declare the nSubscript variable containing the subscript of the SMS displayed

### 3. Enter the following code line in the initialization process:

```
SMSFirst (smsSIM)
```

Points to the first incoming SMS stored in the SIM card

```
EDT_SMSMessage = SMS.Message
```

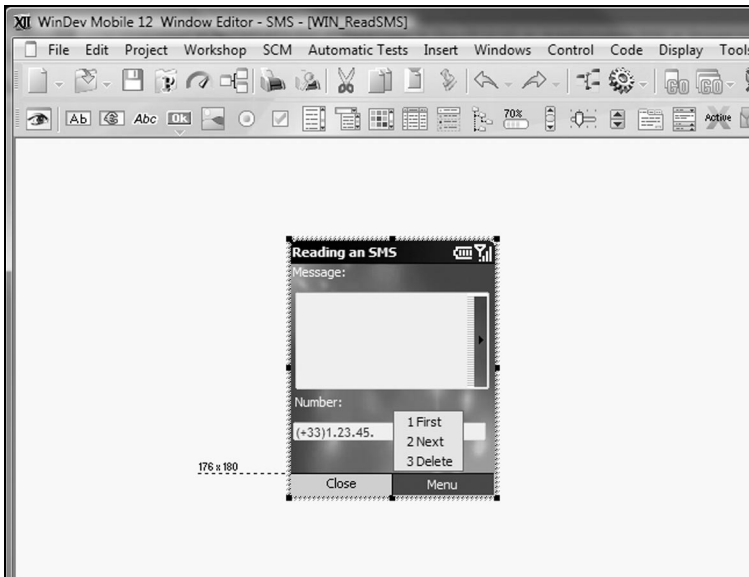
```
EDT_SMSNumber = SMS.Number
```

```
gnSubscript = SMS.Subscript
```

Retrieves the message, number and subscript of the first incoming SMS

### Describing the menu options

We are going to add sub-options used to view the other incoming SMSs and to delete a specific SMS.



#### ► To add the sub-options:

1. Click the menu to the right and press [Enter] to add a sub-option.
2. In this new sub-option, enter the caption: "First".
3. Press [Enter] again to validate the addition of this sub-option.
4. Press [Enter] again to add another sub-option.
5. Enter the caption "Next" and press [Enter] again to validate the addition of this second sub-option.
6. Do the same to add the "Delete" sub-option.
7. Select "First" and press [F2].
8. Enter the following code lines:

```
SMSFirst (smsSIM)
```

Points to the first incoming SMS stored in the SIM card

```
EDT_SMSMessage = SMS.Message
```

```
EDT_SMSNumber = SMS.Number
```

```
gnSubscript = SMS.Subscript
```

Retrieves the message, number and subscript of the first incoming SMS

9. Select "Next" and press [F2].
10. Enter the following code lines:

```
SMSNext (smsSIM)
```

Points to the next incoming SMS stored in the SIM card

```
EDT_SMSMessage = SMS.Message
EDT_SMSNumber = SMS.Number
gnSubscript = SMS.Subscript
```

Retrieves the message, number and subscript of the next incoming SMS

11. Select "Delete" and press [F2].
12. Enter the following code lines:

```
SMSDelete (smsSIM, gnSubscript)
```

Deletes the SMS displayed

We are now going to create a window used to display either the window for sending an SMS, or the window for reading an SMS.

### Creating the selection window


You are now going to create the following window:



- To create the window:
  1. Select "File .. New .. Window". The wizard for creating a new window is displayed.
  2. Select "'Blank Window' Wizard".

By default, this window will use the skin template selected when the project was created.



  3. Validate. The wizard for creating a blank window starts.

4. Keep the "Maximized" mode and go to the next screen.
5. Keep "With menu" checked and to the next screen.
6. Enter the name and title of the window: "WIN\_Selection" and "Your choice".
7. Validate (green button). The created window is displayed in the window editor.
8. Save the window ("File .. Save" or )

## Creating the controls found in the window

### Creating the buttons

We are now going to create two buttons used to display the window for sending SMSs and the window for reading SMSs.

- ▶ To create the first button:
  1. Click  and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the caption: "Sending SMSs".
  4. Click "All the actions". The window of preset actions is opened.
  5. Select "Open a window" and select the "WIN\_SendSMS" window.
  
- ▶ To create the second button:
  1. Click  and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the caption: "Reading SMSs".
  4. Click "All the actions". The window of preset actions is opened.
  5. Select "Open a window" and select the "WIN\_ReadSMS" window.

We will now check the operating mode of this window.

## Running the test of this application

As already seen, several types of test are proposed by WinDev Mobile:

- run the test on the development computer (in simulation mode).
- run the test and debug on the Pocket PC connected to the development computer.
- run the test on the Pocket PC connected to the development computer (by generating the executable of the application).

Sending and reading SMSs can be performed from a Pocket PC equipped with a phone access (GSM) or via a Smartphone.

Therefore, the test of this application cannot be run on the development computer (in simulation mode).

Indeed, when running a test from the development computer, a WLanguage error occurs when "Send" or "New" are used.

**To run the test of this application, a device for managing SMSs must be connected to the development computer.**

- ▶ To run the test of the SMS application:
  1. Connect a Pocket PC equipped with a phone access or a Smartphone to the development computer.
  2. Click "Debug the project (pocket)" (or select "Project .. Test mode .. Debug the project (pocket)").
  3. Define (if necessary) the window that will be displayed first. Select the "WIN\_Selection" window.
  4. The application is automatically started on the Pocket PC:



Test the different features of the application.

The debugger of WinDev Mobile is always available when this test is run. All you have to do is add a breakpoint to the code displayed in the code editor, the debugger will be automatically started when the corresponding code is run.



Note

To read and/or delete SMSs on a Smartphone, the executable of the WinDev Mobile application as well as its framework (WinDev Mobile libraries) must be digitally signed. A certificate is required to perform this operation.

Note: No specific signature is required to send SMSs.

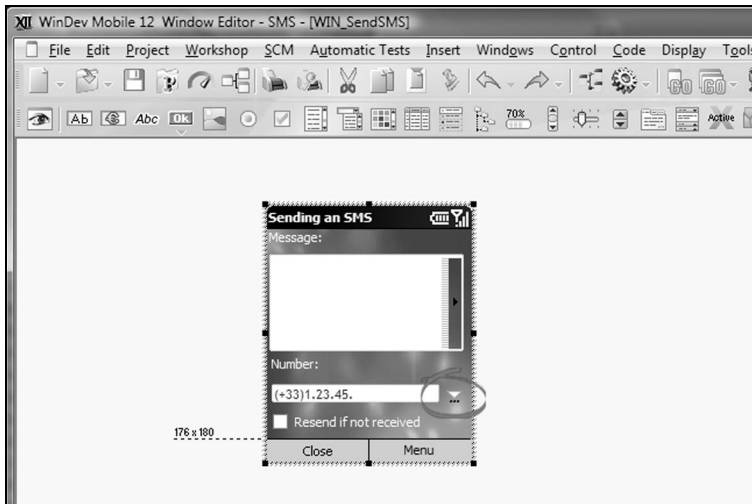
## Improving the application: managing the SIM card

We will now complete this application by allowing the user to retrieve a number from his personal directory (SIM card).

### Adding the necessary elements

#### Creating the button control

We are going to add a button to the window for sending SMSs in order to display all the numbers found in the SIM card (i.e. in your personal directory).



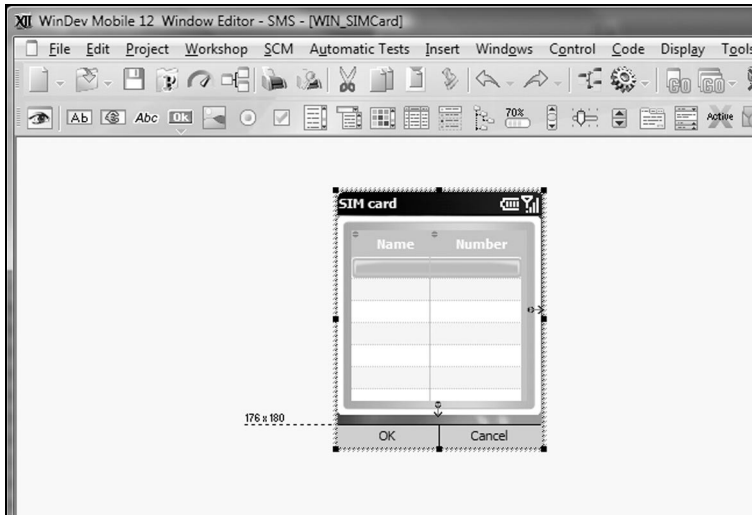
Note




To press a button, give focus to the button (via the "joystick") and press the "joystick" to validate.  
See 'Input mode' (page 125) for more details.

- ▶ To create this button:
  1. Open (if necessary) the "WIN\_SendSMS" window.
  2. Click **Ok** and drag the control into the window.
  3. Right-click the control and select "Description".
  4. Delete the button caption.
  5. Associate an image with this control:
    - click the "Catalog" button.
    - enter "Arrow" in the "Search" control and click "Find".
    - select the requested image and validate.
    - validate the window asking for the number of states to use.
  6. Validate.

### Creating the information window

We will now create the following window. This window is used to display the information found in the SIM card.




- ▶ To create the window:
  1. Click  in the toolbar of WinDev Mobile and select "Window". The window for creating a new window is displayed.
  2. Select "'Blank window' Wizard".  
By default, this window will use the skin template selected when the project was created.
  3. Validate. The wizard for creating a blank window starts.
  4. Keep the "Maximized" mode and go to the next screen.
  5. Keep "With menu" checked and to the next screen.
  6. Enter the name and title of the window: "WIN\_SIMCard" and "SIM card".
  7. Validate (green button). The created window is displayed in the window editor.
  8. Save the window ("File .. Save" or ).
  
- ▶ To create the table:
  1. Click  and drag the control into the window. The wizard for table creation starts.
  2. Keep "I want to fill the table by programming". Go to the next screen.
  3. Select "Read-only table". Go to the next screen
  4. Specify the number of columns: "2" and enter the title of each column: "Name" and "Number".  
Go to the next screen.
  5. Enter the name: "TABLE\_Directory".
  6. Enter the title: "Directory:".
  7. Validate (green button).  
Resize the table if necessary.

8. Press [F2] and enter the following code lines in the "Initialization" process:


<code>nSubscript is int</code>	
<code>FOR nSubscript = 1 TO SIMNbContact()</code>	Add each entry to the table
<code>    TableAdd(TABLE_Directory, ...</code>	
<code>        SIMRead(nSubscript)</code>	
<code>END</code>	
<code>TableSort(TABLE_Directory, COL_Name)</code>	Sort the elements found in the table by name
<code>TableSelectPlus(TABLE_Directory, 1)</code>	Select the 1st row

► To define the menu to the left:

1. Select the menu to the left.
2. Right-click the menu and select "Option Description".
3. Enter the caption: "OK".
4. Click  and replace the existing code by the following code line:

<code>Close("", COL_Number)</code>	Retrieve the number and close the window
------------------------------------	--

► To define the menu to the right:

1. Select the menu to the right.
2. Right-click the menu and select "Option Description".
3. Enter the caption: "Cancel".
4. Click  and enter the following code line:

<code>Close("", "")</code>	Close the window without retrieving the number
----------------------------	--

#### Open the "WIN\_SIMCard" window

The development of the "WIN\_SIMCard" window is now completed. Let's now go back to the "WIN\_SendSMS" window to enter the code for opening the "WIN\_SIMCard" window.

► To open the "WIN\_SIMCard" window by programming:

1. Display the "WIN\_SendSMS" window.
2. Select the button that was previously created and press [F2].
3. Enter the following code line:

<code>EDT_SMSNumber = Open(WIN_SIMCard)</code>	Open the "WIN_SIMCard" window When this window is closed, the phone number is displayed in "EDT_SMSNumber"
--	---

The management of the SIM card is now completed. We will now run the test of our application.

**Reminder: To test this feature, a device managing SMSs must be connected to the development computer.**

## Running the test of SIM card management

- ▶ To run the test of SIM card management:
  1. Connect (if necessary) a Pocket PC equipped with a phone access or a Smartphone to the development computer.
  2. Click "Debug the project (pocket)" (or select "Project .. Test mode .. Debug the project (pocket)").

The application is automatically started on the Pocket PC:



3. Try to retrieve the phone number of someone you know.

## Creating the executable program and distributing the application

The method for creating the executable program and for distributing an application for Smartphone is identical to the method used for a Pocket PC application.

These features were already presented in the previous lesson.



DEVELOP 10 TIMES FASTER

**PART 3**

**Database**

**XIII**  
**Express**



## LESSON 3.1. INTRODUCTION

**This lesson will teach you the following concepts...**

---

- The formats of the databases supported by WinDev Mobile.



Estimated time: 10 min

## Format of the databases

A WinDev Mobile application can be used to handle data. The format of these databases can be:

- Hyper File Mobile (Classic or Client/Server), the database system supplied with WinDev Mobile.
- CEDB, the database system for Pocket PC.
- Oracle Lite, the Oracle database for Pocket PC.
- SQL Server CE, the SQL Server database for Pocket PC.
- AS/400, the AS/400 database for Pocket PC.
- and so on.

### Hyper File Mobile

The Hyper File Mobile format is the database format supplied with WinDev Mobile. This database format is compatible with WinDev, WinDev Mobile and WebDev.

It is a freely distributable Relational DBMS.

This format is identical to the Hyper File format of WinDev and WebDev (".WDD" file, data files, and so on).

However, the available size on Pocket PC being restricted and the operating system of Pocket PC being limited, the following features are not supported by Hyper File Mobile in Classic mode:

- the transactions.
- the log process.
- the management of file locks and record locks.
- the management of files in Hyper File 5.5 format.

Note: Only the files in Hyper File 5.5 format are not supported by Hyper File Mobile in Client/Server mode.



Note

You still have the ability to open and to use an analysis that uses one of these features (replication, log operations, and so on). The same analysis can be used in a WinDev Mobile application and in a standard WinDev application.

The Hyper File Mobile format allows you to:

- access the records quickly,
- optimize the search time,
- handle large databases,
- synchronize the Hyper File Mobile files found on a Pocket PC with the Hyper File files found on a PC,
- and so on.

Most of the features of Hyper File Mobile are available for WinDev Mobile (file link, queries, filters, views, and so on).



Test

When running the test (in simulation mode) of a WinDev Mobile application that handles Hyper File Mobile files, the files used are those found on the **PC**.

See 'Interactions between application' (page 117) for more details.

## CEDB

The CEDB format is a database format for Pocket PC.

A CEDB database corresponds to a ".CDB" file. A CEDB database can contain several data files (also called "tables").

Two types of CEDB databases are available:

- the standard CEDB databases, that correspond to the databases found by default on the Pocket PC. These databases contain the following data files: "Tasks", "Contacts" and "Appointment", and so on.
- the other CEDB databases (called custom databases), that correspond to Access databases (".MDB" files) previously exported from a PC.

Note: When an Access database (".MDB" file) is copied to a Pocket PC (via the file explorer), this database is automatically changed into a CEDB database (".CDB" file).

A CEDB database can be handled:

- from a WinDev Mobile application.
- from a standard WinDev application.

These operations are performed via the ***cdbXXX*** functions of WLanguage.



Note

**Caution:** The structure of the CEDB databases is not intended to process large amount of data. We recommend that you use the Hyper File Mobile databases. Furthermore, Hyper File Mobile enables you to benefit from all the features available for WinDev Mobile (RAD, file link, and so on).



Test

When running the test (in simulation mode) of a WinDev Mobile application that handles a CEDB database, the database used is found on the **Pocket PC**.

See 'CEDB files' (page 98) for more details.

See 'Interactions between applications' (page 117) for more details.

**Oracle Lite, SQL Server CE and AS/400**

These database formats can be accessed via a Native Access. An additional module is required to use one of these native accesses.

Contact PC SOFT Sales Department for more details.

## LESSON 3.2. HYPER FILE MOBILE FILES

**This lesson will teach you the following concepts...**

---

- Generating a Full Application RAD.
- Managing the data files.



Estimated time: 20 min



EXAMPLE

The "Product Catalog.WPP" project corresponds to the full project with the answers to this lesson. This project is found in "\\Tutorial\\Answers\\Product Catalog". To open this project, select "? .. Tutorial .. Hyper File Mobile (answer)".

You can follow this lesson without opening the project.

## Overview

The format of the Hyper File Mobile database was presented in the previous lesson.

Furthermore, you are already familiar with the main characteristics of this format.

This lesson explains how you can generate a full application by RAD for Pocket PC from a Hyper File Mobile analysis.

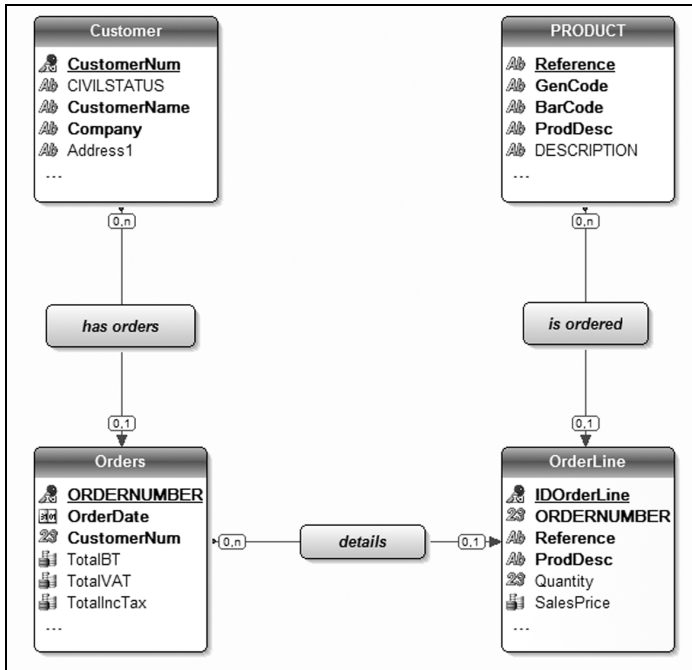
## Generating a full application

An existing project will be used to generate this application.

- ▶ Open the "Product Catalog.WPP" project found in the "Tutorial\Exercises\Product Catalog" sub-directory of the setup directory of WinDev Mobile.

Note: To open this project directly, select "? .. Tutorial .. Hyper File Mobile (exercise)".

This project is associated with the following analysis:



Simple files have been deliberately chosen for this lesson:

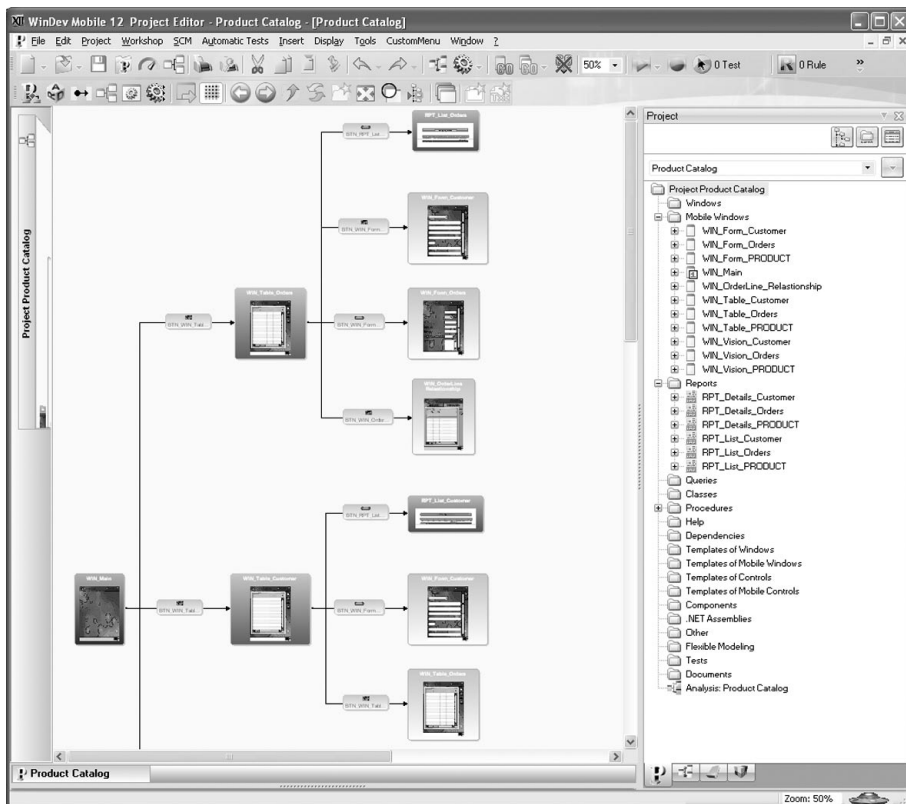
- a CUSTOMER file
- an ORDERS file
- an ORDERLINE file
- a PRODUCT file

We will now create the windows and reports found in this project.

► To generate the full application:

1. Select "Workshop .. Full Application RAD". The wizard for the RAD module starts.
2. Select "Simple RAD Pattern". Go to the next screen.
3. All the analysis files will be used for the generation. Select all the files and go to the next screen.
4. The entry points for the application (proposed by the menus) will be:
  - Customer
  - Orders
  - Product
5. Go to the next screen.
6. Validate. The RAD generation starts.

The project graph is automatically drawn once the application is generated. This full application can already be used.



The method for creating a full application by RAD is identical in WinDev Mobile and in WinDev. Another example of the legendary simplicity of WinDev.

The test of this application can be run:

- in simulation mode on the development computer: select "Project .. Test mode .. Run the project test (simulator)".
- on the Pocket PC connected to the current computer: select "Project .. Test mode .. Run the project test (pocket)".



Note

#### RID (Rapid graphical Interface Design)

WinDev Mobile gives you the ability to generate windows containing the controls linked to the analysis items. The source code required for these windows to operate must be written by the developer.

See the online help (keyword: "RID") for more details.

## Managing the data files

### Generated data files

When calling **HCreation** or **HCreationIfNotFound**, the data files (".FIC" extension), the index files (".NDX" extension) and the memo files (".MMO" extension) are automatically created.

In **test mode** ("GO"), these files are created on the development **PC** (in the executable directory).

At **run time** on the Pocket PC, these files are created on the **Pocket PC** (in the executable directory).

In all cases, this directory can be modified before the file is created by **HChangeDir** and **HSubstDir**.

Note: In Pocket PC, a specific format must be used for the paths of the files. See 'Managing the directories in Windows for Pocket PC' (page 110) for more details.

### Copying the data files

The Hyper File data files can be copied from a PC to a Pocket PC, and conversely (via the file explorer for instance).

### Synchronizing the data files

WinDev Mobile allows you to:

- synchronize the data files found on the PC with the data files found on the Pocket PC.
- synchronize the data files found on the Pocket PC with the data files found on the PC.

This synchronization is automatically performed via ActiveSync when the Pocket PC is connected to the PC.

This synchronization is configured when creating the setup program of the application.

See the online help (keyword: "WDSynchro") for more details.

**WDMap and Pocket Map**

WDMAP is used to view and handle a Hyper File Mobile data file found on the PC when developing an application.

Pocket Map is used to view and handle a Hyper File Mobile data file found on the Pocket PC when running the application test or when using an application. Pocket Map is an example supplied with WinDev Mobile.

## LESSON 3.3. CEDB FILES

**This lesson will teach you the following concepts...**

---

- Adding, modifying and deleting records.
- Browsing records.
- Searching for records.
- Viewing records.



Estimated time: 1h



EXAMPLE

The "Contact.WPP" project corresponds to the full project with the answers to this lesson. This project is found in "\\Tutorial\\Answers\\Contact".

To open this project, select "? .. Tutorial .. CEDB".

## Overview

---

The format of the CEDB database was already presented in 'Introduction' (page 90).

We shall now use a WinDev Mobile application that handles the **standard CEDB database** named "Contacts".

- ▶ Open the "Contact" project found in "\Tutorial\Answers\Contact".  
Note: To open this project directly, select "? .. Tutorial .. CEDB".

This application contains:

- a window used to list all the contacts and to find a contact.
- a window used to display the details of a contact.

## Handling a record

---

The CDB structure of WinDev Mobile enables you to handle a record found in a standard CEDB database. A specific structure is available for each standard database ("Tasks", "Contacts" and "Appointments").

For example, the **cdbContact** structure corresponds to the "Contacts" database.

Each structure variable corresponds the one of the file items in the database.

For example, the **cdbContact** structure contains the following variables: `cdbContact.Name`, `cdbContact.City`, `cdbContact.Email`, and so on.

These variables enable you to easily read, add and modify a record in a standard CEDB database.



Note

### Custom CEDB databases

You have the ability to create custom CEDB databases. Their content is specific to each application. Therefore, a preset structure cannot be used to handle them.

To handle the items of these databases, use **cdbCol** and **cdbWriteCol**.

See the online help (keywords: "cdbCol" and "cdbWriteCol") for more details.

## Adding a record

To add a record to a standard CEDB database, you must:

- specify the variables of the structure corresponding to the database used.
- add the record to the database (**cdbAdd**).

In the "Contact" project, the code used to add a record to the CEDB database ("Contacts") is found in the "Validate" button of the "WIN\_ContactForm" window.

- ▶ To view this code:
  1. Open the "WIN\_ContactForm" window.
  2. Select the "Validate" button.

### 3. Display the code of the button ([F2]).

```
// Retrieve the new data
cdbContact.Name = EDT_Name
cdbContact.FirstName = EDT_FirstName
cdbContact.CivilStatus = COMBO_Title..DisplayedValue
...
cdbAdd("", cdbContact)
```



Note

#### Custom CEDB databases

To add a record to a custom CEDB database, you must:

- specify the values of the items for the new record (***cdbWriteCol***).
- add the record to the database (***cdbAdd***).

### Modifying a record

To modify a record in a standard CEDB database, you must:

- point to the record to modify (***cdbRead***, ***cdbReadSeek***, and so on).
- modify the variables of the structure corresponding to the database used.
- save the record in the database (***cdbModify***).

The number of the record to modify can also be directly specified in ***cdbModify***.

In the "Contact" project, the code used to modify a record in the CEDB database ("Contacts") is found in the "Validate" button of the "WIN\_ContactForm" window.

► To view this code:

1. Open the "WIN\_ContactForm" window.
2. Select the "Validate" button.
3. Display the code of the button ([F2]).

```
// Retrieve the modified data
cdbContact.Name = EDT_Name
cdbContact.FirstName = EDT_FirstName
cdbContact.CivilStatus = COMBO_Title..DisplayedValue
...
cdbModify("", cdbContact, gnRecNum)
```



Note

#### Custom CEDB databases

To modify a record in a custom CEDB database, you must :

- point to the record to modify (***cdbRead***, ***cdbReadSeek***, and so on).
- specify the values of the items to modify (***cdbWriteCol***).
- save the record in the database (***cdbModify***).

## Deleting a record

To delete a record from a standard CEDB database, you must:

- point to the record to delete (***cdbRead***, ***cdbReadSeek***, and so on).
- delete the record from the database (***cdbDelete***).

The number of the record to delete can also be directly specified in ***cdbDelete***.

In the "Contact" project, the code used to delete a record from the CEDB database ("Contacts") is found in the "Delete" button of the "WIN\_ContactList" window.

► To view this code:

1. Open the "WIN\_ContactList" window.
2. Select the "Viewing" pane.
3. Select the "Delete" button.
4. Display the code of the button ([F2]).

```
// Delete the contact from the database
cdbDelete("", cdbContact, RecNum)
```



Note

### Custom CEDB databases

The method for deleting a record in a custom CEDB database is identical to the method for deleting a record in a standard CEDB database.

## Browsing a file

When a data file contains several records, it may be useful to browse it (to display the data in a table or to process the records for instance).

The following functions are used to browse a CEDB data file:

- ***cdbReadFirst***: Points to the first record found in a file and reads this record
- ***cdbReadNext***: Points to the next record found in a file and reads this record.
- ***cdbReadLast***: Points to the last record found in a file and reads this record.
- ***cdbReadPrevious***: Points to the previous record found in a file and reads this record.

***cdbOut*** enables you to find out whether the end of file has been reached.

For instance, the following code is used to browse the records found in the "Contacts" CEDB database (from the first one to the last one):

```
// Read the first record
cdbReadFirst("", cdbContact)
// Record outside the file?
WHILE cdbOut("", cdbContact) = False
  // Process the record
  ...
  cdbReadNext("", cdbContact)
END
```



Note

#### Custom CEDB databases

The method for browsing a file in a custom CEDB database is identical to the method for browsing a file in a standard CEDB database.

## Searching for a record

Searching for a record enables you to access a specific record without having to browse the entire file. All you have to do is specify the search criteria.

`cdbReadSeek` is used to seek a record according to a specified value.

In the "Contact" project, the code used to access a record in the CEDB database ("Contacts") is found in the "Find" button of the "WIN\_ContactList" window ("Find" tab).

► To view this code:

1. Open the "WIN\_ContactList" window.
2. Select the "Find" pane.
3. Select the "Find" button.
4. Display the code of the button ([F2]).

```
// Search for the contact whose name starts with
// the characters specified in the "EDT_Name" control
cdbReadSeek("", cdbContact, "Name", EDT_Name, cdbStartsWith)
```



Note

#### Custom CEDB databases

The method for seeking a record in a custom CEDB database is identical to the method for seeking a record in a standard CEDB database.

## Viewing the records

Several methods can be used to view the records found in a CEDB database. Let's now see how the records can be viewed:

- one by one: Each item found in the record is displayed in a different control (Form mode).
- all of them: All the records are displayed one after the other in a table (Table mode).

### Viewing the record items (Form mode)

To view the items of a record found in a standard CEDB database, you must:

- read the record to view (`cdbRead`).
- read the value of each variable found in the structure of the database used.
- assign this value to a control.

In the "Contact" project, the code used to view a record is found in the "Global declaration" process of the "WIN\_ContactForm" window.

► To view this code:

1. Open the "WIN\_ContactForm" window.
2. Display the code of the window ([F2]).

```
// Read the contact selected in the table
cdbRead("", cdbContact, gnRecNum)
// Read and assign the value of the different variables
// of the structure
EDT_Name = cdbContact.Name
EDT_FirstName = cdbContact.FirstName
...
```



Note

#### Custom CEDB databases

To view a record in a custom CEDB database, you must:

- read the record to view (***cdbRead***).
- read the value of each item (***cdbCol***).
- assign this value to a control.

### Viewing all the records (Table mode)

To view all the records found in a standard CEDB database, you must:

- browse all the records found in the database (***cdbReadFirst***, ***cdbReadNext***, ***cdbReadPrevious***).
- add each record found to a memory table (***TableAddLine***).

The structure variables are used to find out the value of each record item.

In the "Contact" project, the code used to view all the records is found in the "Initialization" process of the "WIN\_ContactList" window.

► To view this code:

1. Open the "WIN\_ContactList" window.
2. Display the code of the window ([F2]).

```
// Read the first contact
cdbReadFirst("", cdbContact)

// Record outside the file?
WHILE cdbOut("", cdbContact) = False
  // Add the record to the table
  TableAddLine(TABLE_Contact, cdbContact.Name+ " " + ...
    cdbContact.FirstName, cdbContact.HomePhone, ...
    cdbContact.Email)
  // Read the next record
  cdbReadNext("", cdbContact)
END
```



Note

**Custom CEDB databases**

To view all the records found in a custom CEDB database, you must:

- browse all the records found in the database (***cdbReadFirst***, ***cdbReadNext***, ***cdbReadPrevious***).
- add each record found to a table (***TableAddLine***).  
***cdbCol*** is used to find out the value of each item for a record.

DEVELOP 10 TIMES FASTER

**PART 4**

**Specific Features of Pocket PC**

**XIII**  
**Express**



# LESSON 4.1. SPECIFIC FORMATS

**This lesson will teach you the following concepts...**

---

- Managing the character strings (UNICODE or ANSI).
- Managing the directories in Windows for Pocket PC.
- Specific features in Windows for Pocket PC.



Estimated time: 20 min

## Managing the character strings

The default format of character strings on PC differs from the default format of character strings on Pocket PC.

On **PCs**, the Windows applications handle character strings in **ANSI** format. On **Pocket PCs**, the Windows applications for Pocket PC handle character strings in **UNICODE** format.

### What is the UNICODE format?

The UNICODE format is used to represent a very large set of characters by representing each letter on 2 bytes. This format can encode 65 536 characters. All the characters found in the 24 most common character sets can be represented in a single set. Each character has a unique identifier. Therefore, characters coming from different character sets can be used at the same time.

### What is the ANSI format?

The ANSI format represents each character on one byte. This format can encode 256 characters in the Indo-European character sets. This format can represent all the character sets. However, a single character set can be used at a time.

### Using character strings in UNICODE format in Pocket PC

In most cases, the character strings are handled in the same way by the WLanguage functions in WinDev Mobile and in standard WinDev. WinDev Mobile automatically supports the different formats of character strings in a way that is completely transparent for the developer and for the user.



Test

When running the test of a WinDev Mobile application that performs simple operations on character strings, WinDev Mobile automatically supports the ANSI or UNICODE format.

For instance, the following code returns the same result in test mode and at run time.

```
MyString is string = "WinDev is fantastic"
Info(Length(MyString)) // Returns 19
```

### AnsiToUnicode and UnicodeToAnsi

*AnsiToUnicode* and *UnicodeToAnsi* are used to convert ANSI to UNICODE (and conversely).

These functions are useful when handling character strings between two computers that use different formats of character strings or when handling external files.

### The "Buffer" type

In standard WinDev, a character string variable can contain characters as well as binary data (an image for instance).

In WinDev Mobile, if a character string variable contains binary data, this data may be wrong (faulty conversion for instance). To handle binary data, we recommend that you use a buffer variable.

Furthermore, the buffer type enables you to use the same source code in a WinDev Mobile application and in a standard WinDev application.

### Handling external files

Depending on the format of the external file used, some conversions are required:

- before writing a character string into an external file:

	Format of the string to write	External file in ANSI format ( <i>fOpen</i> )	External file in UNICODE format ( <i>fOpen</i> associated with the <i>foUnicode</i> constant)
<i>fWrite</i>	ANSI	No conversion is required	Converting the string before the write operation ( <i>AnsiToUnicode</i> )
	UNICODE	Converting the string before the write operation ( <i>UnicodeToAnsi</i> )	No conversion is required
<i>fWriteLine</i>	ANSI	No conversion is required	String automatically converted before the write operation
	UNICODE	String automatically converted before the write operation	No conversion is required

- after reading a character string in an external file:

	Current computer	External file in ANSI format ( <i>fOpen</i> )	External file in UNICODE format ( <i>fOpen</i> associated with the <i>foUnicode</i> constant)
<i>fRead</i>	PC running Windows	Read operation in ANSI format	Read operation in UNICODE format
	Pocket PC	Conversion to UNICODE possible with <i>AnsiToUnicode</i>	Conversion to ANSI possible with <i>UnicodeToAnsi</i>
<i>fReadLine</i>	PC running Windows	Read operation in ANSI format Conversion to UNICODE possible with <i>AnsiToUnicode</i>	
	Pocket PC	Read operation in UNICODE format Conversion to ANSI possible with <i>UnicodeToAnsi</i>	

## Transmission between two computers that use different formats of character strings

Some conversions are required to perform a transmission between two computers that use different formats of character strings:

Format of character strings on the current computer	Write operation ( <i>sWrite</i> or <i>SocketWrite</i> )	Read operation ( <i>sRead</i> or <i>SocketRead</i> )	
		Buffer containing a string in ANSI format / Message in ANSI format	Buffer containing a string in UNICODE format / Message in UNICODE format
ANSI (PC running Windows XP for instance)	The character string / the message will be in ANSI format	No conversion is required	Conversion required ( <i>UnicodeToAnsi</i> )
UNICODE (Pocket PC running Windows CE for instance)	The character string / the message will be in UNICODE format	Conversion required ( <i>AnsiToUnicode</i> )	No conversion is required

## Managing the directories in Windows for Pocket PC

The management of directories on PC differs from the management of directories on Pocket PC.

On a **PC**, the paths have the following format: "C:\MyDocuments\MyFile.txt". On **Pocket PCs**, the paths have the following format: "\MyDocuments\MyFile.txt". There is a single tree structure and the notion of drive does not exist.

### Handling a file by programming

When a file is handled by programming (*TreeAdd*, *fCreate*, *dSaveImageBMP*, *HSubstDir*, and so on), one of the following formats must be used:

- "\<FileName>.<Extension>" (if the file is found at the root of the Pocket PC)
- "\<DirectoryName>\<FileName>.<Extension>"

For example:

```
fOpen("\Poll.txt")
// the file is found at the root of the Pocket PC

TreeInsert(TVRecipe, "Recipes" + TAB + "Desserts", ...
    "Cakes", "\MyImages\CollapsedImg.BMP", ...
    "\MyImages\ExpandedImg.BMP")
// the files are found in the "MyImages" directory

dSaveImageGIF(DrawingImage, fExeDir()+"\Image.GIF")
// the file is found in the directory
// of application execution
```

GO

Test

When running the test (in simulation mode) of a WinDev Mobile application that handles files by programming, the files and the directories used are those found on the development computer (and not the ones found on the Pocket PC).

To use the same code in test mode and at run time, you can group all the relevant files in the directory of the application and use **fExeDir** to build the path of the files to handle.

## Current directory

The notion of current directory does not exist in Windows for Pocket PC (limit of the operating system). This is the reason why the functions that handle the current directory (**fCurrentDir** for instance) are not available in WinDev Mobile.

## File picker

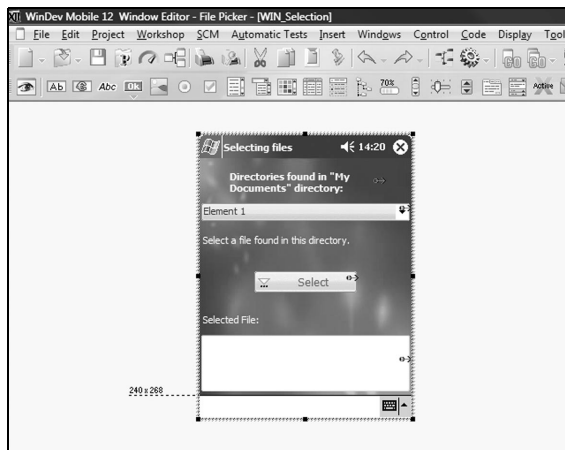
In Pocket PC, the file picker is used to select the files found:

- in the "\\My Documents" directory or in one of its immediate sub-directories.
- in a directory found at the root of a storage card.

Therefore, **fSelect** can only be used to select a file found in one of these directories.

We shall now use the file picker:

- ▶ Open the "File Picker.WPP" project found in the "Tutorial\Answers\File Picker" sub-directory of the setup directory of WinDev Mobile.  
Note: To open this project directly, select "? .. Tutorial .. File picker".
- ▶ Open the "WIN\_Selection.WPW" window ("File .. Open").  
The following window is displayed:



This window is used to select a file from the "My Documents" directory or from one of its immediate directories.


### Test in simulation mode

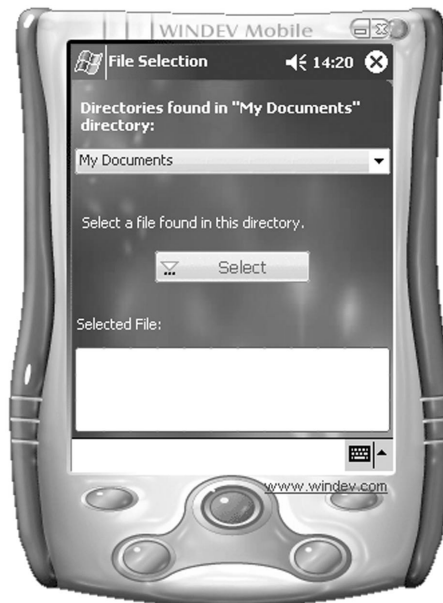
**GO**

Test

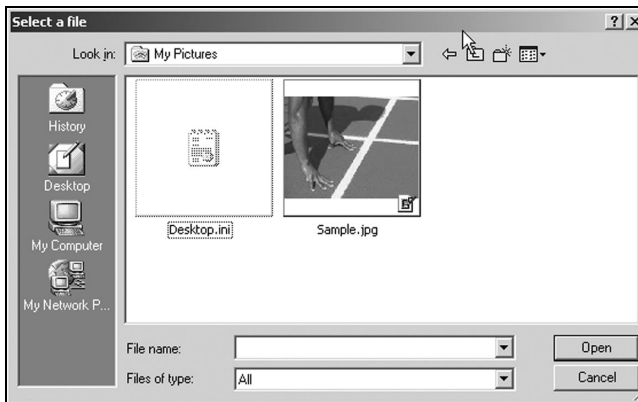
In test mode (simulation on the development computer), when calling **fSelect**, the Windows file picker for PC is displayed and it returns a path in PC format.

We shall run the test of the "WIN\_Selection" window in simulation mode on the development computer:

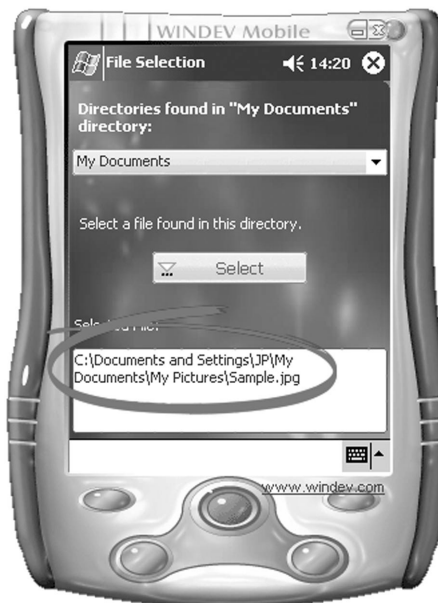
- ▶ Click the "GO" icon  (or press [F9]). The test of the window is run. The window is displayed in a simulator.



- ▶ Select one of the directories found in "My Documents" (from the combo box) and click "Select". The file picker for PC is displayed and it returns the list of files found on the current computer:



- ▶ If you select a file, the path of this file will be in PC format:



- ▶ Close this window. Let's now run the test of the window on the Pocket PC.

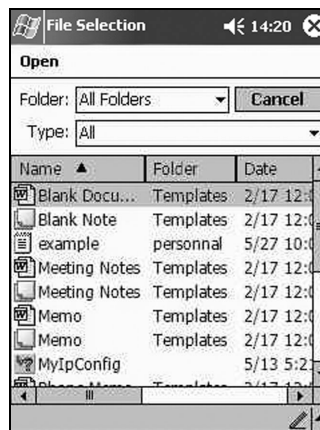
### Test on Pocket PC

Let's now run the test of the window on the Pocket PC. To run this test, a Pocket PC must be currently connected to the current computer.

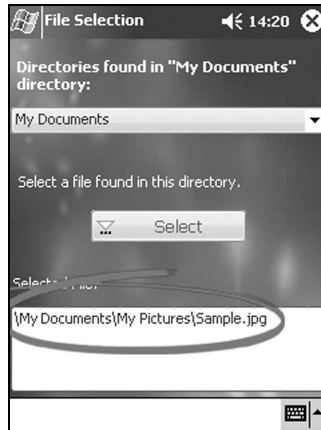
- ▶ Click "Run the project test (pocket)" (or select "Project .. Test mode .. Run the project test (pocket)"). The window is automatically opened on the Pocket PC:



- ▶ Select one of the directories found in "My Documents" (from the combo box) and click "Select a file found in this directory". The file picker for Pocket PC is displayed and it returns the list of files found on the Pocket PC:



- ▶ If you select a file, the path of this file will be in Pocket PC format:



Note

If the "My documents" directory is directly selected from the "WIN\_Selection" window, the file picker of the Pocket PC returns the list of all the files found in "My documents" and in its immediate sub-directories. In this case, the "Folder" option of the picker corresponds to "All the folders".

- ▶ Close this window.

### File explorer

The file explorer of Windows for Pocket PC is not as limited as the file picker. Indeed, the file explorer is used to browse all the directories found on the Pocket PC.

### Storage card

Many Pocket PCs are equipped with one or more storage cards. These cards are used to increase the storage capacity of the Pocket PC.

You have the ability to install a full WinDev Mobile application on a storage card and/or to handle the files found on a storage card.

To handle a file found on a storage card by programming, use the following format: "<code>\"<CardName>\<DirectoryName>\<FileName>.<Extension>\"</code>".

For example:

```
fOpen (" \\MyCard\\MyFiles\\Advertising.txt ")
```



Note

When a hard reset is performed on the Pocket PC, loss of data occur in the main storage card.

On a Smartphone, only the data found in the "Storage" directory (and in its sub-directories) and in the storage card are kept once the phone is switched off. See 'Application on Smartphone' (page 69) for more details.

## Memory space and speed of Pocket PC

---

The available memory space as well as the runtime speed of applications on a Pocket PC are a greatly restricted compared to a standard PC.

Don't forget this fact when you create a WinDev Mobile application!

Don't overload your application with useless files, limit the number of windows, clear the data files, and so on.

## Platform where the project is run

---

To find out the platform where the WinDev Mobile application is run, use `SysWindowsVersion` associated with the `SysVersionPlatform` constant.

## LESSON 4.2. INTERACTIONS BETWEEN APPLICATIONS

**This lesson will teach you the following concepts...**

- Sharing data between a WinDev Mobile application and a standard WinDev application
- Accessing the Pocket PC from a standard WinDev application.



Estimated time: 15 min

## Sharing data between two applications

A WinDev Mobile application can share data with a standard WinDev application.

You have the ability to use:

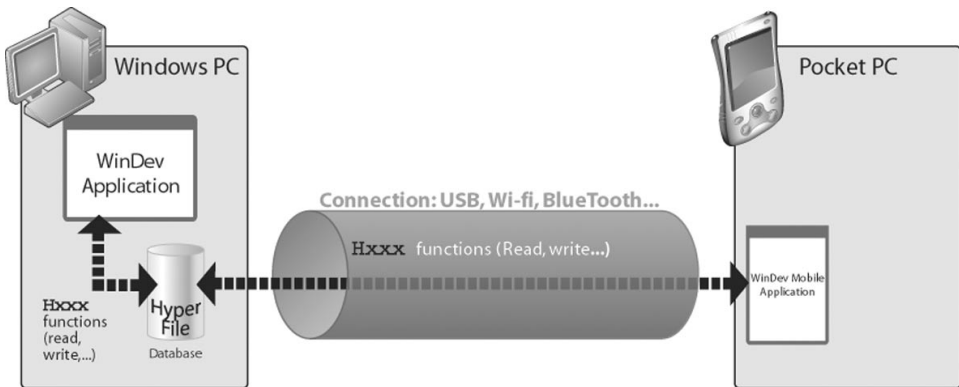
- a standard WinDev application that handles the entire database.
- a WinDev Mobile application that handles the entire database or part of this database.

When two applications share the same data, the data files can be managed in two different ways:

- **Handling the same data files:**

The two applications handle the same data files. These data files are found on the PC. The WinDev Mobile application accesses the data files via Wi-Fi, via infrared, via GPRS, and so on.

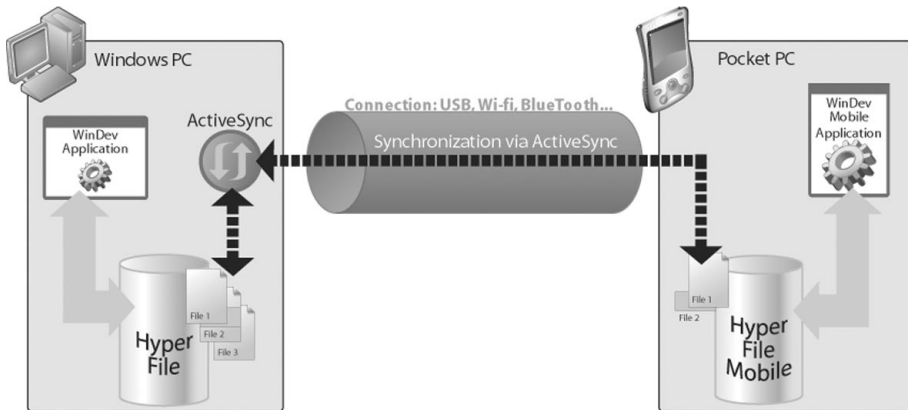
**HSubstDir** is used to specify the data directory to use.



For example: application for taking orders in a restaurant. The new orders are automatically sent to the database found on the PC.

- **Copying the data files to the Pocket PC:**

All the data files (or some of them) are previously copied to each Pocket PC. Each application handles its own files. To take into account the modifications made to each application, the data files must be synchronized (automatically or not).



For example: application for taking street polls. The answers will be available in the standard WinDev application once the data files have been synchronized.

### Handling the same data files

To allow the WinDev Mobile application to access the data files found on the PC:

- the Pocket PCs must be equipped with a network access (Ethernet, Wi-Fi, and so on).
- the data found on the PC must be accessible in read/write via a UNC path (the directory used must be a shared directory).

The data can then be handled (addition, modification and deletion) by the Hyper File functions.



EXAMPLE

Example

#### Databases in Hyper File format (on the Pocket PC and on the PC)

The "Network Tasks" example (supplied with WinDev Mobile) contains a project that can be run on Pocket PC and a project that can be run on PC. These two examples use the data files found on the PC.

## Copying the data files to the Pocket PC

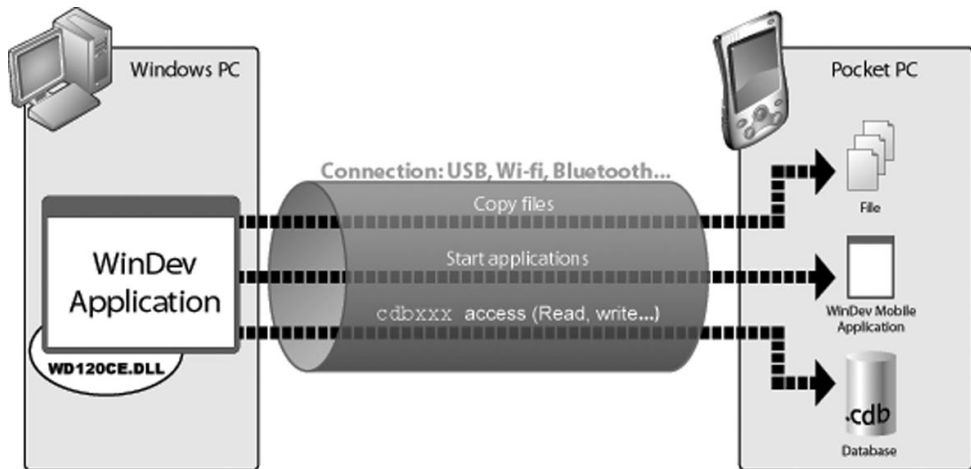
To update the data files found on the PC with the data entered on the Pocket PCs, all you have to do is synchronize the files.

If the data files are in Hyper File format, all you have to do is connect the Pocket PCs to the PC. The automatic Hyper File synchronization via ActiveSync does it all. See 'Synchronizing the data files' (page 96) for more details.

If the data files are not in Hyper File format, you must program the synchronization between the WinDev Mobile application and the standard WinDev application. See the examples supplied with WinDev Mobile for more details.

## Accessing the Pocket PC

The functions for accessing the Pockets PCs (starting with "ce") are used to access the Pocket PCs from a standard WinDev application.



These functions can be used in a standard WinDev application when a Pocket PC is connected to the current computer.

These functions are mainly used to:

- handle the files found on the Pocket PC (copy the files, find out the size of a file, return the list of files found in a directory, and so on).
- retrieve information about the Pocket PC (find out the charge level of the batteries, the type of processor, the Windows version used, and so on).
- manage the registry of Pocket PC (create or delete a key, modify the value of a key, check the existence of a key, and so on).

## LESSON 4.3. SHARING WINDEV ELEMENTS

**This lesson will teach you the following concepts...**

---

- Importing standard WinDev windows into a WinDev Mobile project.
- Sharing source code.



Estimated time: 10 min



EXAMPLE

The "Pocket PC Import .WPP" project corresponds to the full project with the answers to this lesson. This project is found in "\\Tutorial\\Answers\\Pocket PC Import". To open this project, select "? .. Tutorial .. Pocket PC Import (answer)".

You can follow this lesson without opening the project.

## Importing a standard WinDev window

The same WinDev elements (analysis, reports, queries, and so on) can be used in a standard WinDev application and in a WinDev Mobile application.

However, the windows do not have the same format in standard WinDev (".WDW" file) and in WinDev Mobile (".WPW" file). Indeed, the windows do not have the same size, the same characteristics, the same features, the same user approach, and so on.

WinDev Mobile enables you to import a standard WinDev window into a WinDev Mobile project.

### How do I import a WinDev window?

We are going to import the following WinDev windows:



► To import these windows into a WinDev Mobile project:

1. Open the "Pocket PC Import.WPP" found in the "Tutorial\Exercices\Pocket PC Import" sub-directory of the setup directory of WinDev Mobile.

Note: To open this project directly, select "? .. Tutorial .. Pocket PC Import (exercise)".

2. Select "File .. Import .. WinDev elements". The import wizard starts.

3. Select the windows named "WIN\_Form\_Customer.WDW" and "WIN\_Simple\_Form\_Customer.WDW" (found in the "Tutorial\Exercices\PC Import" sub-directory).

4. Validate the import wizard.

The screen of a Pocket PC being smaller than the screen of a PC, the imported windows as well as their controls must be resized.

5. Resize the window as well as their controls.

6. Save the windows ("File .. Save" or [Ctrl]+[S]).

WinDev Mobile proposes to add these new elements to the current project. Accept this addition.

These windows are automatically opened in the editor. These windows correspond to the fol-

lowing files: "WIN\_Form\_Customer.WPW" and "WIN\_Simple\_Form\_Customer.WPW".



## Operations performed during the import

When a window is imported:

- the controls that do not exist in WinDev Mobile are automatically deleted (toolbar controls, HTML controls, and so on).
- the list of errors that occurred (functions that do not exist in WinDev Mobile for instance) is displayed in the "Code" pane.

## Sharing source code

WinDev Mobile proposes several methods for sharing source code:

- the input of multi-product code.
- *InPocketMode*.

## Input of multi-product code

The code editor is used to enter the equivalent of the source code that will be run in WinDev Mobile, in standard WinDev or in WebDev. This code is entered at the same location in the code editor. Tabs are used to select the platform corresponding to the current code.

The corresponding code will be automatically run according to the runtime platform.

The same feature is also available for WinDev and for WebDev. Therefore, creating multi-product components is child's play.

## InPocketMode

**InPocketMode** allows you to share source code between a WinDev Mobile application and a standard WinDev application.

When compiling, the functions that cannot be used in Pocket PC will be displayed in the "Code" pane.

**InPocketMode** will be used at runtime to prevent from running these functions so that no error will occur.

For example, the following code is shared between a WinDev Mobile application and a standard WinDev application.

```
MyParameterFile is string
// Code run from the WinDev Mobile application?
IF InPocketMode() = True THEN
  // WinDev Mobile application
  MyParameterFile = "\My Files\param.INI"
ELSE
  // Standard WinDev application
  MyParameterFile = fCurrentDir() + "\Param.INI"
END
```

When compiling, an error will be displayed in the "Code" pane to indicate that **fCurrentDir** is not allowed in Pocket PC. However, no runtime error will occur when the application is used on a Pocket PC: indeed, **fCurrentDir** will never be called.



Note

**InPocketMode** is used to share the sets of procedures, the classes, the components, ... between a WinDev Mobile application and a standard WinDev application.



Test

In test mode, **InPocketMode** always returns True.

**InPocketMode** returns False only when it is called from a standard WinDev application.

## LESSON 4.4. INPUT MODE

**This lesson will teach you the following concepts...**

---

- The different input modes available for a Pocket PC and for a Smartphone.



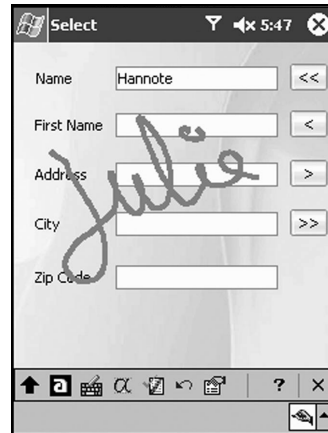
Estimated time: 10 min

## Entering information on a Pocket PC

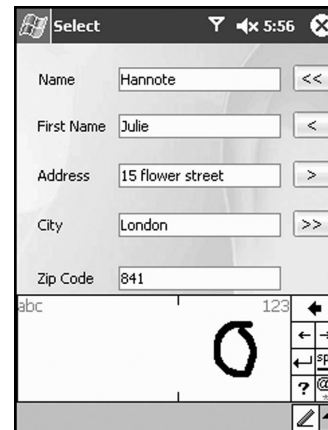
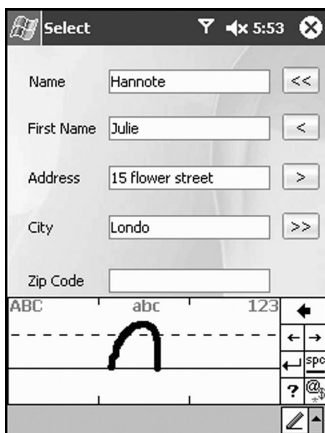
To allow the users of your applications to enter data, the keyboard of the Pocket PC (also called SIP for Software Input Panel) must be managed.

This keyboard allows you to:

- display a miniature keyboard at the bottom of the screen. The user clicks this keyboard with the stylus to enter information.  
For example:
- automatically recognize the different words written on the screen with the stylus (method called "Transcriber").  
For example:




- automatically recognize the letters entered in a specific character set (method called "Letter Recognizer").  
For example:
- automatically recognize the blocks of words entered in a specific character set (method called "Block Recognizer"). For example:



WinDev Mobile enables you to easily manage this keyboard by programming (*SIPList*, *SIPMode* and *SIPVisible*).



**Note** The icon representing the keyboard of the Pocket PC (  icon) enables you to manage the use of the keyboard.



**Note** "Activate the keyboard in edit" ("Details" tab in the description window of an edit control) enables you to automatically make the current keyboard visible when the control is in edit.

## Entering information on a Smartphone

Several default input modes are available on Smartphone:

- the "abc" and "ABC" modes: to enter lowercase and uppercase characters.  
For example, in this mode, "5" must be pressed twice to write the letter "k".
- the "123" mode: to enter digits.
- the "T9" mode (intuitive edit patented by AOL): this test mode is used to avoid pressing the same key several times for a letter.  
For example, press "2665687" to automatically write "hello".  
To choose the different words corresponding to the same sequence of keys, press "0" to display the available suggestions.  
For example, "26663" can be used to write "as" and "good".

To toggle the different input modes, press the "\*" key (long pressures). Short pressures are used to toggle the uppercase/lowercase characters.

The input mode currently used is displayed in the top right corner of the Smartphone:



• icon  for the "abc" mode.



• icon  for the "ABC" mode.



• icon  for the "123" mode.



• icon  for the "T9" mode.



DEVELOP 10 TIMES FASTER

**PART 5**

**Communication**

**XIII**  
**Express**



# LESSON 5.1. INTRODUCTION

**This lesson will teach you the following concepts...**

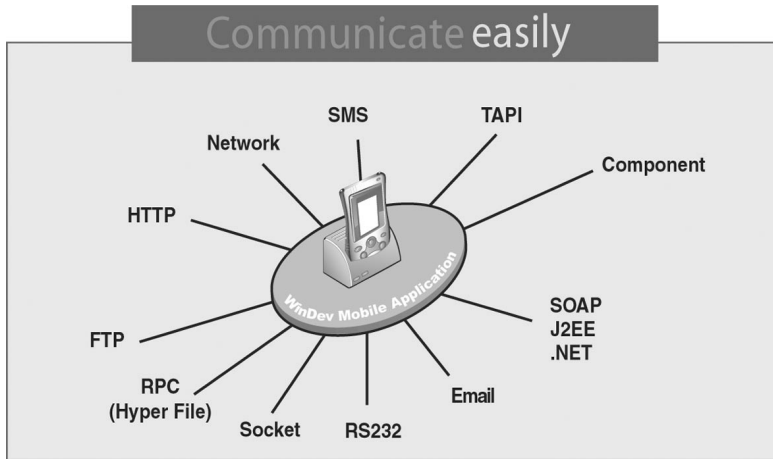
---

- The communication with WinDev Mobile.



Estimated time: 10 min

## Communicate with WinDev Mobile



The communication tools offered by WinDev Mobile allow for total openness!

RPC, TAPI, network, socket, FTP, Email, HTTP, SOAP, J2EE, .NET, ... all these standards are supported by WinDev Mobile.

WinDev Mobile can be used to establish communication between two Pocket PCs, a Pocket PC and a Smartphone, a Pocket PC and a PC, and so on.

These "dialogs" are performed via infrared, via Wi-Fi, via network card, and so on. These "dialogs" are entirely transparent.

We shall not present all the features offered by WinDev Mobile. We shall take a look at email management (see the lesson page 136).

For the other communication modes, we shall briefly explain the main differences compared to standard WinDev.

### Transferring files by FTP

FTP (File Transfer Protocol) is a protocol used to transfer files from one site to another remote site. This protocol is used to exchange files via TCP/IP, Internet, Wi-Fi or ActiveSync.

Several thousand file servers can be accessed by FTP on Internet. These servers propose shareware or freeware to the public.

Several WLanguage functions allow you to manage the files found on an FTP server from your applications (*FTPxxx* functions).



EXAMPLE

Example

The "Pocket FTPClient" example, supplied with WinDev Mobile, is used to manage an FTP server.



Note

#### Differences compared to standard WinDev

In a WinDev Mobile application, the files can be transferred via Wi-Fi, via GPRS and via ActiveSync.

## Remote access (RPC on Hyper File Mobile)

The remote access enables you to consult a Hyper File Mobile database via Internet/Intranet or via STN (Switched Telephone Network).

To reduce network traffic, the functions that do not require database access are performed locally. The Hyper File library (WP120HF.DLL) and the analysis description (".WDD" file) must be found on each Pocket PC.

Data will be exchanged via the RPC protocol (Remote Protocol Call) by using the functions of WD120COM.DLL (found on the server) and WP120COM.DLL (found on the client computer).

Data can be exchanged:

- via Wi-Fi.
- via ActiveSync.
- via GPRS.
- via network card.



Note

#### Differences compared to standard WinDev

In a WinDev Mobile application, the data is exchanged via Wi-Fi, via ActiveSync or via GPRS.

## Managing sockets

Several WLanguage functions allow you to manage the sockets (**SocketXXX** functions).

A socket is a communication resource used by the applications to communicate between computers regardless of the network type.

This communication mode can be used, for instance, to communicate between computers connected via Internet.

WinDev Mobile allows to create a socket that uses the infrared port (**SocketCreateInfrared** and **SocketConnectInfrared**).

Data can be exchanged between two computers:

- via Wi-Fi.
- via Infrared.
- via ActiveSync.
- via GPRS.
- via network card.



Note

#### Differences compared to standard WinDev

In a WinDev Mobile application, the data is exchanged via Wi-Fi, via Infrared, via ActiveSync or via GPRS.



Caution!

#### Exchanging messages

When exchanging messages between a Pocket PC and a PC, don't forget to check the format of the character strings (ANSI or UNICODE). Depending on the current computer and on the format used, some conversions will be required (**AnsiToUnicode** and **UnicodeToAnsi**).

See 'Managing character strings' (page 108) for more details.



EXAMPLE

Example

The "Using sockets" example, supplied with WinDev Mobile, contains a project that can be used in Pocket PC and a project that can be used on PC. These two examples present the use of the functions for socket management.

## Web services (SOAP, J2EE, .NET)

The Microsoft .NET and Sun J2EE server platforms allow you to export their components as XML Web services.

An XML Web service is defined as an application accessible via the standard Internet protocols. More specifically, the Web services allow several computers connected via Internet to interact between themselves.

The Web services enable you to run procedures and processes on a remote Web server (.Net or J2EE) from a Pocket PC.



Note

#### Differences compared to standard WinDev

In a WinDev Mobile application, the procedures and the processes are run via Wi-Fi, GPRS or ActiveSync.



Caution!

**Passing parameters**

When passing parameters to a procedure run, if the value of the parameter is a character string, don't forget to check its format (ANSI or UNICODE). Indeed, this string is automatically converted into ANSI when it is sent. To prevent this string from being automatically converted, use one of the following variables: SOAP.ExtendedValue, J2EE.ExtendedValue or DotNet.ExtendedValue.

**SMS**

WinDev Mobile enables you to easily send and read SMSs (Short Message Service) via the SMS functions of WLanguage.

An SMS corresponds to a text message (up to 160 characters) sent on a cell phone.

See 'Application on Smartphone' (page 69) for more details.

**Summary**

The table below presents the communication modes available for each feature proposed by WinDev Mobile.

	Remote access (RPC on Hyper File)	Email	FTP	HTTP	Telephony	SOAP J2EE .NET	Socket	SMS
ActiveSync	X	X	X	X		X	X	
Network card	X	X	X	X		X	X	
GPRS	X	X	X	X		X	X	
Infrared							X	
Smartphone or phone access (GSM)					X			X
Wi-Fi	X	X	X	X		X	X	

## LESSON 5.2. MANAGING EMAILS

**This lesson will teach you the following concepts...**

---

- Sending and receiving emails.



Estimated time: 5 min

## Overview

Several WLanguage functions allow you to manage incoming and outgoing emails. Furthermore, WinDev Mobile enables you to find out all the characteristics of an email:

- sender
- recipients
- outgoing date
- subject
- message
- attachments
- and so on.

Two methods can be used to manage the emails in WinDev Mobile:

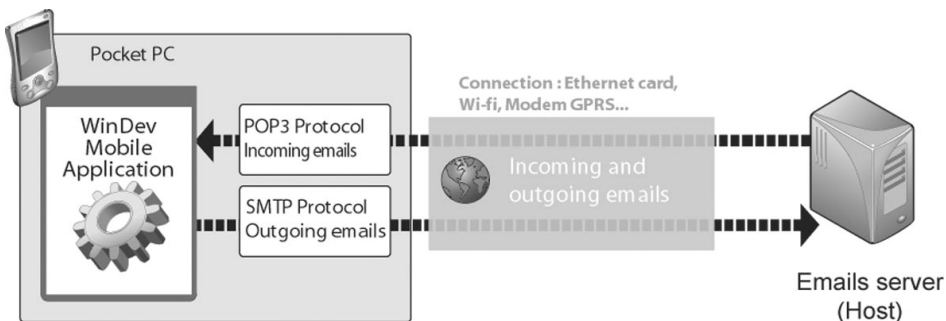
- **the POP3/SMTP protocol** (most common method): this protocol, used to manage emails, is recognized by all the service providers. This protocol enables you to directly communicate with the server, available at your ISP.
- **the "CEMAPI" API**: this management mode of emails uses Pocket Outlook to send and receive emails.

## Managing emails via the POP3/SMTP protocol

The POP3 and SMTP protocols are protocols for email management recognized by all the Internet Service Providers. These protocols allow you to directly communicate with the email server available at your ISP.

The POP3 protocol is used to receive emails.

The SMTP protocol is used to send emails.



Note

The method for sending and receiving emails via the POP3/SMTP protocol in WinDev Mobile is the same as the method for sending and receiving emails via the POP3/SMTP protocol in standard WinDev.

A connection must be established between the Pocket PC and a PC if no Ethernet card or GPRS modem is found on the Pocket PC.

## Managing emails via "CEMAPI"

CEMAPI is an API for email management used by most of the Pocket applications for sending and receiving emails (Pocket Outlook in most cases).

CEMAPI simplifies the management of the emails received by the host. When an email is read, it is automatically loaded in the local message box and deleted from the server (at the host).

All the characteristics required to manage the emails (POP3 protocol, SMTP protocol, remote access, and so on) are grouped in the "User Account".

### User account

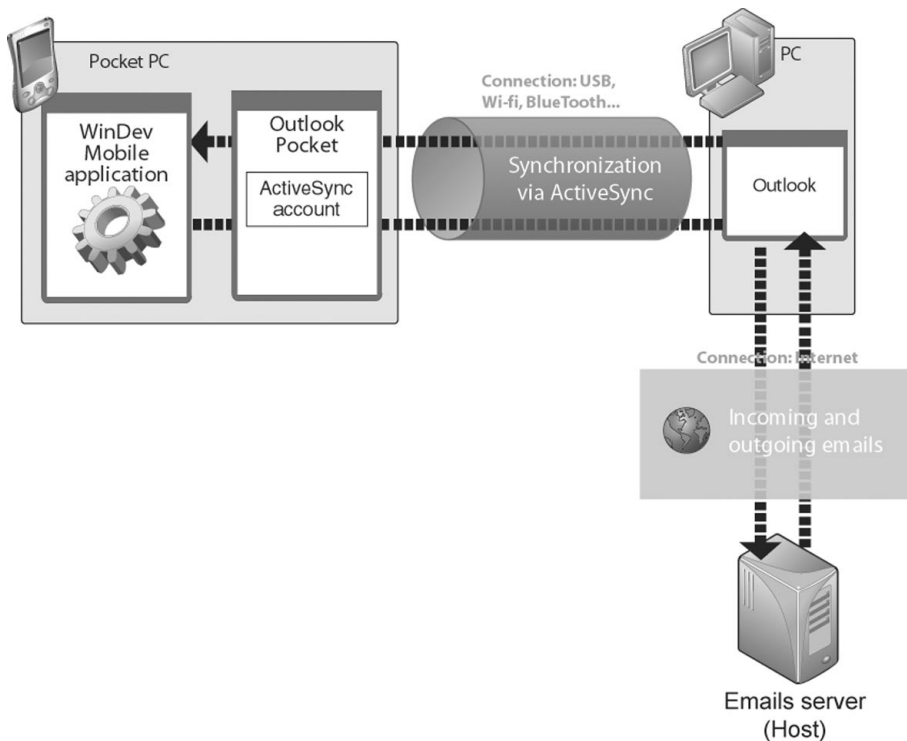
A user account defined in Pocket Outlook is required to manage the emails via CEMAPI.

By default, Pocket Outlook manages the "ActiveSync" user account.

Another user account can be used. It must be defined in this case.

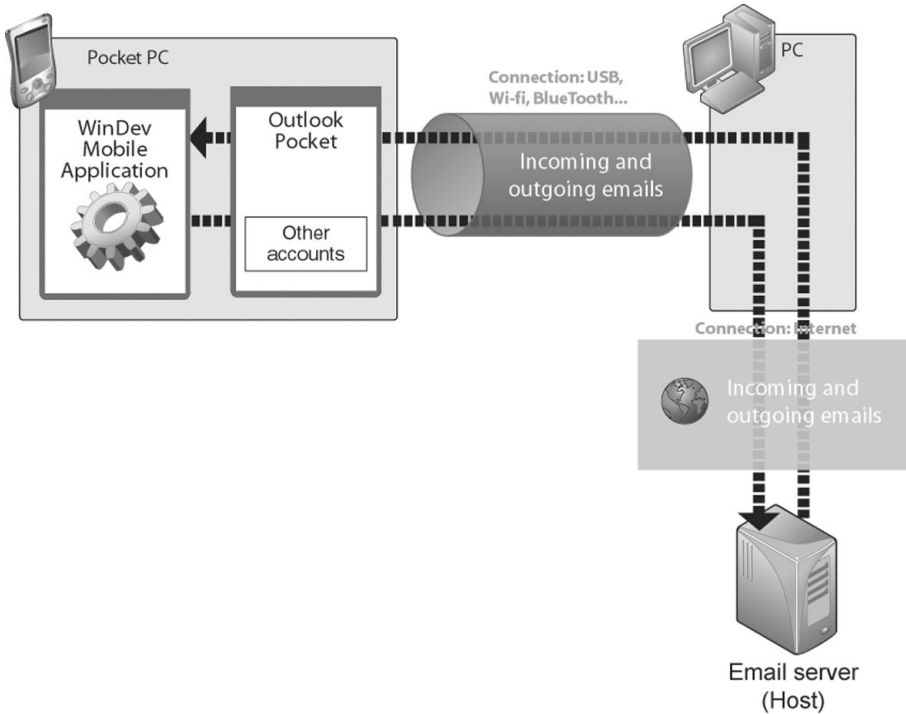
### "ActiveSync" user account

The "ActiveSync" user account is used by default.



**Another user account**

To use a user account other than "ActiveSync", all you have to do is describe a new user account.



Note

The method for sending and receiving emails via the CEMAPI protocol in WinDev Mobile is the same as the method for sending and receiving emails via the SIMPLE MAPI protocol in standard WinDev.

The only difference is:

- **"ActiveSync" user account:** in WinDev Mobile, the synchronization of emails must be configured by ActiveSync.
- **Other user account:** in WinDev Mobile, the emails must be synchronized by the email server.



DEVELOP 10 TIMES FASTER

**PART 6**

How do I proceed?

**XIII**  
*Express*



# QUESTIONS / ANSWERS

**This lesson will teach you the following concepts ...**

---

- Answers to your questions.



Estimated time: 1h

## Controls, windows

### Question How do I change the type of a window?

Two types of windows can be created in WinDev Mobile:

- Maximized window: A maximized window occupies the entire screen of Pocket PC.
- Non-maximized window: A non-maximized window can be resized by the user and occupy part of the screen only.

To change the type of a window:

1. Right-click the window and select "Description".
2. Display the "Details" tab.
3. Select the new type for the window.



Note

The "Style" tab can also be used to change the type of a window.

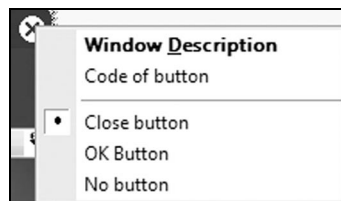
### Question How do I modify the type of the "OK/Close" button displayed in the title bar?

The "OK/Close" button found in the title bar of the WinDev Mobile windows is used to close or validate the current window.

By default, this button is used to automatically close the window.

The type of this button can be modified:

- via the popup menu of the button:



- in the "Style" tab of the window description.
- by programming (**WinSystemButton**).




Note

The "OK/Close" button is not available for the windows associated with the Smart-phone platform.


**Question** How do I display the keyboard on the Pocket PC?

To allow the users of your applications to enter data, the keyboard of the Pocket PC (also called SIP for Software Input Panel) must be managed.

To display the keyboard on the Pocket PC:

- select "Enable the keyboard in edit" ("Details" tab of the description window of an edit control). This option is used to automatically make the current keyboard visible when the control is in edit.
- by programming (*SIPVisible*).
- on the Pocket PC directly, by clicking .

**Question** How do I display all the drop-down menus of a window?

In edit, when the window is not wide enough to display all the drop-down menus, the icon  is displayed. This icon is used to display all the menus.

At run time, only the menus that can be contained in the width of the window are visible.

**Question** How do I duplicate a control found in a window by programming?

We may still be far from "cloning human beings" but WinDev Mobile already enables you to clone controls. *ControlClone* is used to duplicate a control found in a window or in a report by programming.

The control is duplicated with the same characteristics but with a different name.

**Question** How do I delete a control found in a window by programming?

*ControlDelete* is used to delete a control from a window or from a report by programming.

**Question** How do I manage the planes of a window?

The planes found in a window are used to arrange the controls in "layers" to avoid overcrowding the screens and to keep the number of project windows down.

To associate a control with a plane:

1. Right-click the control.
2. Select "Associate with a plane".
3. Choose the number of the plane to which the control must be associated.

Use the [Page Up] and [Page Down] keys to go from one plane to another one in the editor. The num-

ber of the current plane is displayed:

- in the status bar of the editor (bottom right)
- in the home window of the current window (top right).



Tip

To avoid duplicating the same control in a window, associate the control to "no plane". The control is then visible in all the planes.

Only the controls of the current plane and the controls that belong to no plane are visible in edit and at run time.

**Plane** can also be used to:

- find out and change the current plane in a window.
- find out and change the plane associated with a control.

The "Plane Mode" pane ("Display .. Toolbars .. Plane Mode") is used to edit all the captions of the controls found in all the window planes in order to view them and to modify them if necessary.

#### Question How do I make a button invisible?

A button can be made invisible by programming with the following syntax:

```
ButtonName.Visible = False
```

Set the value to "True" to make the control visible.

This syntax can also be applied to all types of controls and to groups of controls.

#### Question How do I modify the color of a static?

The color of the captions is defined in the control style ("Control .. Choose a style").

However, the color of a caption can be modified by programming. The syntax is as follows:

```
// Color the static in red
StaticName.Color = PastelRed

// Color the background of the static in green
StaticName.BrushColor = LightGreen

// Restore the initial color (the one of the style)
StaticName.Color = iDefaultColor
```



Note

This syntax applies to all the types of controls.

**RGB** is used to define a color from the values of the Red, Green and Blue components.

```
<ColorCode> = RGB (<red>, <green>, <blue>)
```

You also have the ability to modify the colors of the rows, columns or cells in a Table control. The syntax is as follows:

```
// Modify the color of a column
ColumnName..Color = <ColorCode>

// Modify the color of a row
TableName[RowSubscript]..Color = <ColorCode>

// Modify the color of a cell
ColumnName[RowSubscript]..Color = <ColorCode>
// or
// TableName[RowSubscript, ColumnSubscript]..Color = <ColorCode>
```

**HSL** is used to create a color from its hue, saturation and lightness.

```
<ColorCode> = HSL (<Hue>, <Saturation>, <Lightness>)
```

### Question How do I display the progress of a process?

**Gauge** uses the status bar of your window to display the progress of a process.

Mostly progress is displayed by a progress bar.

To do so, use a Progress Bar control in a window ("Insert .. Control .. Progress bar").

In the initialization code of the Progress Bar control:

1. Initialize the minimum value of the progress bar:

```
NameProgressBar..MinValue = MinimumValue
```

2. Initialize the maximum value of the progress bar:

```
NameProgressBar..MaxValue = MaximumValue
```

In the code of the requested process, increment the progress bar at each step of the process:

```
NameProgressBar ++
// or NameProgressBar = NameProgressBar + 1
```

### Question How do I link a window to an option of my main menu?

**Open** is used to associate a window with a menu option. Enter the following code line in the click code of your menu option:

```
Open (MYWINDOW)
```



Note

To associate a report with a menu option, use *iPrintReport*:  
*iPrintReport* (MYREPORT)

### Question How do I create a popup menu?

A popup menu can be added:

- at window level.
- at control level.

For a window:

1. Right-click the window and select "Description".
2. Click the "Details" tab. Click the arrow found beside "Popup Menu" and select "Create a new popup menu".

For a control:

1. Right-click the control and select "Description".
2. Click the "GUI" tab. Click the arrow found beside "Popup Menu" and select "Create a new popup menu".

*PopupMenu* enables you to find out or modify the popup menu of a control or window by programming.

### Question How do I pass parameters to a window?

The method for passing parameters to a window is similar to the method for passing parameters to a procedure.

In the declaration code of the global variables of the window, enter the following syntax of WLanguage code:

```
PROCEDURE WindowName (pNameParam1, pNameParam2, ...)
```

When the window is opened by *Open*, pass the parameters after the name of the window, for instance:

```
Open(WindowName, ValueParam1, ValueParam2, ...)
```

If a parameter is initialized when declaring the window, this parameter becomes optional:

```
// pNameParam2 is an optional parameter
PROCEDURE WindowName (pNameParam1, pNameParam2 = "Test")
```



Note

It is better to pass parameters to a window than to declare global variables in the project.

**Question****How do I change a check box into a radio button?**

"Control .. Switch .. Radio Button/Check Box" is used to swap the two types of controls.



Note

This operation can also be performed between a combo box and a combo box with table, or between a table and a loop.

**Question****How do I retrieve the parameters passed in command line to an executable?**

To do so, use **CommandLine** in the initialization code of the project.  
See the online help (keyword: "CommandLine") for more details.



Tip

You can simulate passing parameters in command line to your project in test mode. In the editor, select "Project .. Test mode .. Configure the test mode" and enter the parameters of the command line.

**Question****How do I group the controls in order to modify their properties by programming?**

Perform the following operation:

1. Select several controls with your mouse.
2. In the menu of the editor, select "Control .. Groups .. Associate the selection".
3. Enter the name of the group and validate.

The groups of controls can be used in the windows and in the reports.

The controls will be associated with this group. You can then modify the properties of the controls found in this group with the following syntax:

```
GroupName..<PropertyName> = Value
```



Caution!

Only the properties common to all the controls can be modified.

### Question How do I align the controls?

Several methods can be used to align controls:

- the rulers.
- the interface checker.
- the real-time interface checker.
- the alignment options.

To enable the rulers, press [CTRL] and [R] simultaneously. "Markers" (or "tabulation marks") can be defined on the ruler and moved (by clicking the ruler at the requested location). Then, when the controls are moved in the window (or in the report), they will be "magnetized" when they come near these markers.

The real-time interface checker is automatically enabled when creating or moving a control. Temporary rulers are used to align the selected control with the other controls found in the window.

To enable the interface checker, select "Control .. Alignment .. Interface checker". This wizard proposes tips to perform alignments in the window.

The alignment options can be accessed from the menu of the WinDev Mobile editor or from the icon bar.

From the icon bar of WinDev Mobile, click .

From the menu of WinDev Mobile, select "Control .. Alignment" and choose an action.

After practising a few minutes, you will soon realize what a good thing proper alignment is!

### Question How do I give the same size to the buttons?

- ▶ Select the button that will be used as reference for the size (height and width). Then, select the button that must be resized.
- ▶ Select "Same width, Same height" in the alignment tools ("Control .. Alignment").

### Question How do I add a background image to a window?

On the window:

1. Right-click and select "Description".
2. Select the "Image" tab.

You can then choose an image and configure the display mode.

## Environment


---

### Question How do I display or hide the panes?


Press [CTRL] and [W] simultaneously.

To display or to hide the pane anchored at the bottom of the screen, press [CTRL] and [Q] simultaneously.


### Question How do I view the element to which the current process belongs?

To view the element corresponding to the current process, click . The window containing the requested element is displayed and the control is automatically selected.

### Question How do I print the source code?

To print the current source code (displayed in the code editor), click  in the icon bar of the editor or select "File .. Print the documentation".

### Question How do I print the analysis documentation?

To print the analysis documentation from the data model editor, click  in the icon bar of the editor or select "File .. Print the documentation".


### Question How do I print the full documentation of my project?

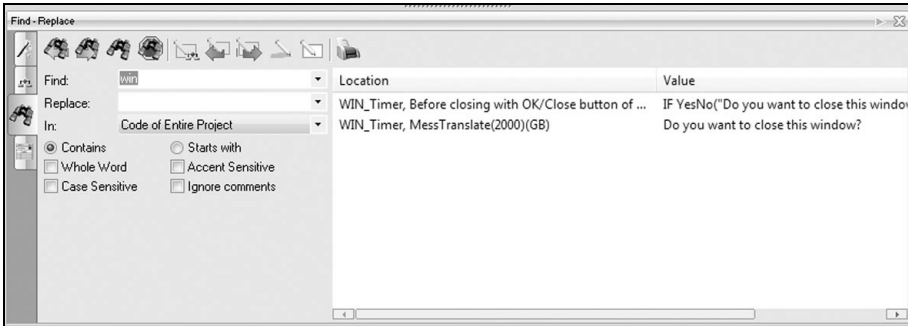
To print the full project documentation from the project graph (or from the dashboard), select "Project .. Print the project documentation".

### Question How do I create a skin template?

This operation contains several steps. See the online help (keyword: "Create, Skin template") for more details. But don't forget: several skin templates are supplied with WinDev Mobile!

### Question How do I find and/or replace a variable in the code?


To access the Find/Replace functions in the code, select "Edit .. Find" or "Edit .. Replace", or display the "Find - Replace" pane (  ) :



The search can be accessed at any time by pressing [CTRL]+[F].

### Question How do I manage a task list in my project?

The task list of a project can be displayed and refreshed in real time in the "Task List" pane of the editor.

- ▶ To view the task list, display the "Task List" pane (  ).
- ▶ To add a new task directly, right-click the task list and select "Create a new task".



Tip

In the code editor, if you enter a comment that starts with:  

```
// TO DO: write blablabla procedure
```

A task will be automatically added to the task list.

This task list is linked to the current project. When working in team, all the developers can see and refresh this list.

### Question How do I find out the list of project elements?

The elements found in a project are: windows, reports, queries, and so on.

- ▶ To view the project elements, select "Project .. List of project elements".

**Question** How do I view and change the tab order of the controls in a window?

The tab order of the controls is defined by the order in which the controls are created in the window. [F5] enables you to view the current order.

- ▶ To modify the tab order:
  1. Select "Windows .. Tab order .. Edit".
  2. In the window, modify the input order of the controls.

"Windows .. Tab order .. Define automatically" automatically defines the order of the controls according to their position and alignment in the window. The controls are browsed from the top left corner of the window.

**Question** How do I enable or disable the automatic data preview (Live Data)?

To enable or disable this feature, check or uncheck "Enable Live Data" in the "Live Data" tab of the project description ("Project .. Project Description").

**Question** How do I add a language to my project?

In the menu of WinDev Mobile, select "Project .. Project Description", then select the "Languages" tab. In the window, check the new languages that must be supported.

**Question** How do I modify the options of WinDev Mobile?

The environment settings of the editor (directory, language, login, and so on) can be modified from the options of WinDev Mobile ("Tools .. Options of WinDev Mobile").

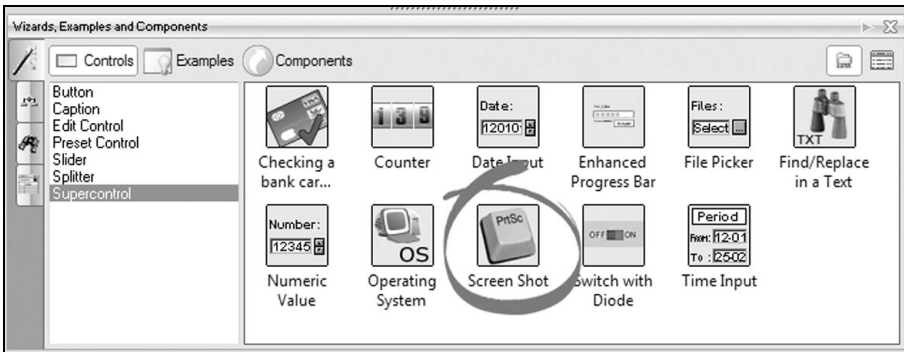
The parameters for displaying the windows or the reports (click, magnetism, and so on) can be modified from "Display .. Options .. Modify the options".

## Various

---

**Question** How do I perform a "screen shot"?

Include the "Screen Shot" supercontrol (available from the "Wizards, Examples and Components" pane) in your window. You don't even have to enter any source code!



To perform a screen shot of the Pocket PC (or Smartphone), use "WDCapture" ("Tools .. WDCapture - Screen shot").

### Question What are the processors on which the WinDev Mobile applications operate?

The applications generated by WinDev Mobile only operate on Pocket PC (equipped with ARM processors, ARM v4T processors and x86 processors). These applications do not operate on Windows PC.

### Question How do I read and write into a .INI file?

*INIRead* and *INIWrite* are used to read and to write in a INI file. See the online help (keywords: "IniRead" and "IniWrite") for more details.

### Question What are the image formats supported by WinDev Mobile?

WinDev Mobile supports the images in the following formats: BMP, JPEG, GIF or ICO.

### Question I want to compress data, can I do this with WinDev Mobile?

WinDev Mobile proposes several functions used to compress and to decompress data. The name of these functions starts with "Zip".

See the online help (keyword: "Zip") for more details.

**Question** How do I read and write in the registry?

**RegistryQueryValue** and **RegistrySetValue** are used to read and to write in the registry.

See the online help (keyword: "Registry, WLanguage functions") for more details.

**Caution!**

Don't forget to save the registry before you make any modification so that it can be restored if a problem occurs.

**Note**

The registry of Pocket PC can also be handled from a standard WinDev application (**ceRegistryXXX** functions).

**Question** How do I uninstall an application created with WinDev Mobile?

The provider and the name of the application must necessarily be specified when creating the setup program. This information is displayed in the panel for uninstalling programs on the Pocket PC.

To uninstall an application:

1. Click the "Start" menu.
2. Select "Parameters".
3. Display the "System" tab.
4. Select "Remove programs".
5. Select the application to delete and click "Uninstall".

**Question** How do I create an executable?

To create the executable of your project, select "Workshop .. Generate the executable".

**Question** How do I install an application?

Once the executable is generated ("Workshop .. Generate the executable"), the EXE directory found in the directory of your project contains all the elements required for your application to operate.

To prepare a setup of your application:

1. Select "Workshop .. Create the setup procedure". The wizard for setup creation starts.
2. Follow the instructions given on the screen.

### Question How do I associate an icon with my executable?

The icon associated with your executable can be defined when creating the executable. This icon must be in ICO format.



Note

A catalog of preset icons is supplied with WinDev Mobile. This catalog is accessible when selecting the icon.

### Question How do I detect the elements not used by my application?

After months or years of development and maintenance, the directory of your project often contains several files that are not used anymore but that you don't dare to delete.

Test files and windows, useless images, and so on. It's time for a clear out!

A tool of WinDev Mobile is used to automatically detect the unused elements and to delete them from the project. The elements deleted from the project will be archived (in ZIP format or in a backup directory) so that they can be restored later if necessary...

- ▶ To use this wizard, select "Tools .. Clear the project directory".

## Managing files and disks

### Question How do I manage the files found on the Pocket PC from a standard WinDev application?

The functions for accessing the Pocket PCs (starting with "ce") allow you to handle the files found on the Pocket PC (copy the files, find out the size of a file, return the list of files found in a directory, and so on).

These functions can be used in a standard WinDev application when a Pocket PC is connected to the current computer.

See the online help (keyword: "Pocket PC, Access functions") for more details.

### Question How do I list all the files found in a directory?

*fDir* associated with the *frFile* constant enables you to return the list of files found in a directory.

For a more advanced use, *fListFile* enables you to retrieve the list of files found in a directory or in several cascading directories.

See the online help (keyword: "fListFile") for more details.



Note

The files found in a Pocket PC directory can also be listed from a WinDev application (**ceDir** and **ceListFile**).

**Question****How do I copy files?**

**fCopyFile** is used to copy files.

See the online help (keyword: "fCopyFile") for more details.



Note

The files found on the Pocket PC can also be copied from a standard WinDev application (**ceCopyFile**).

**Question****How do I create a directory?**

**fMakeDir** is used to create a directory.

See the online help (keyword: "fMakeDir") for more details.



Note

A directory can also be created on the Pocket PC from a standard WinDev application (**ceMakeDir**).

**Question****How do I read a text file?**

Two methods are available:

1. Use **fOpen**, **fReadLine**, **fClose**. For example:

```
FileLine is string
FileNumber is int
FileNumber = fOpen("\My Documents\MYFILE.TXT", ...
                foRead)
IF FileNumber <> -1 THEN
  FileLine = fReadLine(FileNumber)
  WHILE NOT FileLine = EOT
    Info(FileLine)
    FileLine = fReadLine(FileNumber)
  END
END
fClose(FileNumber)
```

2. Use **fLoadText**. For example:

```
FileContent is string
FileContent = fLoadText (" \My Documents\MYFILE.TXT")
Info (FileContent)
```

See the online help (keyword: "Read, In an external file") for more details.

## Tables

### Question How do I modify the search key in a table linked to a file?

The search key in a table control corresponds to the item used as sort criterion for the table rows.

- ▶ To modify the search key of a table linked to a file:
  1. Right-click the browsing table and select "Description".
  2. Click the "Content" tab.
  3. In the "Search Item" combo box, select the file item that will be used as search key.



Note

The search key defined for a table can be modified by programming with **BrowsedItem**.

### Question How do I modify the stored item of a table linked to a file?

The stored item in a browsing table corresponds to the value retrieved in the table for the selected row.

- ▶ To modify the stored item of a table linked to a file:
  1. Right-click the browsing table and select "Description".
  2. Click the "Content" tab.
  3. In the "Stored Item" combo box, select the file item that may be retrieved.



Note

The stored item defined for a table can be modified by programming with **StoredItem**.



Tip

The stored item of a table can be used to link two browsing tables in cascade.

## Hyper File Mobile

**Question** **Is the format of the Hyper File files compatible with the format of the Hyper File Mobile files?**

Yes, the Hyper File format and the Hyper File Mobile format are compatible. Their format is identical. The Hyper File data files and the Hyper File Mobile data files can be used in WinDev Mobile and in standard WinDev.

However, the available size on Pocket PC being restricted and the operating system of Pocket PC being limited, the following features are not supported by Hyper File Mobile:

- the transactions.
- the log process.
- the Hyper File replication. Only the Universal Replication is available.
- the management of file locks and record locks.
- the management of files in Hyper File 5.5 format.

**Question** **How do I disable an integrity constraint?**

The integrity constraints are defined in the analysis when the links are created between the files. Each constraint has a name. All the constraints are enabled by default.

**HSetIntegrity** enables you to disable a constraint. This function can be used before and after some specific processes or in the initialization code of your windows or projects.

See the online help (keyword: "HSetIntegrity") for more details.

**Question** **How do I manage the NULL value?**

To manage the NULL value in your files, you can use:

- In the data model editor:
  - "NULL supported" available in the file description. This option is used to specify whether the management of NULL is supported by this file. In this case, the NULL value can be managed for the different file items.
  - "Default to NULL" available for each item of your files. This option enables you to define the null value as the default value for the item.
- In programming, two properties:

<b>..NULL</b>	Used to: <ul style="list-style-type: none"> <li>• define the NULL value as the default value for a file item during its dynamic description</li> <li>• associate (or not) the NULL value with a file item</li> </ul>
---------------	--

<b>..NullSupported</b>	Used to: <ul style="list-style-type: none"> <li>• define the management mode of the NULL value for a file during its dynamic description.</li> <li>• find out the management mode of the NULL value for a file</li> </ul>
------------------------	---

See the online help (keyword: "Management, Manage the Null value") for more details.

### Question How do I manage a duplicate error when writing into a file?

By default, a window for error management is automatically displayed if a duplicate error occurs when writing into a file (**HAdd** and **HModify**). This window allows the user to modify the values entered.

To find out whether a duplicate error occurred and to process it by programming, **HErrorDuplicates** must be called after **HAdd** and **HModify**.

Example of code:

```
HModify(CUSTOMER) = False
IF HErrorDuplicates() THEN
  Error("Unable to modify the customer", ...
        "A duplicate error occurred")
END
```

### Question How do I manage an integrity error when writing into a file or when deleting from a file?

By default, a window for error management is automatically displayed if an integrity error occurs when writing into a file (**HAdd** and **HModify**) or when performing a deletion (**HDelete**).

To find out whether an integrity error occurred and to process it by programming, **HErrorIntegrity** must be called after **HAdd**, **HModify** and **HDelete**.

Example of code:

```
HDelete(CUSTOMER) = False
IF HErrorIntegrity() THEN
  Error("Unable to delete the customer", ...
        "An integrity error occurred")
END
```

**Question** How do I fill a TreeView control from a file?

Examples of code used to fill a TreeView control from a file:

**Algorithm:**

```
// Browse the file
HReadFirst(FileName, KeyName)
WHILE HOut() = False
  // Add to the treeview
  TreeAdd(NameTreeViewControl, Root+TAB+Branch+TAB+Leaf)
  HReadNext(FileName, KeyName)
END
```

**Example 1.** Load the companies by city:

```
// Browse the file
HReadFirst(COMPANY, CITY)
WHILE HOut() = False
  // Add to the treeview
  TreeAdd(TreeViewControl, Company.CITY+TAB+Company.CompName)
  HReadNext(COMPANY, CITY)
END
```

**Example 2.** Load the customer names by initials (datebook organization):

```
// Browse the file
HReadFirst(CUSTOMER, CUSTNAME)
WHILE HOut() = False
  // Add to the treeview
  TreeAdd(TreeViewControl, Left(Customer.CustName, 1)+TAB+ ...
          Customer.CustName)
  HReadNext(CUSTOMER, CUSTNAME)
END
```

**Example 3.** With a shorter syntax (using the **FOR EACH** operator):

```
// Browse the file
FOR EACH CUSTOMER ON CUSTNAME
  // Add to the treeview
  TreeAdd(TreeViewControl, Left(Customer.CustName, 1)+TAB+ ...
          Customer.CustName)
END
```

**Question** How do I manage a composite key during a search?

Start a search with **HFilter** or **HReadSeek** by using the following notation:

```
HReadSeek(File, COMPOSITEKEY, [1, "A"])
HFilter(FILE, COMPOSITEKEY, [1, "A"], [5, "S"])
```

## CEDB

---

**Question** How do I handle the information included in the applications found by default on my Pocket PC (contacts, notes, and so on)?

The applications found by default on the Pocket PC (management of contacts, appointments, tasks, and so on) use databases in CEDB format.

To handle the information found in these applications, use the CEDB functions (starting with "cdb") and the structure of the standard databases.

See the online help (keywords: "Database (Pocket PC), WLanguage functions" and "Database (Pocket PC), Structure of standard databases") for more details.



Note

The CEDB databases can be handled from a standard WinDev application and from a WinDev Mobile application.

**Question** How do I create a CEDB database?

No CEDB database can be created directly. To have a CEDB database on the Pocket PC, an Access database (".MDB" files) must be exported from a PC.

When an Access database (".MDB" file) is copied to a Pocket PC, this database is automatically changed into a CEDB database (".CDB" file).

This database will be called custom CEDB database.

**Question** Why should I use a CEDB database?

The use of a CEDB database enables you to:

- handle the data managed by the standard applications of Pocket PC.
- retrieve and manage an Access database found on the PC.

For all the other cases, we recommend that you use the Hyper File Mobile databases.

## Queries

---

### Question How do I optimize the speed of a query?

The optimization of a query is based on the following principle: a wizard defines the composite keys to modify and/or to add into the analysis associated with the project.

To optimize the execution speed of the current query, select "Query .. Optimize the query". The window for query optimization is displayed and it proposes the different modifications that can be made to the analysis.

**Caution:** Adding a lot of composite keys to an analysis increases the size of the index files and slows down the access to the data files.

**Note:** This option is available only if the following conditions are fulfilled:

- "Live Data" is enabled ("Project .. Project Description .. Live Data"). See the online help (keyword: "Live Data") for more details.
- the data files are found in the test directory of the project ("Project .. Project Description .. Files"). See the online help (keyword: "Test, Project") for more details.

### Question How do I add or modify a condition in a query?

- To add or modify a condition in a query:
1. Select and open the query to modify.
  2. Right-click the query and select "Query Description".
  3. Select the item for which a selection condition must be added or modified.
  4. Click "Selection Conditions" and select "New condition" or "Modify the condition".

**Reminder:** When defining a condition, you can refer to a value (constant in the query) or to a parameter (value passed in parameter when calling the query).

### Question How do I add or modify a sort in a query?

- To add or modify a sort in a query:
1. Select and open the query to modify.
  2. Right-click the query and select "Query Description".
  3. Right-click the requested item, select "Sort..." and choose the operation to perform.

**Reminder:** The red arrow is used to change the sort direction.

## Printout

---

### Question How do I print from a WinDev Mobile application?

The PCL format is the format used when printing from a Pocket PC. This printout can be performed in a PCL file or on a PCL printer directly.

To format the information to print:

- create a report via the report editor of WinDev Mobile.
- use the print functions of WLanguage.

See the online help (keywords: "Report (Report editor)" and "Printout") for more details.

### Question What is the PCL standard?

PCL (Printer Control Language) is a standard allowing the Pocket PC to send commands to a printer that supports it. This standard was developed by Hewlett Packard.

A PCL file is a binary file containing all the commands sent. This file contains the commands required to define the print areas.

### Question Why does the font on the printed page differ from the font in my report?

The result of a printout performed on Pocket PC depends on the features of the printer used (management of images and lines, management of fonts, and so on).

To limit the resources required for printing, WinDev Mobile uses the printer fonts. The printer automatically selects the font that best suits the specified criteria. Check the fonts available on your printer as well as their sizes and effects.

## Ports

---

### Question How do I read a bar code?

Two methods can be used to read a bar code:

1. If you are using a bar code reader that is directly interfaced with the keyboard, you won't have to write any code in WinDev Mobile. When the bar code is read, the value is returned to the keyboard as if the code was entered directly. To do so, you must be positioned in an edit control.
2. If you are using a bar code reader that is connected to the serial port, you must use the functions for managing the serial ports. **sOpen**, **sRead**, **sWrite** and **sClose** are used to manage the

dialog with a serial port.

**Question** How do I read the data sent by a reader of magnetic card?

Proceed exactly the same way as for bar code readers.

**Question** How do I manage a serial port?

- ▶ Use *sOpen*, *sRead*, *sWrite*, *sClose*, and so on.

See the online help (keyword: "Serial port") for more details.

**Question** How do I manage a parallel port?

The functions for managing the parallel ports are the same as the functions for managing the serial ports.

- ▶ Use *sOpen*, *sRead*, *sWrite*, *sClose*, and so on.

See the online help (keyword: "Parallel port") for more details.

**Question** How do I manage an infrared port?

The functions for managing the infrared ports are the same as the functions for managing the serial ports.

- ▶ Use *sOpen*, *sRead*, *sWrite*, *sClose*, and so on.

See the online help (keyword: "Infrared port") for more details.



# CONCLUSION

The lesson is over!

This course has discussed a variety of subjects, but not all the features of WinDev Mobile, far from it!

You are now familiar with the main concepts.

We recommend that you to spend another day exploring the menu options of WinDev Mobile, for each one of the modules.

You can also explore the examples supplied with WinDev Mobile: some are simple and only address one topic, while others are more complex. These examples illustrate the different aspects of WinDev Mobile. Reading the source code is also a good way to learn.

It would take too much room to discuss all the available topics (there are hundreds, even thousands!). WinDev Mobile proposes several features that were not presented in this course:

- sockets functions, HTTP, telephony,
- creation of skin templates,
- nested reports, queries with parameters,
- ...

See the online help for more details.

Reminder! To benefit from intermediate updates and tips, subscribe to **LST** (quarterly magazine + CD), in French.

We wish you great development experiences with **WinDev Mobile!**

# APPENDIX

## Glossary of the main words

A quick vocabulary reminder.

Window	A <b>window</b> is also called a " <b>dialog box</b> ". We will use both "dialog box" and "window"; we hope the purists won't mind. A window can also be called a screen.
Button	A <b>button</b> (text or graphic) is an area materialized by a "rectangle" that can be clicked in order to perform an action. In the toolbars, we talk of icons rather than buttons. A button is also called a "control".
Project	A <b>project</b> is a set of windows, reports, and so on. A project can use an <b>analysis</b> created by the data model editor.
Analysis	An " <b>analysis</b> " is a set of files (or tables) created to build a database.
Application	An <b>application</b> is a set of programs that perform specific actions.
Control / Item	We will use " <b>control</b> " for the screen areas and " <b>item</b> " for the file areas. Therefore, we will talk about the "NAME" control (that is found in the window) and about the "NAME" item (that is found in a file); there can also be a program variable named "NAME".
Table	A <b>table</b> is a control used to display the content of a file or the content of a memory zone. A table is also called "browse" or "data sheet".
Combo box / Drop-down list box	A " <b>combo</b> " or " <b>combo box</b> " corresponds to a " <b>drop-down list box</b> ".
Scrollbar	We will use <b>scroll bar</b> or <b>scrollbar</b> .
File	A <b>file</b> is also sometimes called a "table". We will use " <b>table</b> " for an object used to view the content of a file or the content of a memory zone.
Record	A <b>record</b> is sometimes called a row. A record groups several items found in a file.
Item	An <b>item</b> is an area that belongs to a record.
SQL language	The <b>SQL language</b> is a language used to handle the data found in the files. It is both a query language and a language used to update the files (addition, modification, deletion). This language can be used in the query editor or by programming. However, no prior knowledge of the SQL language is required to use the query editor.