
16
DEVELOP 10 TIMES FASTER
PC SOFT

WinDev Mobile 16
Documentation version 16 - 0311

Remember to visit our Web site (**www.windev.com**) on a regular basis to find out whether upgrades are available.

Email address of Free Technical Support: **freetechnicalsupport@windev.com**

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

TABLE OF CONTENTS

Introduction

PART 1 - DISCOVERING WINDEV MOBILE

Before we start ...

PART 2 - FIRST APPLICATIONS

Lesson 2.1. Your 1st application

Lesson 2.2. Application on Smartphone

Lesson 2.3. Android development

PART 3 - DATABASES

Lesson 3.1. Introduction

Lesson 3.2. HyperFileSQL Mobile files

PART 4 - SPECIFIC FEATURES OF POCKET PC

Lesson 4.1. Specific Formats

Lesson 4.2. Interactions between applications

Lesson 4.3. Sharing some WinDev elements

Lesson 4.4. Input mode

PART 5 - COMMUNICATION

Lesson 5.1. Introduction

Lesson 5.2. Managing emails

PART 6 - HOW DO I PROCEED?

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 our standard WinDev product 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 our standard WinDev product, we recommend that you to read the WinDev tutorial beforehand.

Note: To receive the standard WinDev tutorial, 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 well worth it!

If you try to develop an application before practicing, you will lose time, and a lot 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 156).

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!

Tip

The Tutorial may have evolved since this document was published. Don't forget to consult the online version of the Tutorial (PDF file accessible from "? .. Tutorial .. Tutorial (PDF)".

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, press 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.

Legend of the symbols

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

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 15 ...

If you are familiar with WinDev Mobile 15, following this course will do no harm: it's a good opportunity to "review" 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, booth, 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, ...
- On-call doctors or vets
- Taking information in a temporary outlet: trade fair, street poll, stadium, ...
- Restoring leased heavy equipment (tools, vehicles, and so on) to a parking lot
- ...

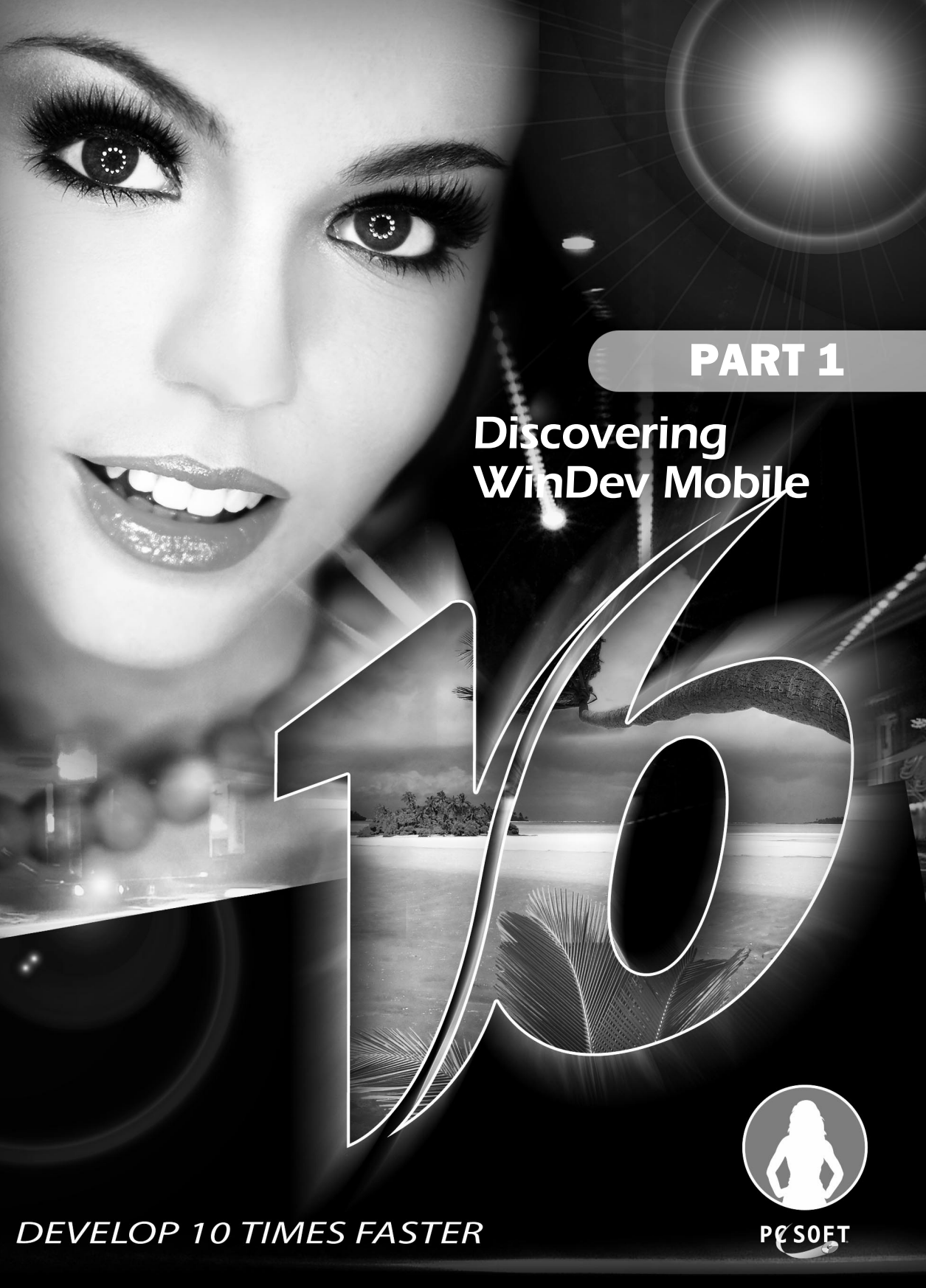
WinDev Mobile is a development environment that includes all the tools required for developing an application.

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, 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.


PART 1
Discovering
WinDev Mobile
16
DEVELOP 10 TIMES FASTER
PC SOFT

BEFORE WE START ...

This lesson will teach you 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 16 allows you to manage, step by step, the entire development of an application for Pocket PC, from conception to deployment.

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

WinDev Mobile 16 enables you to create applications for managing data. WinDev Mobile applications access most of the available databases. WinDev Mobile 16 is supplied with HyperFileSQL Mobile, a powerful database already used in thousands of sites!

WinDev Mobile 16 proposes a powerful work environment. Your teams will be able to easily create user-friendly applications.
The window editor of WinDev Mobile 16 is 100% WYSIWYG ("What You See Is What You Get"). It enables you to easily create outstanding windows linked to data.

Starting WinDev Mobile

This lesson teaches you to perform your first operations (don't worry, nothing too difficult!) in the environment of WinDev Mobile 16.

▶ Start WinDev Mobile 16:
- click the desktop icon:

- select "Start .. Programs .. WinDev Mobile 16 .. WinDev Mobile 16".

WinDev Mobile starts.

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.

Tip

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 children for instance).

To customize the wizards, right-click the image of a wizard and select the image to use.

For example:

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

Environment of the Tutorial

WinDev Mobile allows you to configure the environment. Several modes are available:

- **Simplified environment**: This mode enables you to discover the main features of WinDev Mobile.
- **Full environment**: This mode proposes all the features of WinDev Mobile, including the most recent ones.
- **Retrieve the configuration of your xx environment**: This mode restores the features available in version xx (if version xx is installed on your computer).

At any time, regardless of the type of environment used, you have the ability to add or delete the access to some unused features.

To follow this Tutorial, we advise you to work with a simplified environment. The advanced features will be added as this Tutorial goes along.

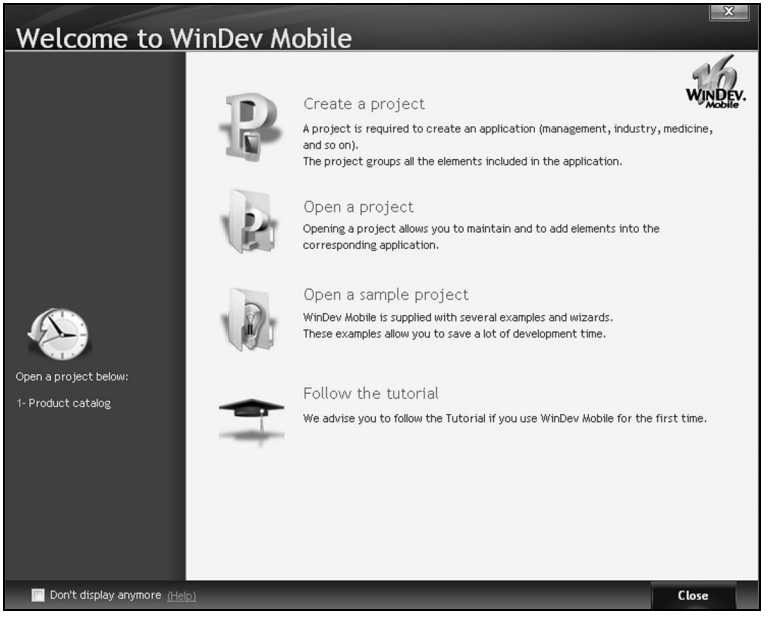
- To use the simplified environment of WinDev Mobile:
  1. Start WinDev Mobile 16.
  2. A welcome wizard starts if WebDev 16 was never started before. This wizard enables you to choose your work environment.
  3. Select "Simplified environment" and validate.
- This welcome wizard is not displayed if WinDev Mobile was already started on your computer. To check and modify (if necessary) the configuration of your environment, perform the following operations:
  1. Select "Tools .. Options .. Options of the environment".

2. Click "Restart the wizard for configuring the environment...".
3. Select "Simplified environment".
4. Validate your choice.
5. Validate the options of the environment.

That's it, WinDev Mobile is configured to follow the Tutorial.

Overview of WinDev Mobile

During the first startup of WinDev Mobile, once the environment was configured, 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.

Note: If WinDev Mobile opens directly on an existing project, select "File .. Open a project" from the popup menu of WinDev Mobile and select the Pocket Beach.WPP project found in the "Examples\Mobile Windows\Beach Booking\Pocket Beach" sub-directory of the setup directory of WinDev Mobile.

3. Click the "Open" button to validate. The "Pocket Beach" project opens.

The WinDev Mobile environment

The WYSIWYG ("What You See Is What You Get") environment of WinDev Mobile 16 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 redisplay the elements that were previously opened.

3. The "Wizards, Examples and Components" pane proposes a set of elements: components (elements that can be easily re-used in all your projects), preset controls, examples, ... An invaluable time saver!

4. The "Project Explorer" pane returns the list of project elements: a double click performed on an element opens this element in the relevant editor. This pane is used to perform searches in the project.

The environment of WinDev Mobile 16 is highly intuitive. The different panes can be displayed at any time via "Display .. Toolbars .. Panes".

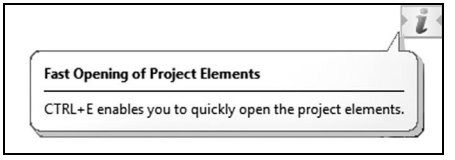
- The dashboard enables you to easily find out whether bugs have occurred in the project, whether automatic tests have been created and whether the project can be optimized. It also gives you the ability to find project elements.
- You are now going to display the graph of the "Pocket Beach" project: click the "Graph" button in the dashboard or select "Project .. Project graph" .
  The project graph is used to view the sequences between the different project elements.

Note

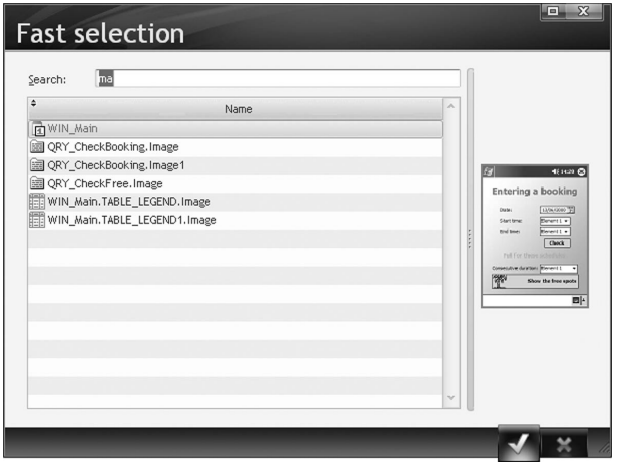
FishEye
To enlarge the section of the project graph hovered by the mouse cursor, use the FishEye effect ( ).

- You are now going to 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 diagram.

You've probably noticed that some information was displayed in the top right corner 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.



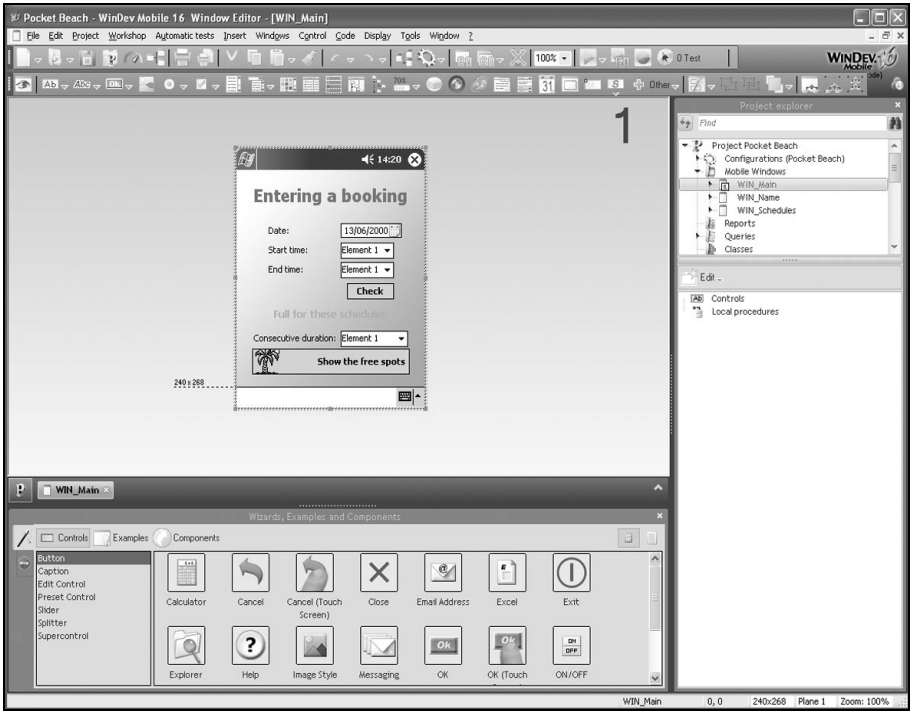
This AAD proposes to find some elements via [CTRL]+[E].
Enter the letters found in the sought element and the list of corresponding elements is displayed:



- Double-click the selected window to open it.

The window editor

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



- The digit displayed at the top represents the number of the plane currently displayed. This window contains several planes: the [Pg Up] [Pg Dn] keys of your keyboard allow you to position on the first plane ("1" is displayed in the top right corner).
  Note: A plane is used to display a set of controls found in a 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 some 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, ...

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

- Select the "Check" control in order to view the code associated with it. To do so:
  1. Click the "Check" control. This control is named "BTN_Check".
  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, ...

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

For example, in the code window that was just opened:

- The "Hourglass" word is colored in blue: it is a WLanguage function.
- The "COMBO_STARTTIME" word is colored in cyan: it is a project element.
- The "hHour" word is colored in green: it is a local variable.

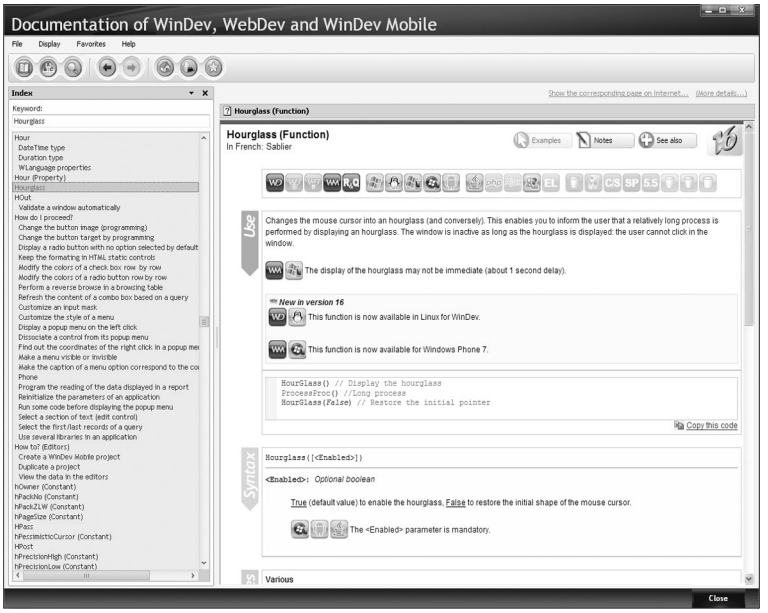
An assisted mode is also available for entering the source code: when typing the name of a function, the type of parameter expected by this function is displayed in a tooltip and in the status bar of WinDev Mobile. The completion is also available for the names of the variables or functions.

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

▶ You are now going to display the help about the ***Hourglass*** function. To do so:

1. Position the mouse cursor on the name of the function ("Hourglass").

2. Press [F1] on your keyboard. The help of the function is displayed in a specific "help browser".



If your computer is equipped with an Internet access, the help page currently found in the "help browser" can be displayed in its Internet version ("View the corresponding page on Internet" link).

Notes

The online help for WinDev, WebDev and WinDev Mobile is available on Internet. Therefore, the online help can be accessed from any computer equipped with Internet access, without having to install the product. This help is updated on a regular basis.

Each Web user can add some comments about the documentation pages: personal notes, examples, links, ...

The online help of WebDev enables you to get information about the 2000 WLanguage functions. The online help also contains the help about the editors, controls, tips, ...

The online help is common to WinDev, WebDev and WinDev Mobile. The pages displayed correspond to the product currently used.

To start the Internet online help from the product:

1. Select "Tools .. Options .. General options of WinDev Mobile".
2. In the "Help" tab, select the access mode to the help database.

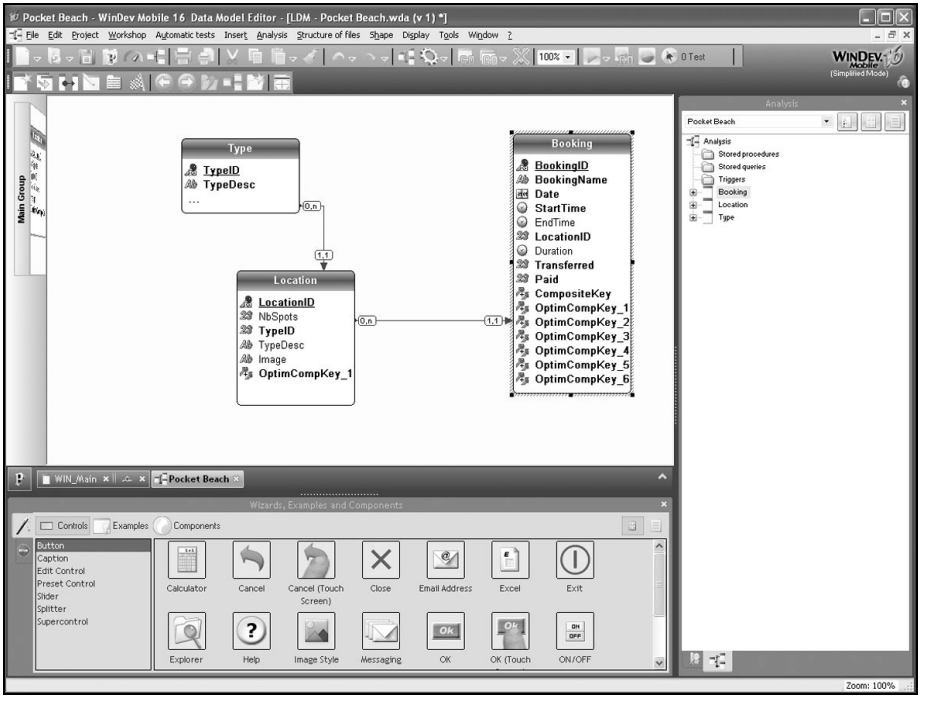
The data model editor

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 are now going to open the analysis found in the "Pocket Beach" example. To do so, select "Project .. Load the analysis".
  The data model editor is opened.



Several data files linked together are found in the analysis.

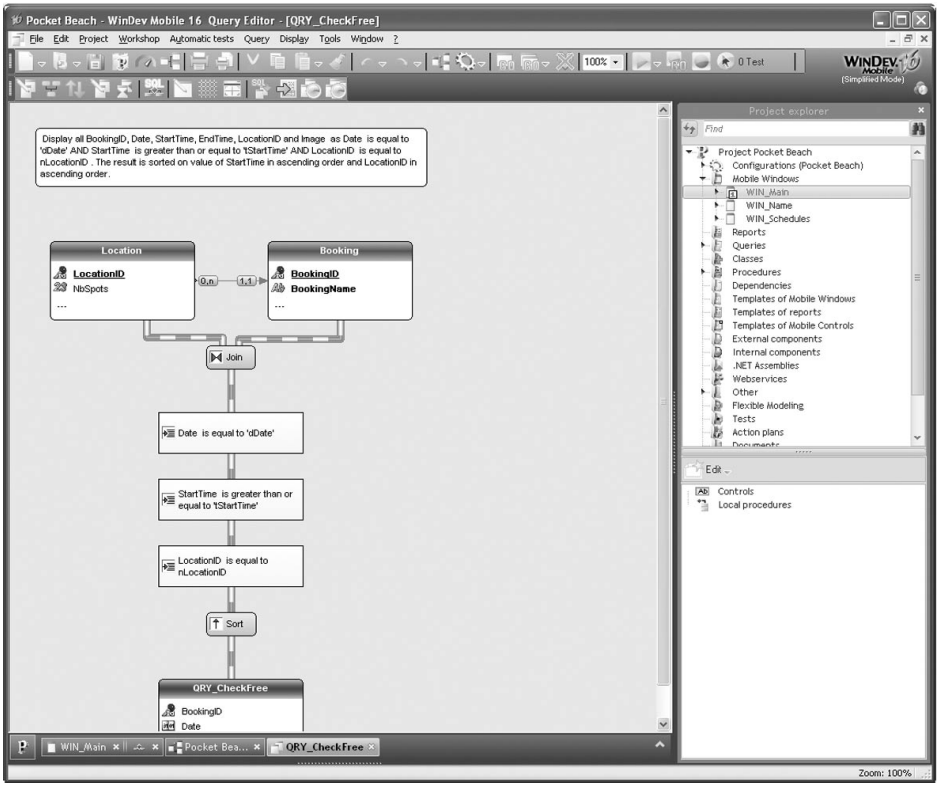
WinDev Mobile 16 enables you to create some applications linked to data via RAD (Rapid Application Development, "Workshop .. Full application RAD").

The query editor

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

- You are now going to 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 query named "QRY_CheckFree" is found. Double-click its name to open it.

The query editor is displayed:



This query is used to display the free spots.

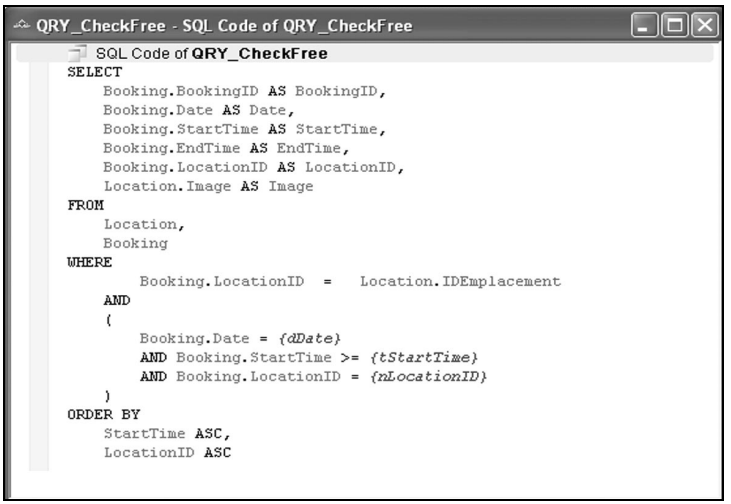
The data source of a query can be a HyperFileSQL Mobile database or an external database (AS/400 for instance).

Choose the items to include, enter the selection conditions via the query creation wizard, ... 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 WinDev Mobile is displayed in the query editor:



The report editor

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

The method for creating a report is the same as then method for creating a window!

The principle for printing a report in Pocket PC is straightfoward: 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.

Some WLanguage code can be entered in all the report elements: the most specific processes can be performed without hassle.

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 16.

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 or our "Assistance Directe©" service 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.


PART 2
First
applications
16
DEVELOP 10 TIMES FASTER
PC SOFT

LESSON 2.1. YOUR 1ST APPLICATION

This lesson will teach you the following concepts ...

- 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

The "Discovery.WPP" project corresponds to the full project with the answers to this lesson. To open this project, select "? .. Tutorial .. Discovery".

You can follow this lesson without opening the project.

Now down to work!

Starting WinDev Mobile

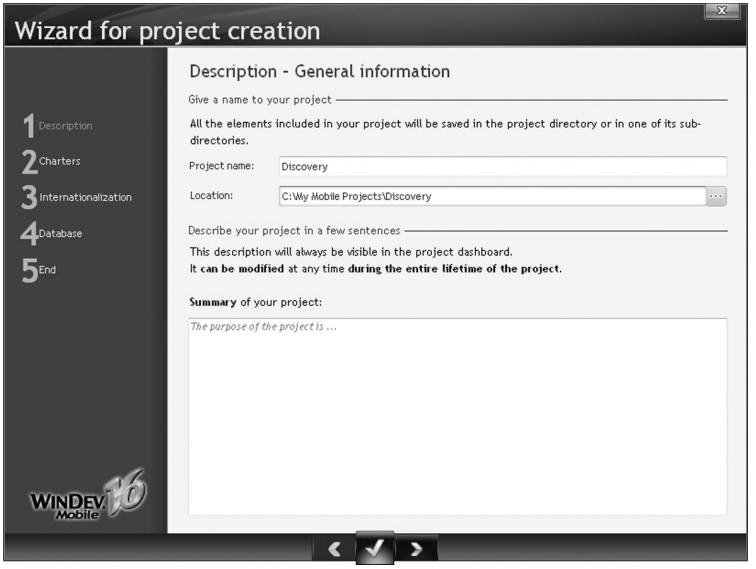
The following window is displayed when WinDev Mobile is started:

This window gives access to the most common features when starting WinDev Mobile. 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 ("Discovery"). The name of the project is the name that will be used to save the project file, with the "WPP" extension.
  The project directory is initialized by default ("My Mobile Projects\Discovery").



- Go to the next wizard screen.

  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 "Description - Type of project". In this example, we are going to create an application for Mobile. Select "Windows Mobile application" and go to the next plane.
- If a device (Pocket PC, Smartphone, ...) is currently connected to the PC, its characteristics can be automatically detected ("Click here to start the detection" button). Otherwise, go to the next plane to choose the device to use.
- Select (if necessary) the type of device on which your project will be run ("Windows Mobile 5/6" for example).

  WinDev Mobile enables you to develop applications for different families of products: Pocket PC, Smartphone, Android, ...
  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 image 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.

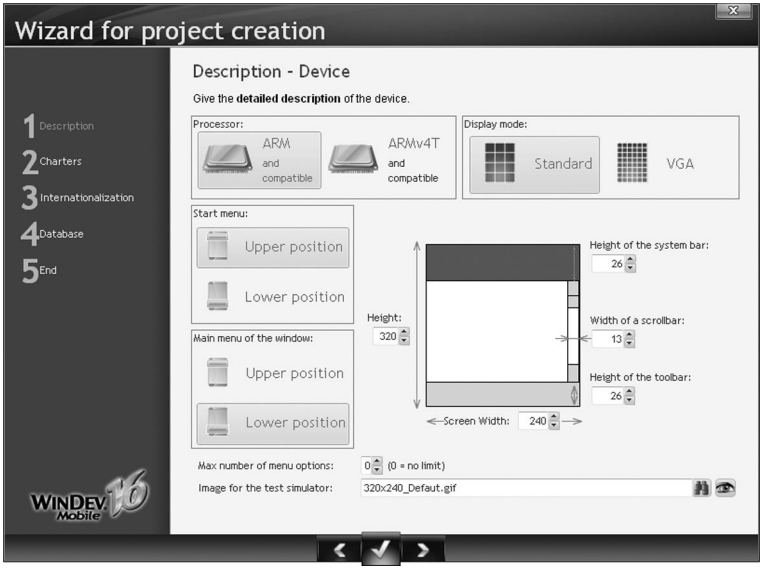
Note

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, ...).

See “Application on Smartphone”, page 57 for more details.

▶ Go to the next plane: you have the ability to create a detailed description of the platform used:



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

▶ Display the next screens until you reach the "Charters - Style book" screen.

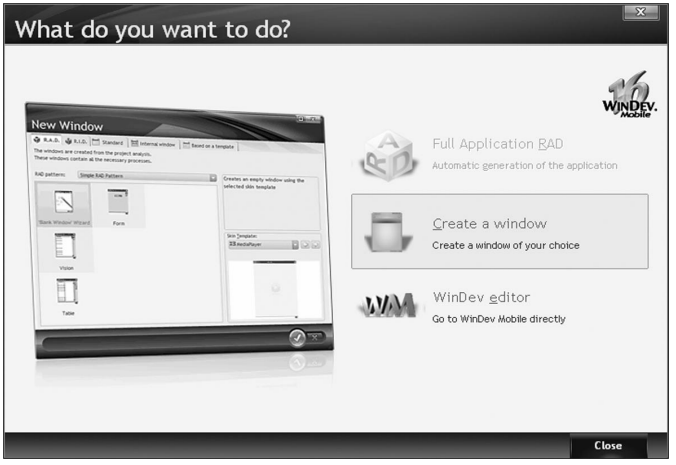
▶ You have the ability to choose the style of your application by selecting a style book. Select the "Media Center" theme for example.

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, the applications built for Pocket PC all look alike: a square window, a white background, a blue title bar, gray buttons, no image, ... 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. Our 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:



Select "WinDev editor".

Create 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:

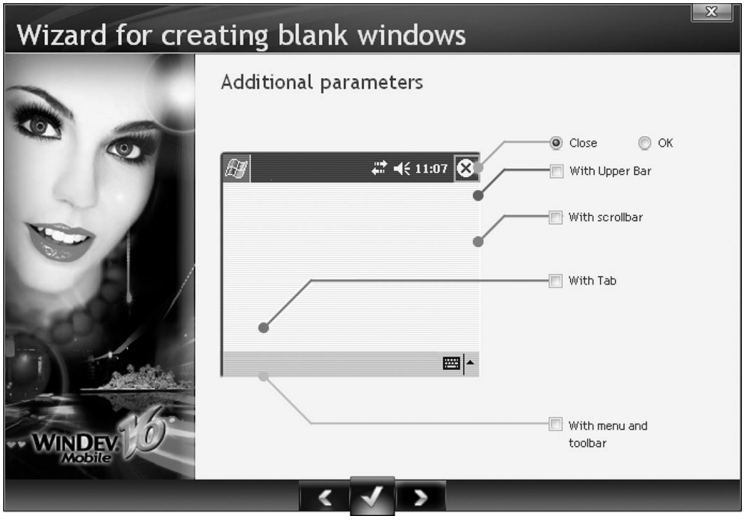
  1. Click in the toolbar of WinDev Mobile.
  2. Click "Window" to create a new window.
  3. Select "Wizard for blank window".
  By default, this window will use the skin template selected during the project creation.
  4. Validate. The wizard for creating a blank window starts. We are now going to enter information about the window (type, name, title, ...).

- 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 a 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 closing of the window.
- ***Upper bar***: used to display some information, some buttons, ...
- ***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***: allows the user to easily access a feature of the application. In an application for Pocket PC, this menu is located at the bottom of the windows.

▶ Keep the options selected by default ("Close" button) and go to the next screen.

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 validate the window.
  This button is associated with a process used to customize the closing of the window.
- Border of window.

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. The created window is displayed in the window editor.
Note: All the window characteristics specified in this wizard can be modified afterwards 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, ...).

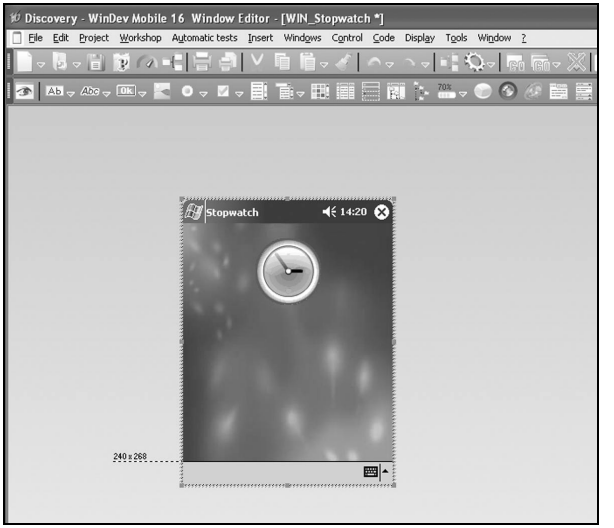
This feature enables you to easily identify the element that will be used in your programs.

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

Create the controls found in the window

Creating the image control

To give this window a cheerful appearance, 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.
   - uncheck "Clipart" and 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). Validate.
6. Click (at the bottom of the description window) and enter the following code line in the initialization code:

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

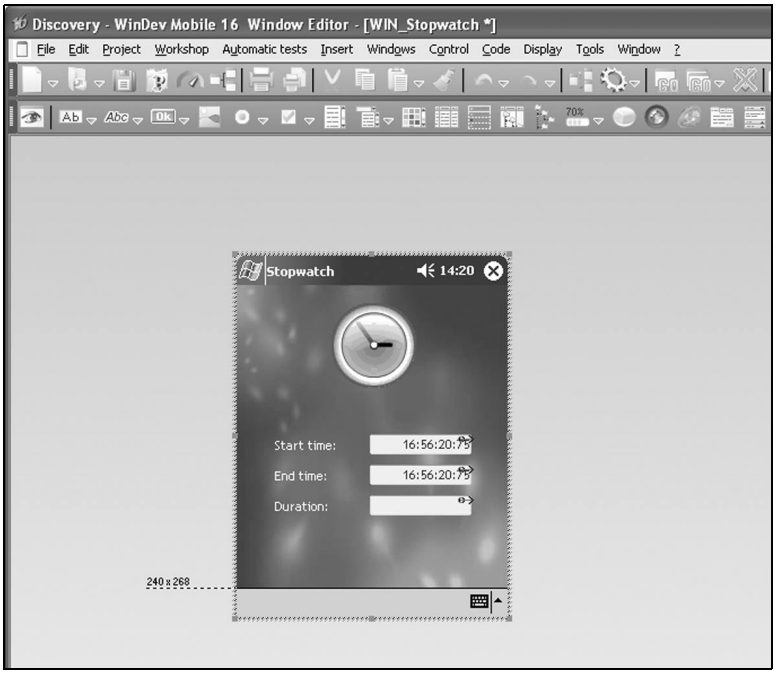
Disabling the animation of the image

Note

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

Creating the 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 the arrow found on the right of the icon Ab. A window presenting the different types of edit controls is displayed. Click the "Time" control and click the position where the control must be created in 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:". This control is a "Time" control. Indeed, this control will be used to display the start time of the stopwatch.

5. 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 a control corresponds to the mask defined by the project ("Project .. Project description", "Language" tab). Therefore, the same mask is automatically used in all the application controls. This feature is also very useful in the multilingual applications.

6. Select the "GUI" tab.

7. Select "Read-only". No data can be entered by the user in this control.

8. Validate.

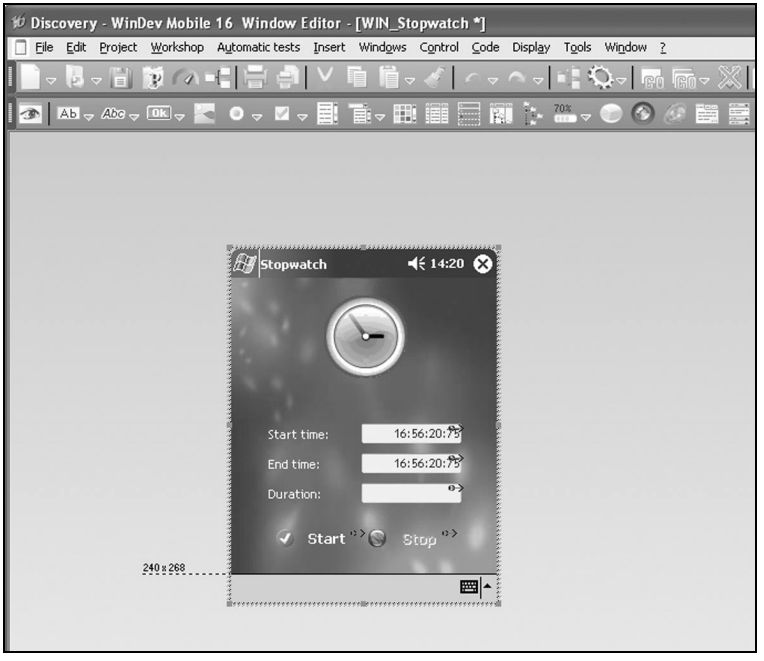
- The second edit control will be created by "Copy/Paste":
  1. Select the control that was just created.
  2. Press [CTRL] + [C], then [CTRL] + [V]: a new control is automatically created. Modify the name and caption of this new control:
  - its name: "EDT_EndTime".
  - its caption: "End time:".

- To create the third edit control:
  1. Click the arrow found on the right of the icon . A window presenting the different types of edit controls is displayed. Click the "Duration" control and click the position where the control must be created in the window.
  2. Right-click the control and select "Description".
  3. Enter the name of the control: "EDT_Duration". This control will display the elapsed time.
  4. Enter the caption: "Duration:".
  5. Select the "GUI" tab.
  6. Select "Read-only". No data can be entered by the user in this control.
  7. 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 , then click the location where the button must be created in the window.

2. Right-click the control and select "Description".
3. Enter the caption: "Start".
4. Click and enter the following code lines:

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

- To create the procedure used to calculate the elapsed time:

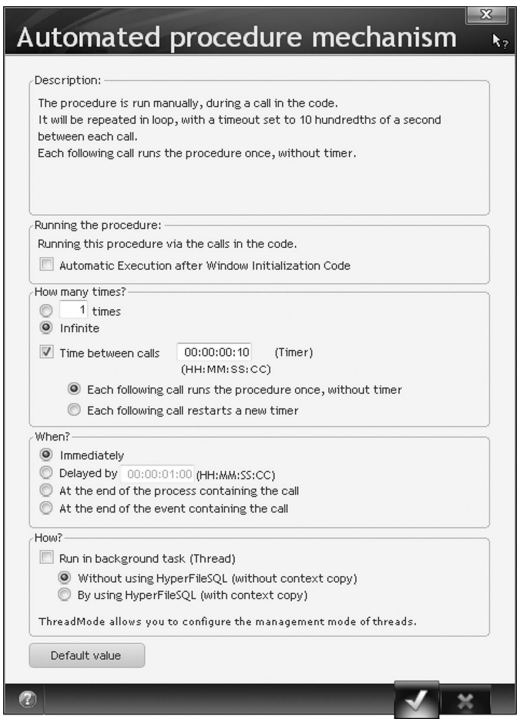
1. Click the "Stopwatch" word in the code editor and select "Insert .. New local procedure" (or press [F4]).
2. The name of the procedure ("Stopwatch") is automatically proposed. Validate this name.
3. Enter the following code lines:

```
PROCEDURE Stopwatch()
EDT_EndTime = Now()                      Display the end time
EDT_Duration = ChronoValue()             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 on 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 Ok, then click the location where the button must be created in the window.
2. Right-click the control and select "Description".
3. Enter the caption: "Stop".
4. In the "GUI" tab, specify that the initial status of the control is grayed. Indeed, the button will be enabled only when the "Start" button is clicked.
5. Click and enter the following code lines:

```
EndAutomatedProcedure(Stopwatch)
```
Stop the automatic call to the procedure

```
ChronoEnd()
```
Stop the stopwatch

```
IMG_AnimatedImage..Animation = False
```
Disable the animation of the image

```
BTN_Stop..State = Grayed
```
Make the "Stop" button grayed.

6. Display the code of the "Start" button: select the control and press F2. We are going to enable the "Stop" button when a click is performed on the "Start" button. Enter the following code after the existing code:

```
BTN_Stop..State = Active
```
Make the "Stop" button active.

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

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

Test of a WinDev Mobile window

WinDev Mobile proposes several types of test:
- 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.
- test and debug on the Pocket PC connected to the development computer. This test is directly performed on the Pocket PC but its allows the use of the debugger.
- 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, ...).
Some 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 run the test of the window in simulation mode.

- Click the "GO" icon (or press the [F9] key). WinDev Mobile informs you that the test will be run in simulation mode. Validate this screen ("Yes" button). 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!

Tip

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:

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.

GO Test

The automatic tests

Once a window test has been run, WinDev Mobile allows you to save the corresponding automatic test via GO.

The automatic tests are a category of specific tests. The automatic tests are used to automatically perform some operations of your windows. These tests are recorded as WLanguage scenarios and they can be easily modified in the code editor.

Once recorded, the automatic test can be re-run as many times as necessary, to test for instance the impact of a modification made to a window, a procedure, ...

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

Test and debug on the Pocket PC

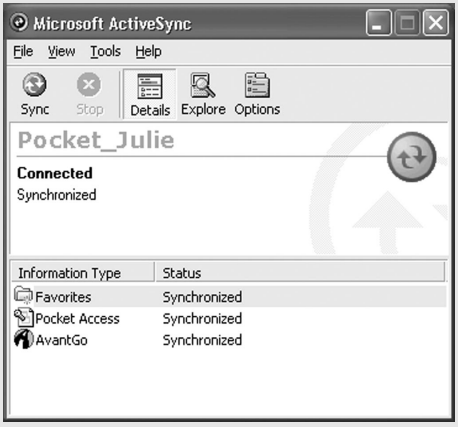
Let's now run the test of the window on the Pocket PC. To run this test, a Pocket PC must be currently connected to the current computer.

Note

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

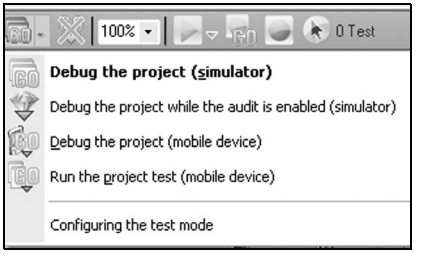
In most cases, ActiveSync is supplied with the Pocket PC. ActiveSync can also be downloaded from Internet.

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



From Windows Vista, the "ActiveSync" program has been replaced by the "Manager for Windows Mobile devices".

- First, we are going to 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, click the arrow found on the right of the "GO" icon. A drop-down menu is displayed:



- Select "Debug the project (mobile device)".
- 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 a single one in our example.
  This window will be called "First project window".

Select the proposed window: "WIN_Stopwatch" and validate.

2. The wizard for executable creation is automatically started.
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 the Pocket PC" is displayed.
Check "Yes: Copy the executable to the Pocket PC".
Uncheck "Run the application on Pocket PC at the end of copy". Validate the wizard once again.

Note

Copying the executable onto the Pocket PC can take a few minutes (indeed, the PC SOFT Framework is copied when installing a WinDev Mobile application 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.

4. Select "GO .. Debug the project (mobile device)".
5. 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

No on-the-fly correction is available 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 found on the right of the "GO" icon. A drop-down menu is displayed.
- Click "Run the project test (mobile device)".
- 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 be run only 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 executable of the application (see “Creating the executable program”, page 53).

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:

Type of window

Managing the scrollbar

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 the 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, ...).
- 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. The 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.

We are going to display a dialog box 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 "Button code". The "Before closing with OK/Close button" process is displayed.
  3. Enter the following code lines:

```
SWITCH Dialog(                                    Display a dialog box
```

  4. A "Wizard" tooltip appears: click this tooltip. The wizard for using directive questioning is displayed. Go to the next plane.
  5. The wizard proposes a list of questions. Enter "close". The question "Do you want to close this window?" appears. Select this question in the table. Go to the next screen.
  6. Validate the wizard to include the message in your code.

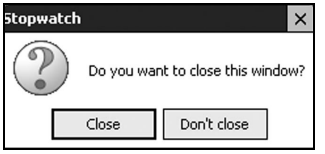
7. Complete the code as follows:

```
SWITCH Dialog("Do you want to close this window")
   // Close
  CASE 1
  Close
  // Don't close
  CASE 0
END
```

Display a dialog box

Yes (case 1): the window is automatically closed.
No (case 0): the window remains opened.

8. Run the test of 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 closing of the window.

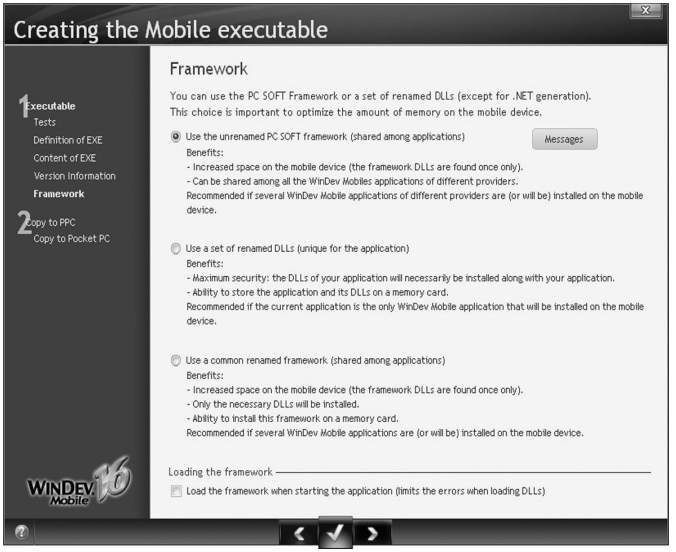
Creating the executable program

Create 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 WinDev Mobile environment:

1. Select "Workshop .. Generate the Windows CE/Mobile application" or click  .
The wizard for creating an executable starts.
2. A screen informs you that no automatic test was created. Go to the next screen.
3. The general options of the executable are displayed.
By default, the executable will be created with the same name as the project.
4. Click the "Catalog" button to select the icon associated with the executable.
5. Choose an image among the proposed ones (a stopwatch for instance). Double-click the image and validate.
6. Click the "Framework" link found on the left of the wizard.



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:

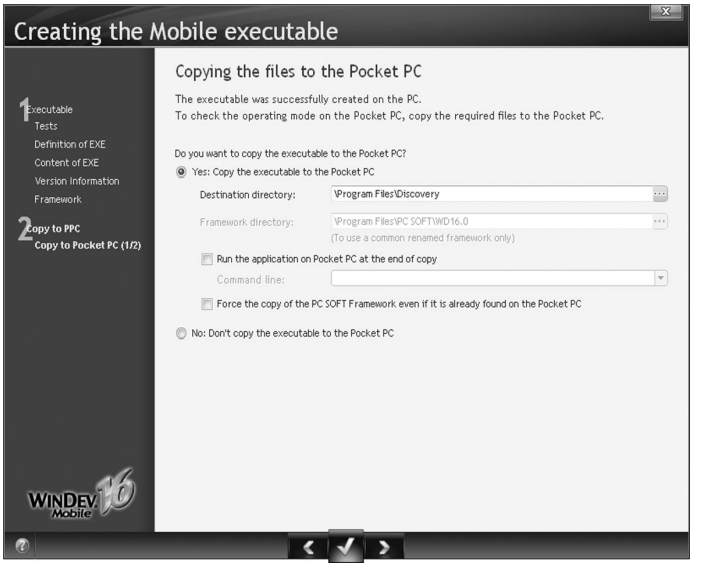
- to **use the PC SOFT Framework**: the WinDev Mobile Framework will be installed in the "Windows\PC SOFT\WD16.0" directory of Pocket PC.
  This Framework is copied onto the Pocket PC during the first setup of the application or when updating the application.
  This option is used to limit the amount 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 in order to avoid conflicts with the other Frameworks.
  This Framework is copied onto the Pocket PC during the first setup of the application or when updating the application.
  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.

7. 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 have created a stand-alone executable program. This program is compatible with all the versions of Windows for Pocket PC!

Run 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 onto the Pocket PC connected to the current computer.



- To copy the executable onto the connected 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. Unselect "Force the copy of the PC SOFT Framework even if it is already found on the 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) onto 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 during the next synchronization between the PC and the Pocket PC.

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 a reinitialization 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 the following concepts ...

- **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

The "SMS.WPP" project corresponds to the full project with the answers to this lesson. 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 some SMSs and to read the 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.
- and/or on a Windows phone.

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 some SMSs.
These examples are accessible from the "Wizards, Examples and Components" pane.

Note: Several generations of cell phones are available:
- the smartphones, that allow you to handle the applications via two menus and a joystick. These phones are not touch-friendly and they use a "Windows 2003" operating system.
- the "Windows phones" on which the applications behave like standard Pocket PCs. In most cases, these telephones have touch-friendly screens and they use a "Windows Mobile" operating system.

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, ... This information is stored in the SMS structure.

The SMS structure is as follows:

ReceiveDate	Date and time when the SMS was received.
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: • smsInternationalNumber (default value): these numbers can be accessed anywhere and are in 06.xx.xx.xx.xx. format • smsNationalNumber: short numbers, accessible within the country only

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 number of the recipient. 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.

This application is created for a Smartphone but it can be adapted for a touch-friendly phone for example.

Creating the project

▶ To create our application:

1. Select "File .. New .. Project". The wizard for project creation starts.

2. Specify the name and summary of the project: "SMS" and "Project for sending and reading SMSs". Go to the next screen.

3. There is no document to attach. Go to the next screen.

4. The generated application will be a windows Mobile application. Go to the next screen.

5. If a Smartphone is currently connected to the PC, its characteristics can be automatically detected. Otherwise, go to the next plane to choose characteristics of the device used.

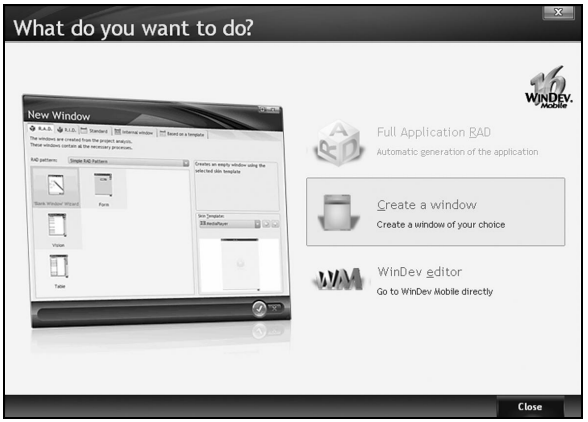
6. Choose "Smartphone". Go to the next screen. Display the next screens until you reach the "Charters - Style book" screen.

7. Choose the appearance of your application by selecting the "iStyle" style book for example.

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:



We are now going to create the window used to send the SMSs.

Creating the window for sending the SMSs

You are now going to create the following window:

Compared to the "WIN_StopWatch" window created in the previous lesson, you can see straightaway that the Smartphone windows are really small.

To create the window for entering the 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. All you have to do to create a window is click "Window".
2. Select "Wizard for blank windows".

   By default, this window will use the skin template selected during the project creation.
3. Validate. The wizard for creating a blank window starts.
4. Keep the "Maximized" mode and go to the next screen.
5. Check (if necessary) "With menu" and go to the next screen.
6. Enter the name and 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).

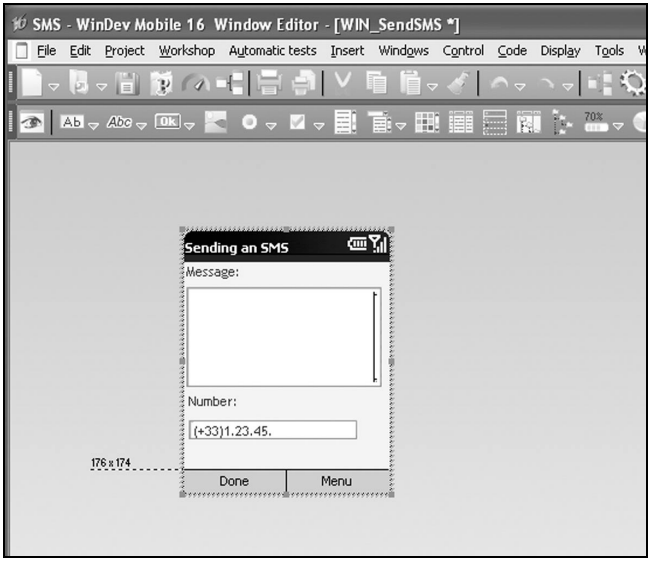
Note

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 the 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 Ab, then click the location where the control must be created in the window.
2. Right-click the control and select "Description".
3. Enter the name: "EDT_SMSMessage".

4. Enter the caption: "Message:".
5. Select the "Multi-line text" type.
6. Select the input size: "160 characters". Indeed, the maximum size for an SMS cannot exceed 160 characters.
7. Select the "Details" tab and select "Vertical scrollbar".

Note

Smartphone mode
You will notice that the "Smartphone mode" option is checked (sub-option of "Multi-line text"). This option is used to automatically manage the display of the control in a Smartphone.
For example, during the input in a multi-line 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 caption of the control is "overlapping".
9. Validate.

Note

The Smartphone keys enable you to enter the text of the message.

▶ To create the second edit control:
1. Click the arrow found on the right of the icon  . A window presenting the different types of edit controls is displayed. Click the "Phone: (+33)1.23.45.67.89" control and click the location where the control must be created in the window.
2. Right-click the control and select "Description".
3. Enter the name: "EDT_SMSNumber".
4. Enter the caption: "Number:".
5. Check 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 113 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 caption of the control 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. Click the "Content" tab and enter the caption of the option: "Re-send if not received".
5. Click the "Style" tab and choose a "No caption" style for example.
6. Validate.

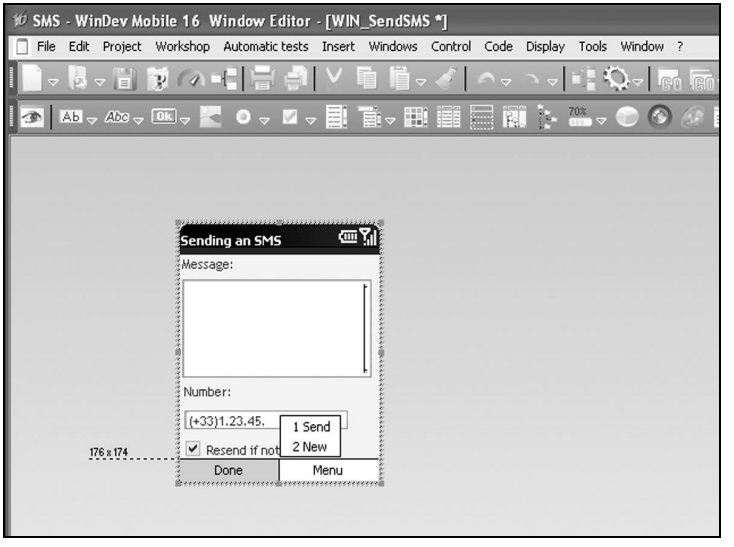
Note

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 113 for more details.

Describing the menu options

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



To add the sub-options:

1. Click the menu on 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:

```
SMS.Message = EDT_SMSMessage                    Initialize the SMS structure
SMS.Number = EDT_SMSNumber
SMS.Retry = CBOX_Resend

ResSend is boolean = SMSSend()                  Send the SMS

IF ResSend = False THEN                         Error occurred?
   Error(ErrorInfo(errMessage))
ELSE
   Info("SMS sent")
END
```

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

```
SMSReset()                                      Re-initialize the SMS structure

Reset()                                         Re-initialize the window controls
```

Note

When the 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.

On Smartphone, only two menus can be displayed simultaneously. 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.

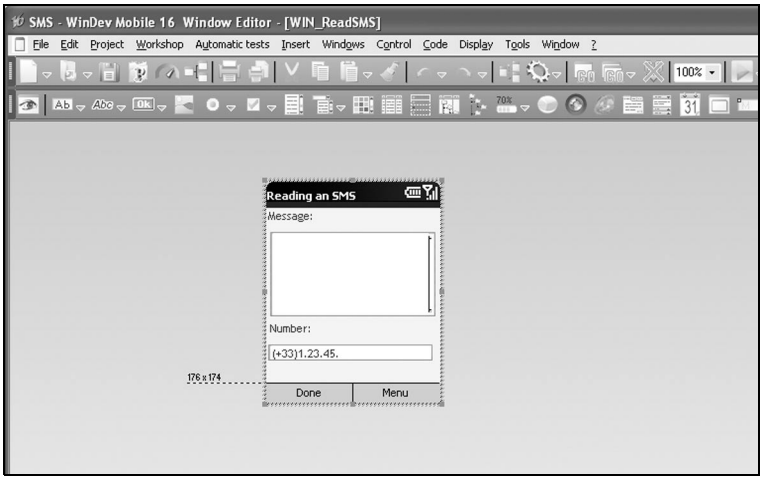
Creating the window for reading the SMSs

▶ To create the window:
1. Select "File .. New", hover the "Window" element and choose "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 during the project creation.
3. Validate. The wizard for creating a blank window starts.
4. Keep the "Maximized" mode and go to the next screen.
5. Check (if necessary) "With menu and toolbar" and go 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 the 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 are going to 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 explorer" pane).
  2. Select the "EDT_SMSMessage" and "EDT_SMSNumber" controls.
  Reminder: To select several controls at the same time, keep the [Ctrl] key down while selecting 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 opening the window:
  1. Right-click the window and select "Code".
  2. Enter the following code lines in the process for declaring the global variables:

```
Subscript is int
```

Declare the Subscript variable containing the subscript of the displayed SMS

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

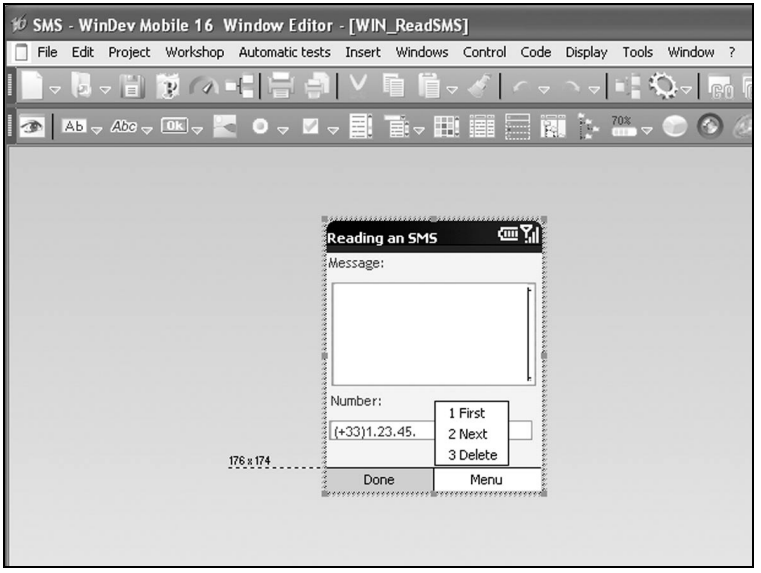
```
SMSFirst(smsSIM)

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

Positions on the first incoming SMS stored on the SIM card

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

Describing the menu options

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



- To add the sub-options:
  1. Click the menu on 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)

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

Positions on the first incoming SMS stored on the SIM card

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

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

```
SMSNext(smsSIM)

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

Positions on the next incoming SMS stored on the SIM card

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

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

```
SMSDelete(smsSIM, Subscript)
```

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", hover the "Window" element and choose "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 during the project creation.
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).

Create 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 Ok and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the caption: "Sending SMSs".
  4. Click the button named "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 Ok and drag the control into the window.
  2. Right-click the control and select "Description".
  3. Enter the caption: "Reading SMSs".
  4. Click the button named "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:
- test on the development computer (in simulation mode).
- test and debug on the Pocket PC connected to the development computer.
- 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.

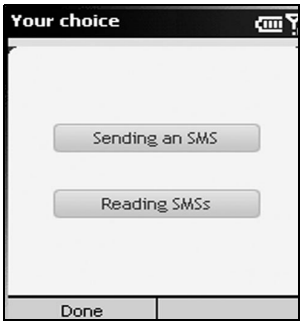
Therefore, the test of this application cannot be run on the development computer (in simulation mode).

Indeed, when running a test from the development computer, a WLanguage error occurs when using the menu options named "Send" or "New".

To run the test of this application, a device for managing the 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 the "Debug project (mobile device)" icon (or select "Project .. Test mode .. Debug the project (mobile device)").
  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 during this test. All you have to do is add a breakpoint into 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 the 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 the SMSs.

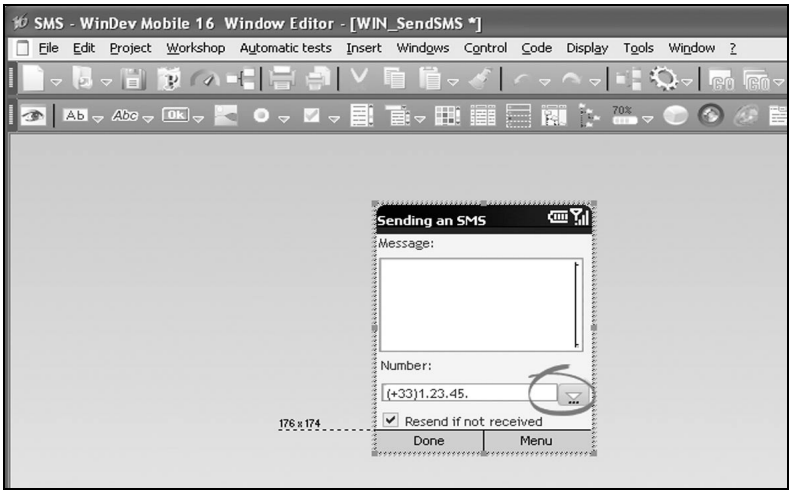
Improving the application: managing the SIM card

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 (which means in your personal directory).



Note

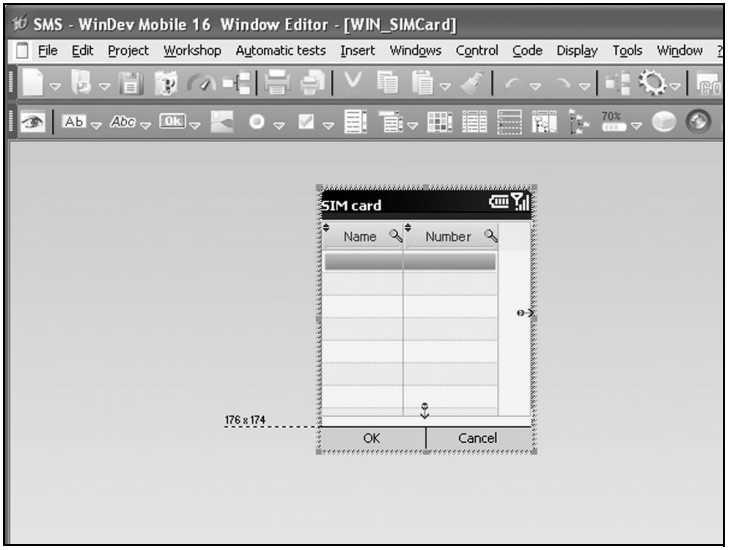
To press a button, give focus to the button (via the "joystick") and press the "joystick" to validate.
See “Input mode”, page 113 for more details.

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".
   - in the "Theme and Dimensions" section, indicate that the format of the image must be equal to "16 x 16".
   - select the requested image and validate.
   - validate the window asking for the number of states to use.
6. Validate.

Creating the information window

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



▶ To create the window:

1. Select "File .. New", hover the "Window" element and choose "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 during the project creation.

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 creating a table starts.

2. Keep "Fill the table by programming". Go to the next screen.

3. Keep "Vertical" for the orientation of the table. Go to the next screen.

4. Select "Read-only table". Go to the next screen

5. Specify the number of columns: "2" and enter the title of each column: "Name" and "Number".

Go to the next screen.

6. Enter the name: "TABLE_Directory".

7. Enter the title: "Directory:".

8. Validate (green button).

Resize the table if necessary.

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

```
Subscript is int
FOR Subscript = 1 _TO_ SIMNbContact()
   TableAdd(TABLE_Directory, …
                  SIMRead(Subscript))
END
TableSort(TABLE_Directory, "COL_Name")

TableSelectPlus(TABLE_Directory, 1)
```

Add each entry into the table

Sort the elements found in the table by name

Select the 1st row

▶ To define the menu on the left:

1. Select the menu on 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:

```
Close("", COL_Number)
```

Retrieve the number and close the window

▶ To define the menu on the right:

1. Select the menu on the right.

2. Right-click the menu and select "Option description".

3. Enter the caption: "Cancel".

4. Click  and enter the following code line:

```
Close("", "")
```

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:

```
EDT_SMSNumber = Open(WIN_SIMCard)
```

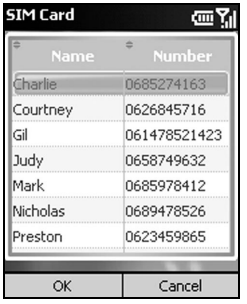
Open the "WIN_SIMCard" window
When this window is closed, the phone number is displayed in "EDT_SMSNumber"

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 the SMSs must be connected to the development computer.

Checking the management of the SIM card

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

LESSON 2.3. ANDROID DEVELOPMENT

This lesson will teach you the following concepts ...

- Configuring WinDev Mobile for the Android development
- Specific features of Android development
- Test of an application for Android
- Modifying an application for Android

Estimated time: 1h


EXAMPLE

The "Android Password.WPP" project corresponds to the full project with the answers to this lesson. To open this project, select "? .. Tutorial .. Android (answers)".

You can follow this lesson without opening the project.

Overview

WinDev Mobile allows you to develop applications for the Android operating system of Google. The principle for developing applications for this platform is identical to the principle for developing applications intended to be run on Windows Mobile. However, developing applications for Android presents some distinctive characteristics that will be presented in this lesson.

Note

WinDev Mobile generates applications compatible with all the versions of the Android operating system from version 1.5.

Configuring the Android SDK

To develop an application for the Android platform, the Android SDK must be installed on the development computer. This setup can be done before or after the setup of WinDev Mobile.

The WinDev Mobile projects must be configured to use the SDK. This configuration is performed:

- when creating the project.
  Note: The path of the specified SDK is stored for all the forthcoming projects.
- by selecting "Project .. Project description" and by clicking the "Platform" button found in the "Project" tab. The path of the SDK is indicated at the bottom of the screen.
  Note: The project must be an Android project or it must be in an Android configuration.

Note: See the online help for more details.

The project used in this lesson is named "AndroidPassword.WPP". This project is a password manager. We are going to modify it in order to discover the development for Android.

First, we are going to configure the SDK for the project found in this lesson :

- Open the "Android Password.WPP" project (select "the option "? .. Tutorial .. Android Password (Exercise)").
- Select "Project .. Project description". Click the "Platform" button.
- Configure the location of the Android SDK.
- Start the automatic detection of your Android device. Ignore this step if you have no Android device.

Databases in Android

The Android operating system is supplied with a system for database management that can be used by any application. This database is named SQLite.

When developing a project for Android, you can:

- configure the database connection in the data model editor to use SQLite
- declare the connection by programing as follows:

```
// In Android, the files are in SQLite
IF InAndroidMode() THEN
  // Describes the connection that will be used
  IF HDescribeConnection(cstConnectionName, "", "", ...
    "AndroidPassword", "", hAccessSQLiteAndroid, ...
    hOReadWrite) = False THEN
      Error("Unable to describe the connection", ...
        HErrorInfo(hErrMessage))
      EndProgram(True)
   END
   // Changes the connection of the application files
   IF HChangeConnection("*", cstConnectionName) = False THEN
     Error("Unable to change the connection", ...
       HErrorInfo(hErrMessage))
     EndProgram(True)
   END
END
```

Test of the application

Three solutions are proposed to run the test of the Android application:

- **Test in the WinDev simulator**: the WinDev simulator is used to easily run the test of a code section or algorithm. It can also be used to debug the application. However, it does not entirely reproduces the behavior of a real Android device and it cannot replace the other test modes. The data files used are HyperFileSQL data files.
- **Test on an Android device**: this is the most reliable test as the application is generated and run on a real device. This test mode requires an Android device. The data files used are in SQLite format.

- **Test in the emulator of Android SDK**: this mode is a compromise between the two previous modes. Its operating mode is similar to the operating mode of a device and no Android device is required. The data files used are in SQLite format.

▶ Run the test of the "Android Password" application in simulator mode ("Run the project test (WinDev simulator)") then in one of the other modes to see the differences of behavior.

▶ During the test, create a general password and add different passwords ("+" button). Select one of the passwords and display its characteristics (). The information is displayed by a simple dialog box.

▶ Stop the test. Go back to the editor and display the "WIN_PassPpal" window (by double-clicking its name in the project explorer for example).

▶ Display the code of the consultation button. The code used is as follows:

```
// Displays the user name
// and the associated password (in plain letters)
Info("User name:" + Password.UserName, ...
     "Password: " + Password.Password)
```

We are going to improve this operating mode. We are going to create a window used to display the password. This window will be automatically closed after 10 seconds in order to preserve the confidentiality of the password.

Adding a new window for displaying the passwords

Creating the window and its controls

▶ To add a new window:

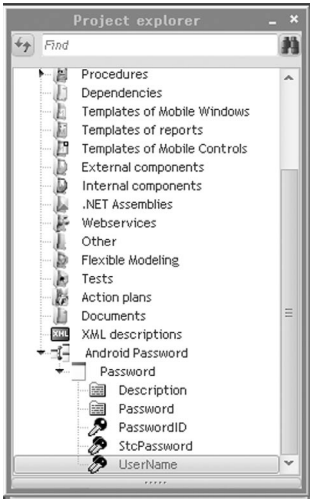
1. Select "File .. New" and click "Window".

2. In the wizard, select "Blank" and validate. A blank window appears in the editor.

3. Save the window as "WIN_ViewPassword" ("File .. Save").

▶ We are now going to create two controls for displaying the information about the passwords.

1. In the project explorer, expand the analysis branch and the description of the "Password" file.



2. Select the "UserName" and "Password" items and perform a Drag/Drop in the window editor. Two edit controls linked to the items are automatically created.

▶ To improve the interface of the created controls :

1. Select "Choose a style" from the popup menu of each control. Select the "EDT_Internal" style and validate.

2. Resize the controls so that they occupy the width of the window.

3. Select "Anchoring" from the popup menu of each control to anchor the controls in width.

Therefore, if the orientation of the Android device changes, the controls will be automatically adjusted to the new window size.

4. Save the window.

Displaying the data

This window will be used to display the data coming from the Password file. The controls are already linked to the file items. All you have to do is display the data in the controls. This operation is performed in the initialization code of the window by **FileToScreen**.

▶ To display the data in the window:

1. Click in the window and display the popup menu of the window (right mouse click). Select "Code". The code editor is displayed.

2. Enter the following code in the "Initialization process of WIN_ViewPassword" :

```
// Displays the user name
// and the associated password (in clear)
FileToScreen()
```

In this code, **FileToScreen** is used to display the content of the linked items in the edit controls.

3. Close the code editor.

Adding a closing button

Our window is nearly created. However, the user must be able to easily close this window. The window will be automatically closed if it is not closed by the user.

▶ We are now going to add a button used to close the window.

1. Click the arrow found on the left of the icon Ok. A window presenting the different types of buttons is displayed.

2. Click the "Close" button and click the location where the control must be created in the window.

3. Select "Control .. Alignment .. Center horizontally" to center the control in the window.

4. Select "Anchoring" from the popup menu of the control to anchor the controls in width.

▶ A timer will be used to manage the automatic closing of the window after 10 seconds. This timer will be started when opening the window and it will trigger the closing button after 10 seconds.

1. In the project explorer, select the "WIN_ViewPassword" window. Display the sub-elements of the window.

2. Position on the "Local procedures" branch and select "New local procedure" from the popup menu. This new local procedure is named "AutoClosing". Validate.

3. Enter the following code in the code editor:

```
PROCEDURE AutoClosing()

Close()
```

4. Close the code editor.

5. Display the code of the window ("Code" from the popup menu of the window.

6. Add the code for triggering the trigger in the "Initialization code of WIN_ViewPassword".

This process must now contain the following code:

```
// Displays the user name
// and the associated password (in clear)
FileToScreen()
// Starts the timer for automatic closing
TimerSys("AutoClosing",10*100)
```

Displaying the window for managing the passwords

Our window for managing the passwords is now created. We must now branch this window instead of the former display code by using **Info**.

▶ To do so:

1. Open the "WIN_PalPassword" window (double-click its name in the project explorer for example).

2. Select the "BTN_VIEW" button () and display the description of this control ("Description" from the popup menu).

3. Choose the preset action named "Open the WIN_ViewPassword window" and validate.

4. Display the code of the button ("Code" from the popup menu).

5. Put the code of the process in comment: select the code and press [CTRL]+ /.

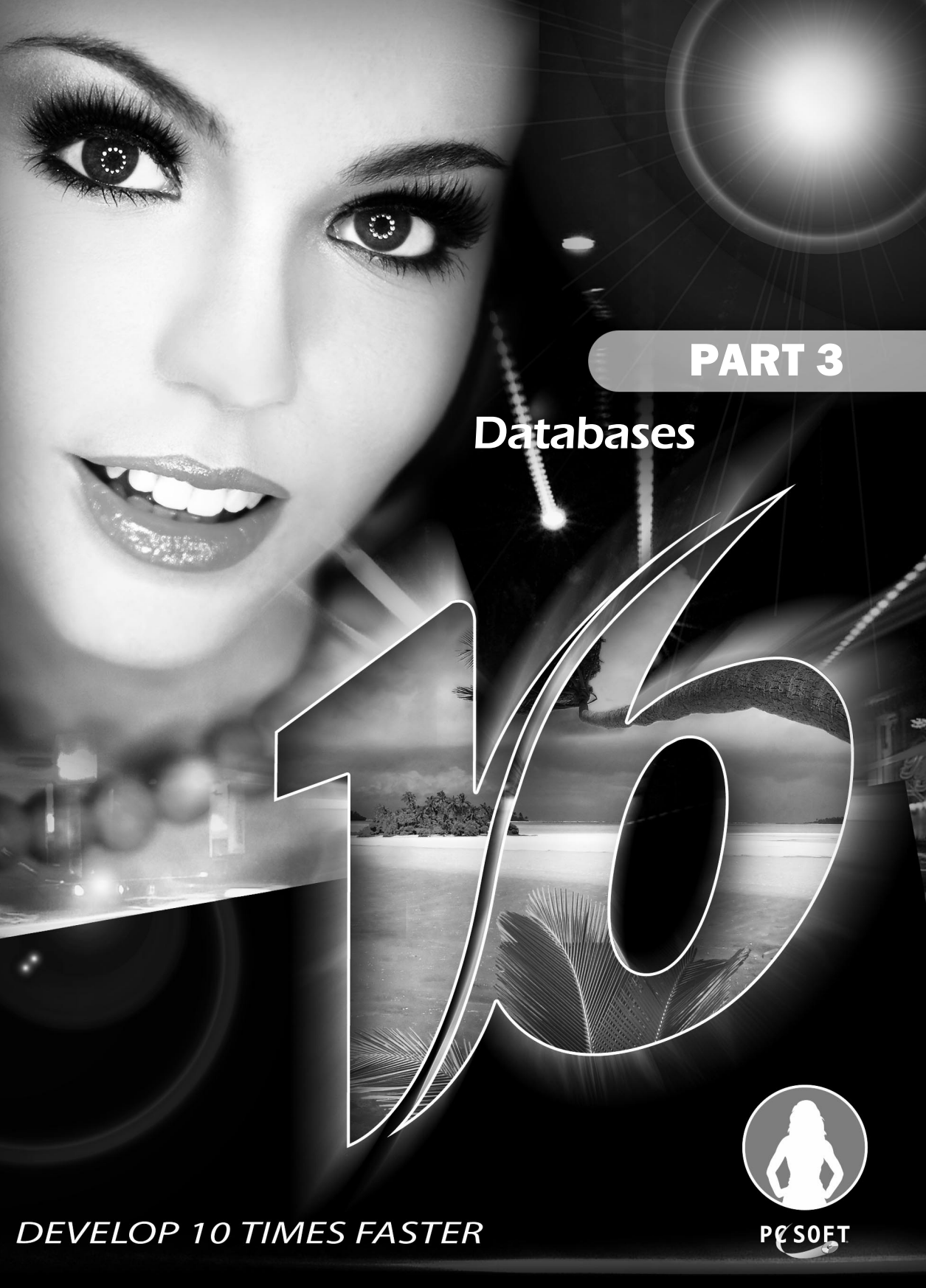
Test of the application

The test of the application can now be run. If the different operations have not been performed, open the corrected project via "? .. Tutorial .. Android (answers)".

▶ Run the test of the "Android Password" application in simulator mode for example ("Run the project test (WinDev simulator)").

1. Select a password that was previously entered.

2. Display its characteristics (). The information is displayed in the window that was just created.


PART 3
Databases
10
DEVELOP 10 TIMES FASTER
PC SOFT

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 some data. The format of these databases can be:

- HyperFileSQL 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.
- AS/400, the AS/400 database for Pocket PC.
- SQLite, database that can be used in Android.
- ...

HyperFileSQL Mobile

The HyperFileSQL 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 HyperFileSQL format of WinDev and WebDev (".WDD" file, data files, ...).

However, since the available size on a Pocket PC is small and the operating system of the Pocket PC is limited, the following features are not supported by HyperFileSQL 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 HyperFileSQL 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, ...). The same analysis can be used in a WinDev Mobile application and in a standard WinDev application.

The use of the HyperFileSQL Mobile format allows you to:

- access the records quickly,
- optimize the search time,
- handle some large databases,
- synchronize the HyperFileSQL Mobile files found on a Pocket PC with the HyperFileSQL files found on a PC,
- ...

In summary, most of the features of HyperFileSQL Mobile are available in WinDev Mobile (file link, queries, filters, views, ...).

Example

The "Pocket Notes", "Pocket Attendance" and "Pocket Telephony" examples (supplied with WinDev Mobile) handle the HyperFileSQL Mobile files.

These examples are accessible from the "Wizards, Examples and Components" pane.

Test

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

See “Interactions between applications” page 105 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", ...
- **the other CEDB databases** (called customizable databases), that correspond to the Access databases (".MDB" file) previously exported from a PC.

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

A CEDB database can be handled:

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

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

Note

Caution: From Windows Mobile 5, the access to a standard database (tasks, contacts, appointments, ...) is no longer possible from a Windows application (and therefore from a standard WinDev application).

The CDB databases are accessible from the PC, from the simulator, from the Pocket PC.

The standard database can only be accessed from the Pocket PC.

Note

Caution: The structure of the CEDB databases is not intended to process a large amount of data. Therefore, we advise you to use the HyperFileSQL Mobile databases. Furthermore, HyperFileSQL Mobile enables you to benefit from all the features available in WinDev Mobile (RAD, file link, ...).

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**.

AS/400

This database format is accessible via a Native Access. To use this Native Access, an additional module is required in addition to WinDev Mobile.

Contact PC SOFT Sales Department for more details.

SQLite

The "SQLite" database is the default database found on the Android operating system. The native access to this database is included in WinDev Mobile (for the Android projects only).

LESSON 3.2. HYPERFILESQL MOBILE FILES

This lesson will teach you the following concepts ...

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

Estimated time: 20 min

EXAMPLE

The "Product Catalog.WPP" project corresponds to the full project with the answers to this lesson. To open this project, select "? .. Tutorial .. HyperFileSQL Mobile (answer)".

You can follow this lesson without opening the project.

Overview

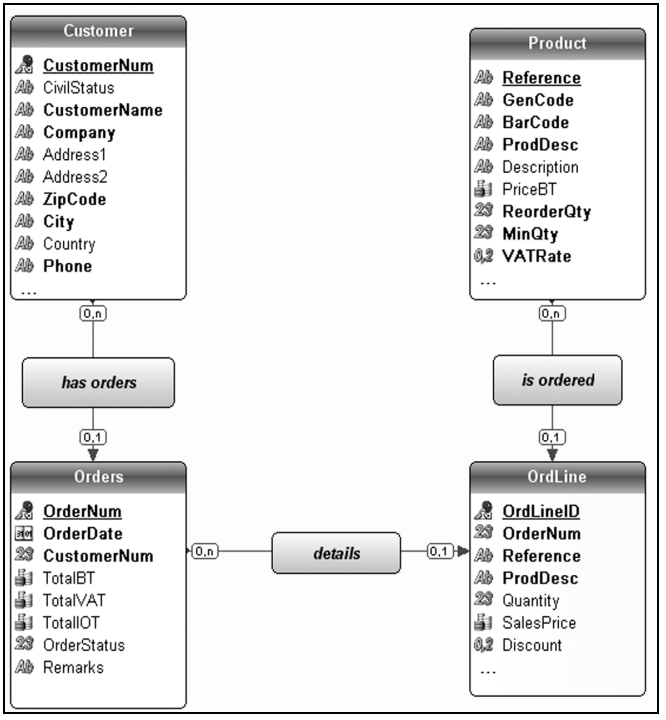
The format of the HyperFileSQL Mobile database was presented in the previous lesson.
Furthermore, you are already familiar with the main characteristics of this format.
This lesson explains how to generate a full application by RAD for Pocket PC from a HyperFileSQL Mobile analysis.

Generating a full application

An existing project will be used to generate this application.

▶ Open the "Product Catalog.WPP" project (select "? .. Tutorial .. HyperFileSQL Mobile (exercise)").

This project is associated with the following analysis:



Some simple files have been deliberately chosen for this lesson:

- a CUSTOMER file
- an ORDERS file
- an ORDLINE file
- a PRODUCT file

We will now create the windows and the reports of this project.

To generate the full application:

1. Select "Workshop .. Full Application RAD". The wizard of the RAD module starts.

2. All the analysis files will be used for the generation. Select all the files and go to the next screen.

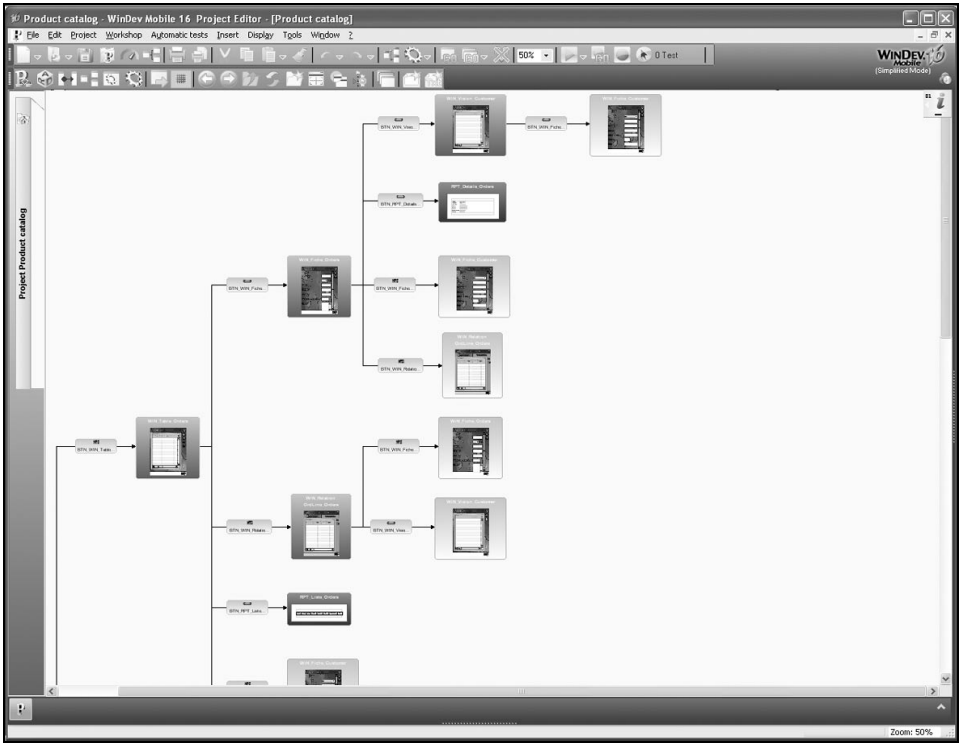
3. The entry points for the application (proposed by the menus) will be:

- Customer
- Orders
- Product

4. Go to the next screen.

5. Validate. The RAD generation starts.

The project graph is automatically drawn once the application is generated. To display it, select "Project .. Graph of the project". This full application is ready to 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.

Note

GUI errors affecting reports

Some GUI errors affecting reports may occur during the RAD generation. These errors are used to draw the attention of the developer to the tables that must be modified in order to get a legible report.

Indeed, the RAD automatically generates some optimized reports. However, if the number of items is quite important, the captions may have to be modified.

You will find, in the answers of the example, some modified reports the no longer trigger any GUI errors.

The test of this application can be run:

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

Note

RID (Rapid Interface Design)

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

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

Managing data files

Generated data files

During the call to ***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" icon), these files are created on the development **PC** (in the directory of the executable).

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

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

Note: On a Pocket PC, a specific format must be used for the paths of the files. See “Managing directories in Windows for Pocket PC” page 98 for more details.

Copying data files

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

Synchronizing data files

WinDev Mobile allows you to:

- synchronize data files found on the PC with the data files found on the Pocket PC.
- synchronize 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 HyperFileSQL Mobile data file found on the PC when developing an application.

Pocket Map is used to view and handle a HyperFileSQL 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.


PART 4
Specific features
of Pocket PC
10
DEVELOP 10 TIMES FASTER
PC SOFT

LESSON 4.1. SPECIFIC FORMATS

This lesson will teach you the following concepts ...

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

Estimated time: 20 min

Managing character strings

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

On the **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 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.

GO

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 the ANSI strings into UNICODE strings (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 some 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 the external files

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

- before writing a character string into an external file:

	Format of the string to write	**External file in ANSI format** (*fOpen*)	**External file in UNICODE format** (*fOpen* **associated with the** *foUnicode* **constant)**
fWrite	ANSI	No conversion is required	Conversion of the string before the write operation (***AnsiToUnicode***)
	UNICODE	Conversion of the string before the write operation (***UnicodeToAnsi***)	No conversion is required
fWriteLine	ANSI	No conversion is required	Automatic conversion of the string before the write operation
	UNICODE	Automatic conversion of the string before the write operation	No conversion is required

- after reading a character string in an external file:

	Current computer	**External file in ANSI format** (*fOpen*)	**External file in UNICODE format** (*fOpen* **associated with the** *foUnicode* **constant)**
fRead	PC running Windows	Read operation in ANSI format Possible conversion to UNICODE with ***AnsiToUnicode***	Read operation in UNICODE format Possible conversion to ANSI with ***UnicodeToAnsi***
	Pocket PC		
fReadLine	PC running Windows	Read operation in ANSI format Possible conversion to UNICODE with ***AnsiToUnicode***	
	Pocket PC	Read operation in UNICODE format Possible conversion to ANSI with ***UnicodeToAnsi***	

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 (*sWrite* or *SocketWrite*)	Read operation (*sRead* or *SocketRead*)	
		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 (***UnicodeToAnsi***)
UNICODE (Pocket PC running Windows CE for instance)	The character string / the message will be in UNICODE format	Conversion required (***AnsiToUnicode***)	No conversion is required

Managing directories in Windows for Pocket PC

The management of directories differs on a PC and on a Pocket PC.

On a **PC**, the paths have the following format: "C:\MyDocuments\MyFile.txt". On a **Pocket PC**, 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***, ...), 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")
// file in the runtime directory of the application
```

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 example) are not available in WinDev Mobile.

File picker

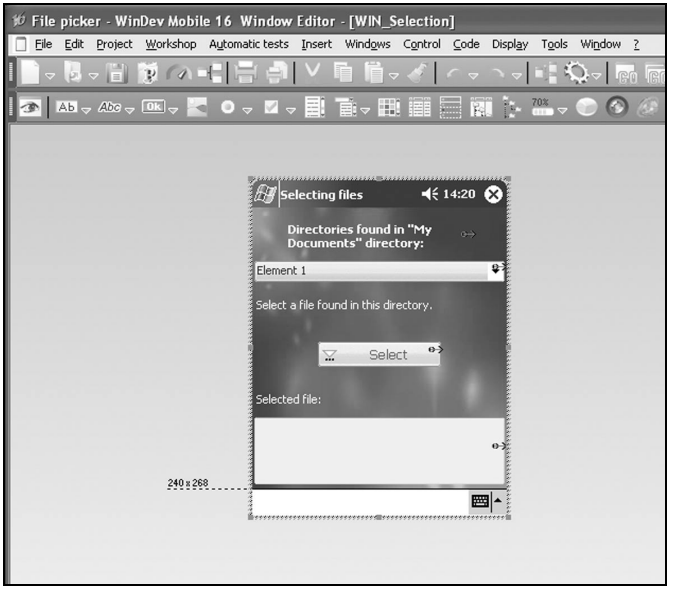
In Pocket PC, the file picker is used to select the files found:

- 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 are now going to use the file picker:

- Open the "File Picker.WPP" project ("? .. Tutorial .. File picker").
- Open the "WIN_Selection.WPW" window (from the project explorer).
  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

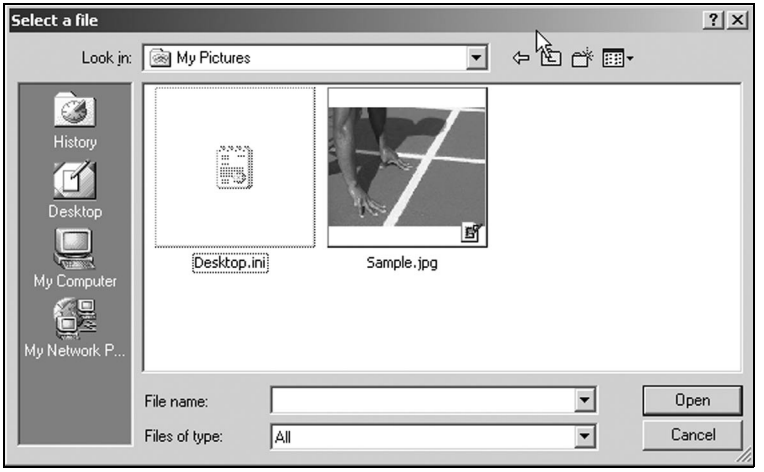
GO Test

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

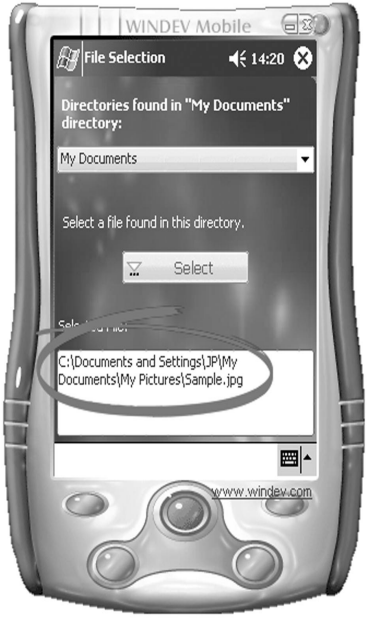
We are going to run the test of the "WIN_Selection" window in simulation mode on the development computer:

- Click the "GO" icon (or press the [F9] key). The test of the window is run. The window is displayed in a simulator. WinDev Mobile proposes to run the test in simulation mode: accept (answer "Yes").

- Select one of the directories found in the "My Documents" directory (from the combo box) and click the "Select" button.
  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:



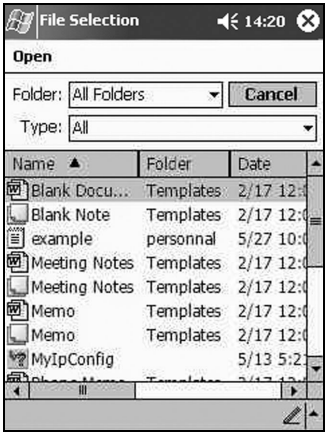
- Close this window. Let's now run the test of the window on the Pocket PC.

Test on the Pocket PC directly

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 the "Run the project test (mobile device)" icon (or select "Project .. Test mode .. Run the project test (mobile device)"). The editor proposes to generate the executable. Accept and follow the different steps of the wizard. At the end of the wizard, the widow is automatically started on the Pocket PC:

- Select one of the directories found in the "My Documents" directory (from the combo box) and click the "Select a file found in this directory" link.
  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 the "My documents" directory 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 by programming a file found on a storage card,all you have to do is use the following format: "\<CardName>\<DirectoryName>\<FileName>.<Extension>".

For example:

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

Note

When a hard reset is performed on the Pocket PC, loss of data occurs 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 57 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, ...

Platform on which the project is run

To find out the platform on which 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

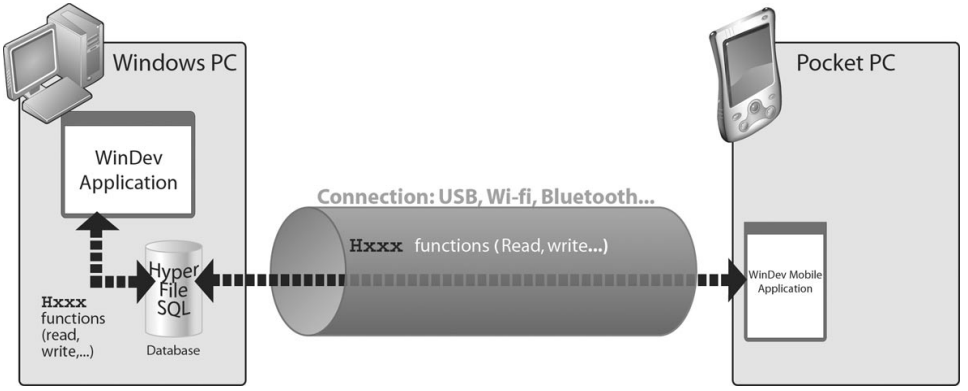
Sharing data between two applications

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.

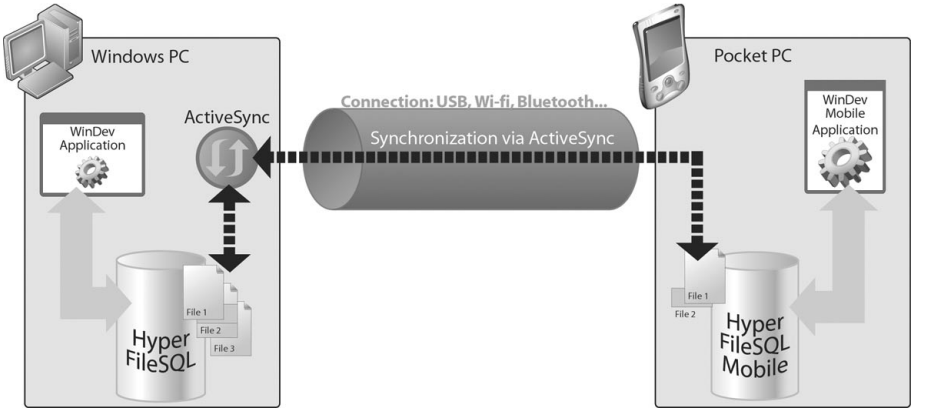
When two applications share the same data, the data files can be managed in two different ways:

- ***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, ... ***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 onto the Pocket PC:***
  All the data files (or some of them) are previously copied onto each Pocket PC. Each application handles its own files. To take into account the modifications performed in 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 HyperFileSQL functions.

EXAMPLE

Example

Databases in HyperFileSQL 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 onto 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 used are in HyperFileSQL format, all the Pocket PCs must be connected one by one to the PC. The automatic HyperFileSQL synchronization via ActiveSync takes everything in charge. See 'Synchronizing the data files' (page 93) for more details.

If the data files used are not in HyperFileSQL 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.

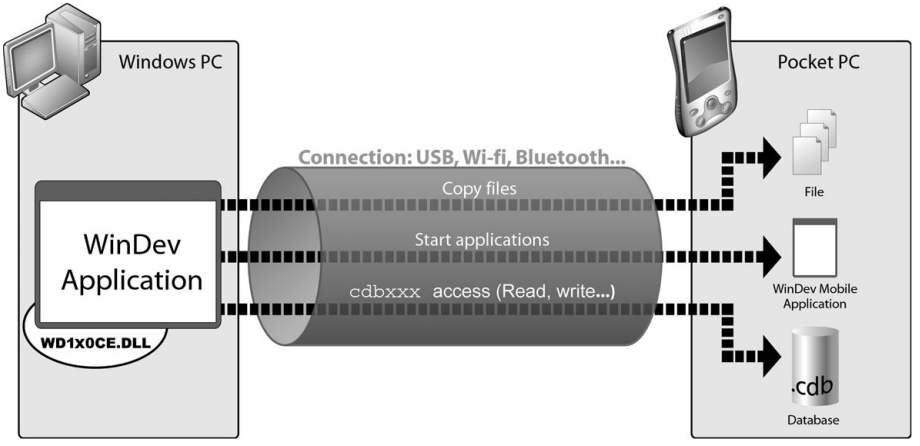
EXAMPLE

Example

Databases in HyperFileSQL format (on the Pocket PC and on the PC)
The "Sending SMSs", "Managing lists of purchases" and "Stocks" examples, supplied with WinDev Mobile, include a project that can be used on Pocket PC and a project that can be used on PC. These examples present the synchronization of the data entered in the two projects.

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, ...).
- retrieve some information about the Pocket PC (find out the charge level of the batteries, the type of processor, the Windows version used, ...).
- manage the registry of Pocket PC (create or delete a key, modify the value of a key, check the existence of a key, ...).

LESSON 4.3. SHARING SOME WINDEV ELEMENTS

This lesson will teach you the following concepts ...

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

Estimated time: 10 min

EXAMPLE

The "Pocket PC Import .WPP" project corresponds to the full project with the answers to this lesson. 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, ...) can be used in a 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, ...

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" project: select "? .. Tutorial .. Pocket PC import (exercise)".

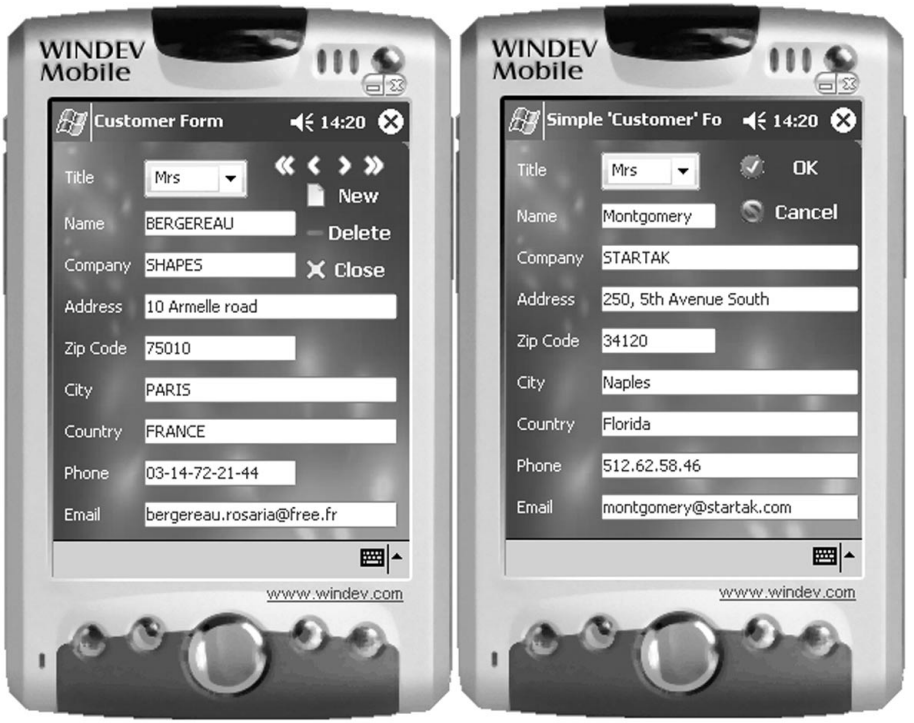
2. Select "File .. Import .. WinDev elements". The import wizard starts.

3. Select the windows named "WIN_Form_Customer.WDW" and "WIN_Simple_Form_Customer.WDW" (found in the "Tutorial\Exercises\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 (by modifying the anchors of the controls if necessary).

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 following files: "WIN_Form_Customer.WPW" and "WIN_Simple_Form_Customer.WPW".



Operations performed during the import

When importing a window:

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

Sharing some code

WinDev Mobile proposes several methods for sharing some code:

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

Input of multi-product code

The code editor is used to enter the equivalent of the code that will be run in WinDev Mobile, in standard WinDev or in WebDev. Via the "Conditional Target-Code" mechanism, this code is entered at the same location in the code editor. Some 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 function

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

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

InPocketMode will be used at run time to prevent these functions from running 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 some information on a Pocket PC

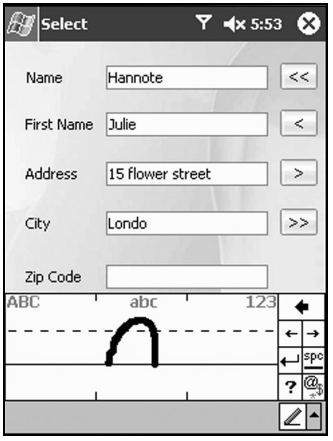
To allow the users of your applications to enter data, the keyboard of the Pocket PC must be used (also called SIP for Software Input Panel).

This keyboard allows you to:

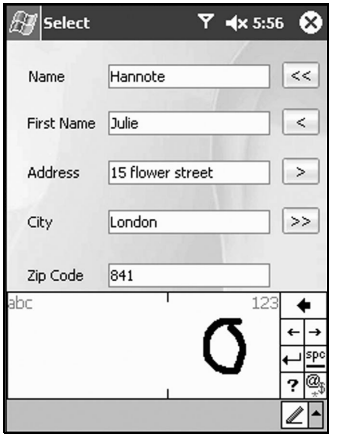
- display a miniature keyboard at the bottom of the screen. The user clicks this keyboard with the stylus to enter information.
  For example:

- 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:



Note: Other types of keyboards are available.

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 ( for example) enables you to manage the use of the keyboard.

Note

The "Activate the keyboard in edit" option ("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 some information on a Smartphone

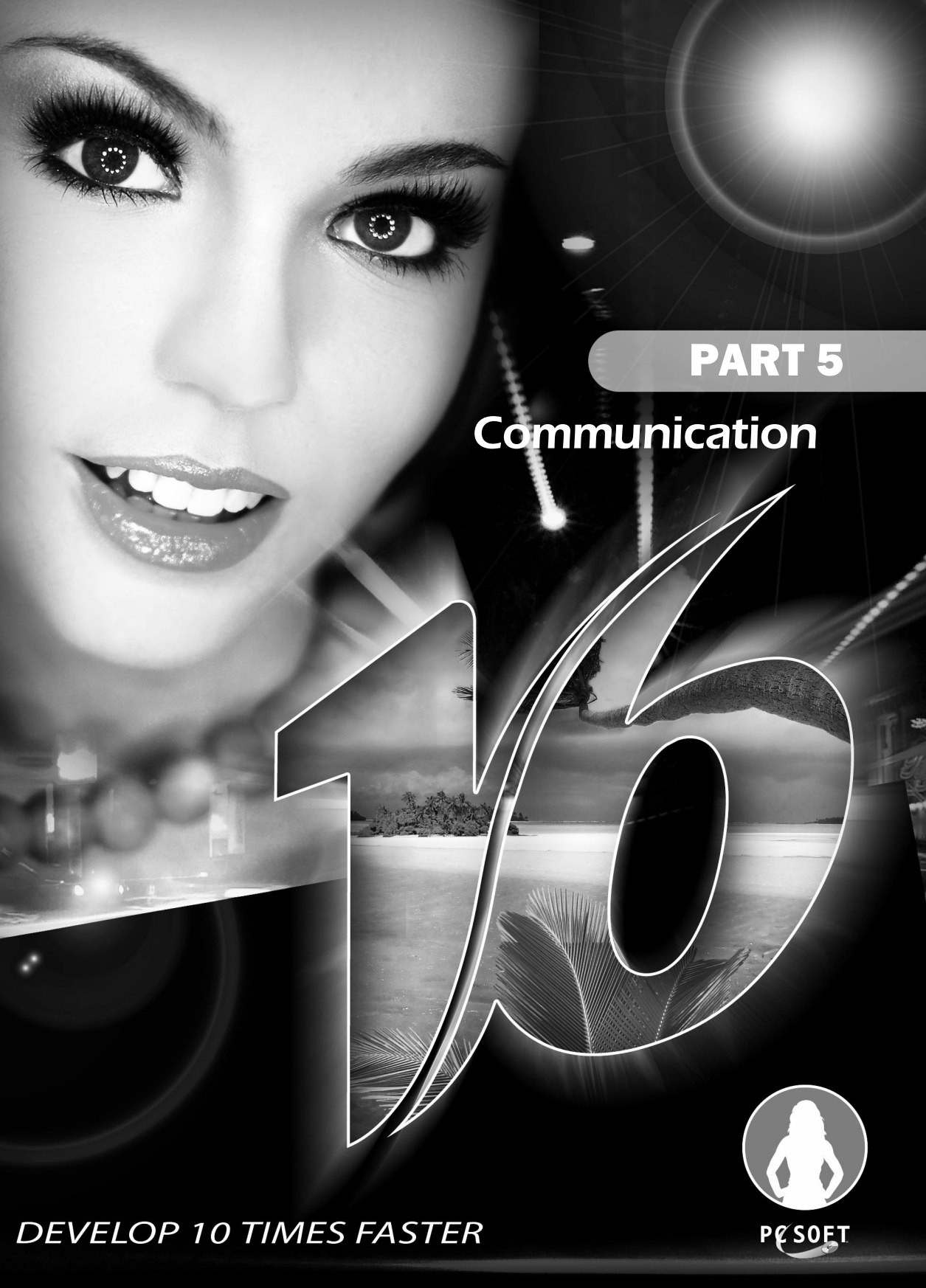
Several default input modes are available for a Smartphone:

- 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 some 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 "43556" 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, "2355" can be used to write "bell" and "cell".

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.


PART 5
Communication
10
DEVELOP 10 TIMES FASTER
PC SOFT

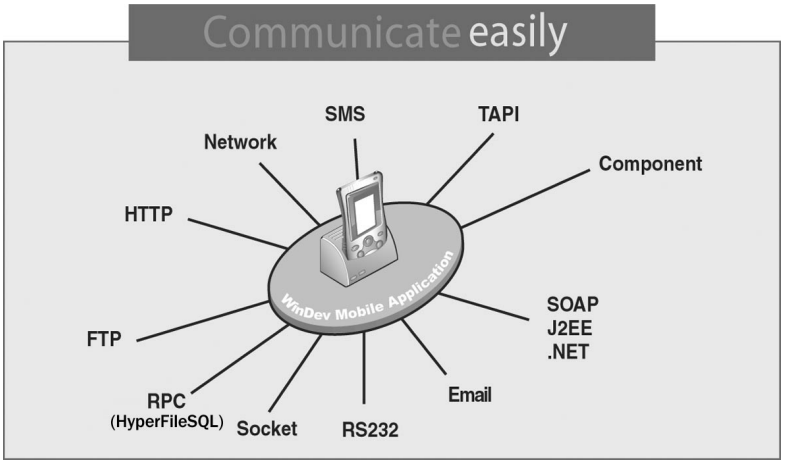
LESSON 5.1. INTRODUCTION

This lesson will teach you the following concepts ...

- **Communication with WinDev Mobile.**

Estimated time: 10 min

Communicating with WinDev Mobile



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

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

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

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

We shall not present all the features proposed by WinDev Mobile. We shall only take a look at the management of emails (see the lesson page 124).

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

Transferring files by FTP

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

Several thousands of 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

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 HyperFileSQL Mobile)

Remote access enables you to consult a HyperFileSQL 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. Therefore, the HyperFileSQL library (wp160hf.dll) and the analysis description (".WDD" file) must be found on each Pocket PC.

The data exchanges will be performed according to the RPC protocol (Remote Procedure Call) by using the functions of the communication library wd160com.dll (found on the server) and wp160com.dll (found on the client computer, which means on the Pocket PC).

These data exchanges can be performed:

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

Note

Differences compared to standard WinDev

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

Managing the sockets

Several functions can be used to perform an advanced management of 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***).

The exchange of data between two computers can be performed:

- by Wi-Fi.
- via Infrared.
- by ActiveSync.
- by GPRS.
- by 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 96 for more details.

Example

The example for socket management supplied with WinDev Mobile includes a project that can be used on Pocket PC ("Pocket Socket UNICODE") and a project that can be used on PC (PC Socket Unicode). 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 allow you to run some 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, 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 to the ANSI format 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 allows you to easily send and read some SMSs (Short Message Service) via the SMSxxx functions of WLanguage.

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

See "Application on Smartphone", page 57 for more details.

Summary

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

	Remote access (RPC on HyperFileSQL)	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 access by phone (GSM type)					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 the 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
- ...

WinDev Mobile proposes several methods for managing emails:

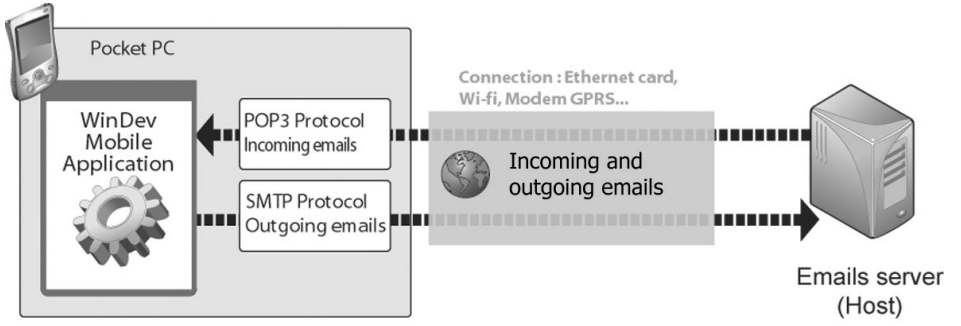
- **the POP3/SMTP protocol** (most common method): this protocol, used to manage emails, is recognized by all the service providers. This protocol enables you to directly communicate with the server, available at your ISP.
- **the IMAP protocol**: this protocol for receiving emails stores the emails on the server in order to consult them from different messaging clients or webmail.
- **the "CEMAPI" API**: this management mode of emails uses Pocket Outlook to send and receive emails.

Managing the emails via the POP3/SMTP/IMAP protocol

The POP3, IMAP 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 and IMAP protocols are used to receive emails.

The SMTP protocol is used to send emails.



Note

The method for sending and receiving emails via the POP3/IMAP/SMTP protocol is identical in WinDev Mobile and 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 the emails via "CEMAPI"

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

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

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

User account

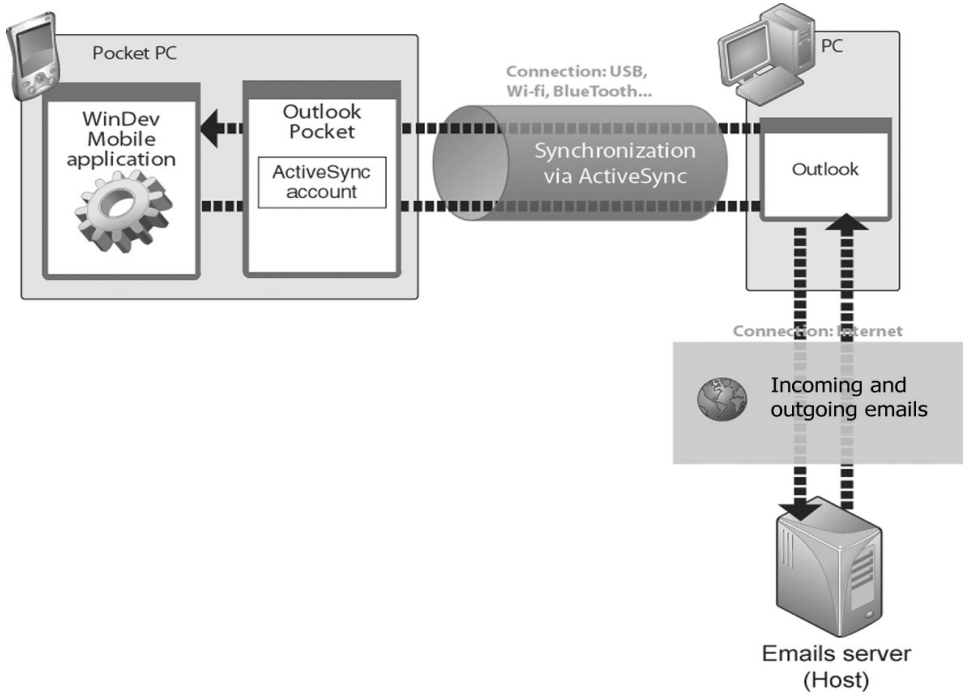
A user account defined in Pocket Outlook is required to manage the emails via CEMAPI.

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

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

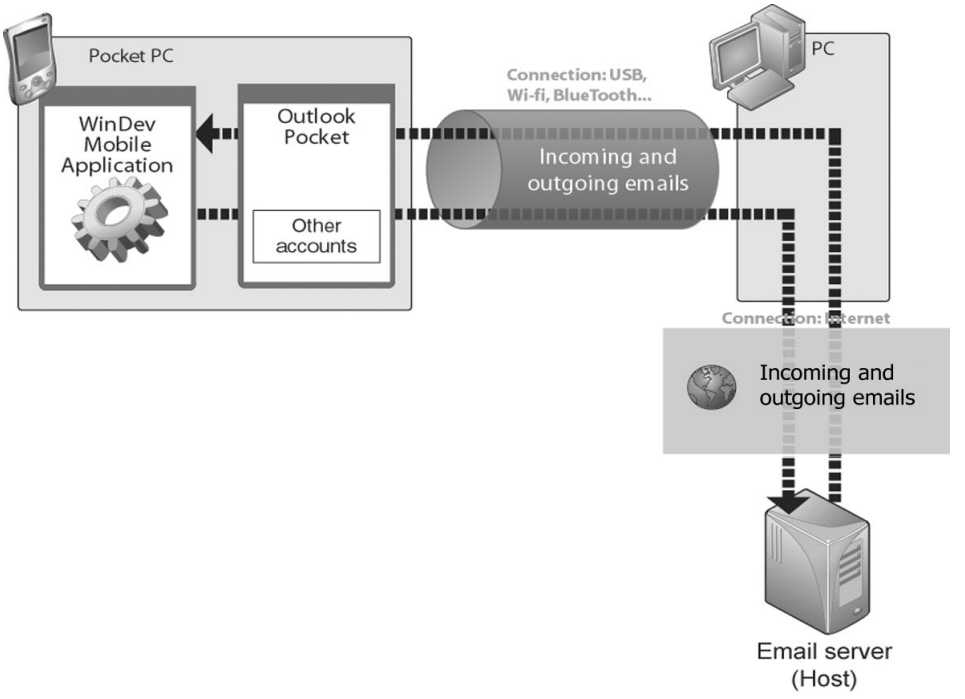
"ActiveSync" user account

The "ActiveSync" user account is used by default.



Other user account

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

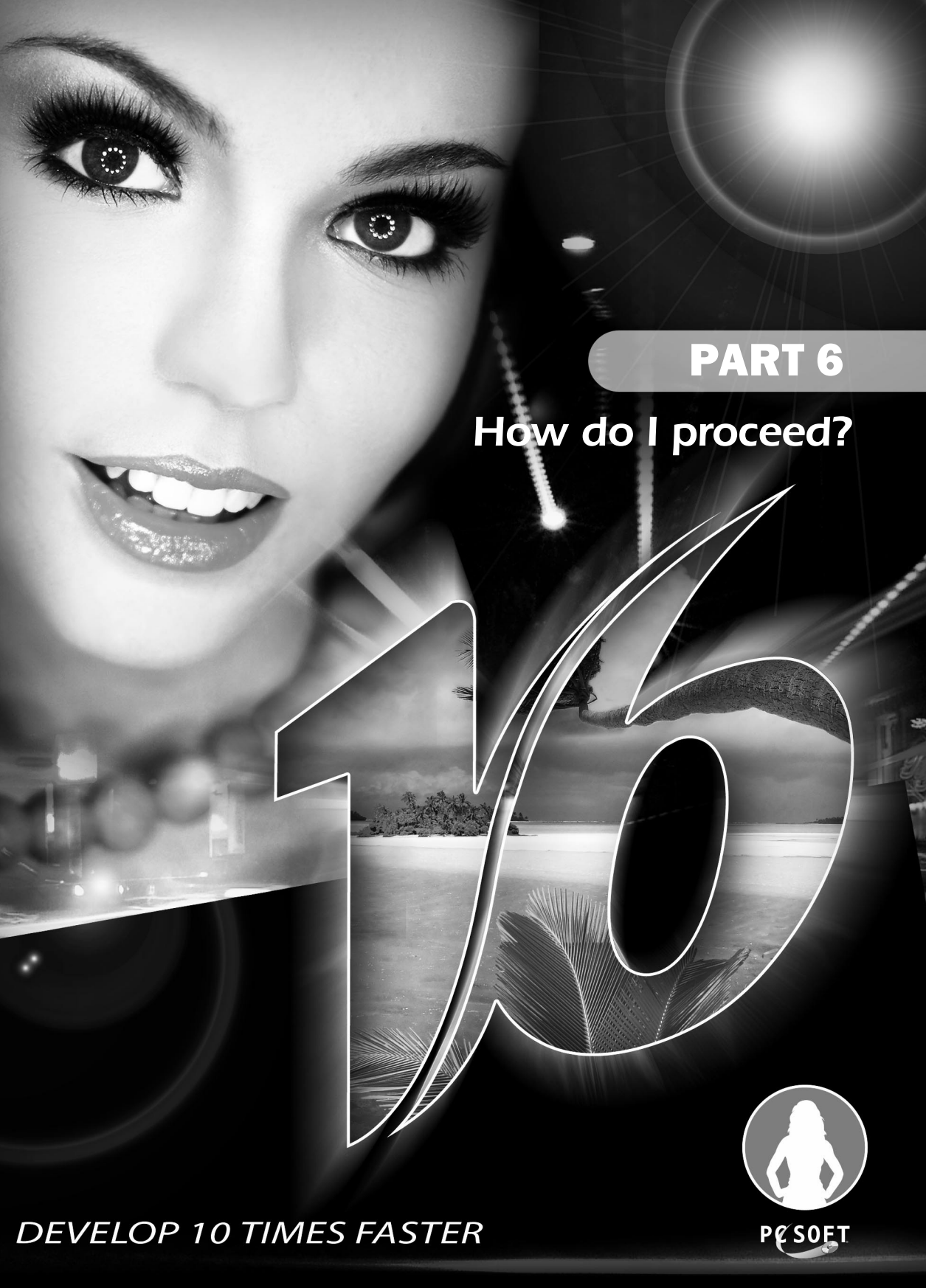


Note

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

Only differences:

- ***"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.


PART 6
How do I proceed?
10
DEVELOP 10 TIMES FASTER
PC SOFT

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 the Pocket PC.
- Non-maximized window: A non-maximized window can be resized by the user and it can 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.

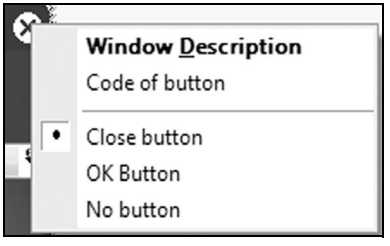
Note

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 Smartphone or Android platform.

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

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

To display the keyboard on the Pocket PC:

- select "Enable the keyboard in input" ("Details" tab in 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.

The [Page Up] and [Page Down] keys allow you to go from one plane to another in the editor. The number 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 will be 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 .. Panes .. Other panes .. 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 static 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 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.

A progress bar is used to display the progress of a process.
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 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 and 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

We advise you to pass parameters to a window rather than declaring some global variables in the project.

Question **How do I transform a check box into a radio button?**

"Control .. Swap .. 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 looper.

Question **How do I retrieve the parameters passed by 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

Passing parameters by command line to your project can be simulated 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 the mouse.
2. In the menu of the editor, select "Control .. Groups .. Associate the selection". Click "New".
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 the controls:
- the rulers.
- the interface checker.
- the real-time interface checker.
- the alignment options.

To enable the rulers, press [CTRL] and [R] simultaneously. Some "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 some 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 practicing 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 options ("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 now have the ability to choose an image and to 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?**

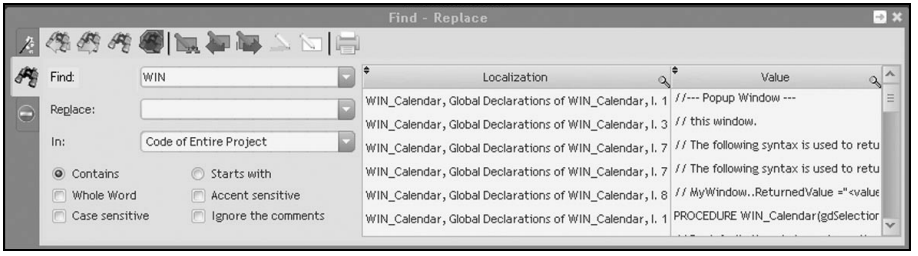
To print the full project documentation from the project graph (or from the dashboard), select "Project .. Print the project documentation".

Question **How do I create a skin template?**

This operation includes 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?**

The functions for performing searches or replacements in the code can be accessed via "Edit .. Find" and "Edit .. Replace" or in the "Find - Replace" pane ( ) :



The search can be accessed at any time by pressing [CTRL]+[F].

Question **How do I find out the list of project elements?**

The elements found in a project are: windows, reports, queries, ...

- 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.

The [F5] key 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?**

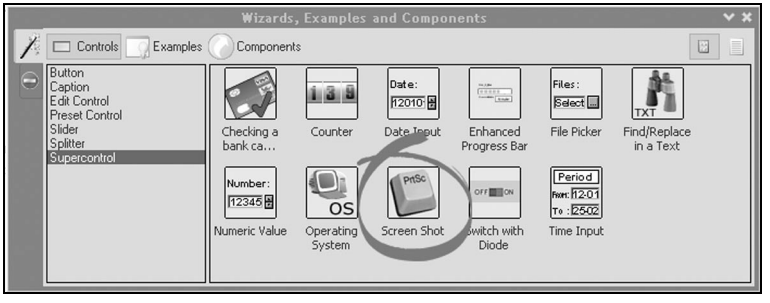
The environment parameters of the editor (directory, language, login, ...) can be modified from the options of WinDev Mobile ("Tools .. Options .. General options of WinDev Mobile").

The parameters for displaying the windows or the reports (click, magnetism, ...) 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 code!



To perform a screen shot of the Pocket PC (or Smartphone), use "WDCapture" ("Tools .. WDCapture - Screen shot").

Question **How do I read and write into an .INI file?**

INIRead and ***INIWrite*** are used to read and to write in an 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 following image formats: BMP, JPEG, GIF, PNG 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 the data. The name of these functions starts with "Zip".

See the "Pocket Zip" example supplied with WinDev Mobile or 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 for 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 that is 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 delete.

Test files and windows, useless images, ... It's time to clean up!

A WinDev Mobile tool is available 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 .. Clean 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, ...).

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 some 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
   fClose(FileNumber)
END
```

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.

HyperFileSQL Mobile

Question **Is the format of the HyperFileSQL files compatible with the format of the HyperFileSQL Mobile files?**

Yes, the HyperFileSQL format and the HyperFileSQL Mobile format are compatible. Their format is identical. The HyperFileSQL data files and the HyperFileSQL Mobile data files can be used in WinDev Mobile and in WinDev.

However, the available size on a Pocket PC being restricted and the operating system of the Pocket PC being limited, the following features are not supported by HyperFileSQL Mobile:

- the transactions.
- the log process.
- the HyperFileSQL replication.
- 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 is used 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:
  - the "NULL supported" option 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.
  - the "Default to NULL" option 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:

..NULL	Enables you to: • define the NULL value as the default value for a file item during its dynamic description • associate (or not) the NULL value with a file item

..NullSupported	Enables you to: • define the management mode of the NULL value for a file during its dynamic description. • find out the management mode of the NULL value for a file

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", ...
                "Duplicate error")
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", ...
              "Integrity error")
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"])
```

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 performed in 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:

- the "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 the "Selection condition" button 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. Printing 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 print job performed on a Pocket PC depends on the features of the printer used (management of images and lines, management of fonts, ...).

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 magnetic card reader?**

Proceed exactly the same way as for bar code readers.

Question **How do I manage a serial port?**

- Use ***sOpen***, ***sRead***, ***sWrite***, ***sClose***, ...

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***, ...

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***, ...

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

CONCLUSION

The tutorial is over now!

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 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:

- groupware user
- sockets, HTTP and telephony functions
- creation of skin templates ...
- nested reports, queries with parameters ...

See the online help for more details.

We wish you great development experiences with **WinDev Mobile**!

APPENDICES

Glossary of the main words

A quick vocabulary reminder.

Window	A **window** is also called a **"dialog box"**. 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 **button** (text or graphic) is an area materialized by a "rectangle" that can be clicked in order to perform an action. In the icon bars, we talk of icons rather than buttons. A button is also called a "control".
Project	A **project** is a set of windows, reports, ... A project can use an **analysis** created by the data model editor.
Analysis	An "**analysis**" is a set of files (or tables) created to build a database.
Application	An **application** is a set of programs that perform specific actions.
Control / Item	We will use "**control"** for the screen areas and "**item"** 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 **table** 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 "**combo**" or "**combo box**" corresponds to a "**drop-down list**".
Scrollbar	We will use **scroll bar** or **scrollbar**.
File	A **file** is also sometimes called a "table". We will use "**table**" for an object used to view the content of a file or the content of a memory zone.
Record	A **record** is sometimes called a row. A record groups several items found in a file.
Item	An **item** is an area that belongs to a record.
SQL language	The **SQL language** 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.