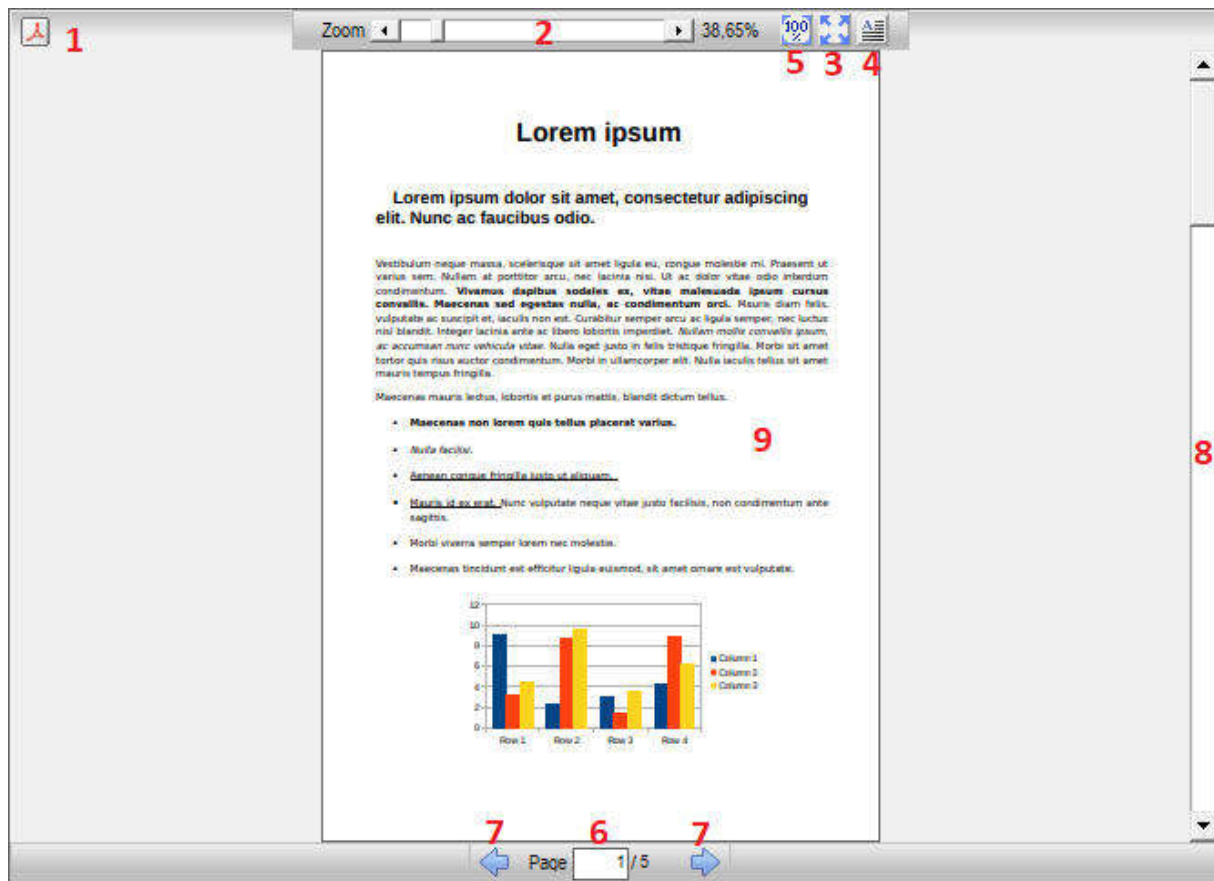
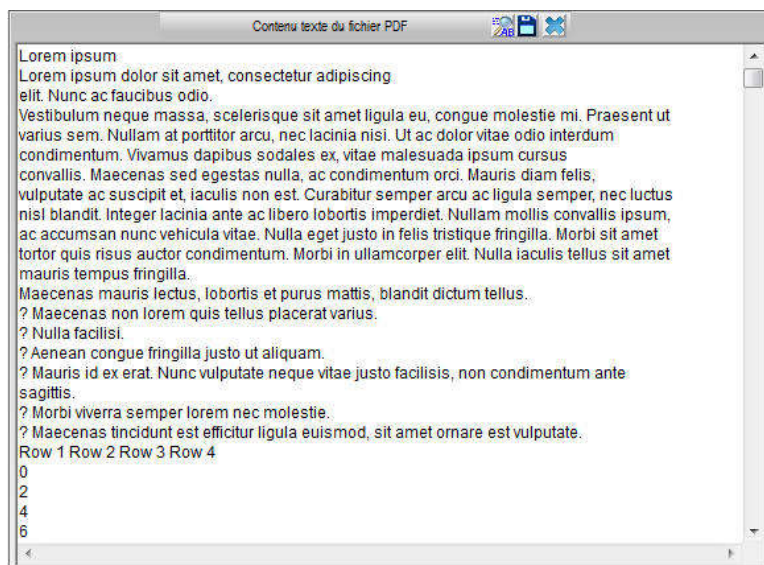


## The user interface




To interact with the control, the user has:

1. Button to open a PDF file: clicking on this button opens a file selector allowing you to load a PDF
2. Display zoom: allows you to enlarge or reduce the display area
3. Adapts the display to the size of the control: restores the full display of the page, after zooming
4. Displays the textual content of the PDF file:



a click on the icon  saves this text

If the file does not contain usable text and **Tesseract-OCR** is installed, the control proposes to attempt character recognition on the document (OCR). It is also possible to request an OCR by clicking on the button .

5. Displays the document at its actual size (Zoom 100%)
6. Displays or sets the number of the current page

7. Move arrows from page to page
  - Page UP and Page Down keys on keyboard double these arrows
8. Elevator also allowing to move within the document
9. Page display: it can be moved, especially when zooming
  - with the mouse
  - with the keyboard arrows
  - double click readjusts the display to the size of the control

The mouse wheel is managed according to the position of the cursor:

- on the display of the page (9):
  - scroll through pages of documents
  - If the Shift key ( ⇧ ) is pressed: scrolls the page up or down
  - If the right mouse button is pressed: scrolls the page to the right or left
  - If the Ctrl key is pressed: modifies the zoom of the page
- on the other parts of the control:
  - scroll through pages of documents

## Methods

Here is the list of methods specific to **PDFReader**

---

### Cls method

Clears the control display area and closes the current PDF document

#### Syntax

*object* .**Cls**

The syntax of the **Cls** method includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control

---

### CopyPageToClipboard Method

Copy the image of the page currently displayed to the Clipboard

#### Syntax

*Bool* = *object* .**CopyPageToClipboard** ()

The syntax for the **CopyPageToClipboard** method includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>bool</i>	Boolean, <i>True</i> if an image has been copied, otherwise <i>False</i>

---

### FitControl method

Adapts the displayed page to the size of the control

#### Syntax

*object* .**FitControl**

The **FitControl** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control

---

## GetPagesCount Method

Returns the page number of the document.

### Syntax

*number* = *object* .**GetPageCount**

The **FitControl** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>number</i>	Integer, contains the number of pages of the document

---

## GetPDFText Method

Returns a character string containing the text of the PDF file, if it exists.

### Syntax

*String* = *object* .**GetPDFText**

The **GetPDFText** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>String</i>	Character string, textual content of the document

### Remarks

- If the PDF file consists only of images (after using a scanner, for example), the function returns an empty string.
- 

## Load method

Load a PDF file into the control and optionally display it

### Syntax

*Boolean* = *object* .**Load** ( *FileName* , [ *ShowPDF* ] )

The **Load** method syntax includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>FileName</i>	Character string, Path and full name of the PDF document to load
<i>ShowPDF</i>	Boolean, optional. If True displays the first page of the document in the control
<i>Boolean</i>	True if the document has been loaded, otherwise False

## Remarks

- If the file does not exist, is not in PDF format, or if it is corrupted the function returns False, but does not display an error. On success the function returns True
  - The *ShowPDF* variable is optional, defaults to True
  - if *ShowPDF* = False, the document is loaded in the control, without being displayed. To display it use the **ShowPDF** method
  - To extract the text content from a PDF file, use the **Load** method with *ShowPDF* = False then the **GetPDFText** method , and finally the **Cls** method to close the document
- 

## MakeOCROnFile method

Loads a PDF document and performs character recognition on the document, if *Tesseract-OCR* is installed.

### Syntax

*String* = *object* .**MakeOCROnFile** ( [ *FileName* ] )

The **MakeOCROnFile** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>FileName</i>	Character string, Path and full name of the PDF document to load
<i>String</i>	Character string, OCR text made on the document

## Remarks

- If *FileName* does not exist, returns a live string
  - if Tesseract-OCR is not installed, displays an error message and returns an empty string
- 

## PrintPDF method

Print all or part of the open PDF document

### Syntax

*object* .**PrintPDF** ( [ *Copies* ], [ *Orientation* ], [ *FromPage* ], [ *ToPage* ], [ *PrinterName* ] )

The syntax for the **PrintPDF** method includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>copies</i>	Integer, optional. Number of copies of the document to print. If omitted takes the value 1
<i>Orientation</i>	oPrinterOrientation, optional. Printing orientation. If omitted takes the value oAuto
<i>FromPage</i>	Integer, optional. Number of the first page to be printed. If omitted takes the value 1
<i>ToPage</i>	Integer, optional. Number of the last page to print. If omitted takes as value the number of pages of the document
<i>String</i>	Character string, Name of the printer on which to print the document. If omitted prints on the default printer

## Remarks

- oPrinterOrientation can take the following values:
  - oAuto = 0: the orientation of the page to be printed is calculated automatically by the PDFReader component
  - oPortrait = 1: Portrait Orientation
  - oLandscape = 2: Landscape Orientation
- The default system printer is not changed during printing.

---

## SelectPDFFile Method

Displays a file selector, loads the PDF document chosen by the user and optionally displays it

### Syntax

*String* = *object* .**SelectPDFFile** ( [ *PDFPath* ], [ *ShowPDF* ])

The **SelectPDFFile** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>PDFPath</i>	Character string, Initial path of the file selector, optional
<i>ShowPDF</i>	Boolean, optional. If True displays the first page of the document in the control
<i>String</i>	Character string, Path and file name of the loaded document, otherwise empty string

## Remarks

- If *PDFPath* is an empty string or is omitted, the file selector opens in the user's "Documents" folder
- The *ShowPDF* variable is optional, defaults to True
- if *ShowPDF* = False, the document is loaded in the control, without being displayed. To display it use the **ShowPDF** method

---

## SelectPDFFileForOCR method

Displays a file selector, loads the PDF document chosen by the user and performs character recognition on the document, if **Tesserract-OCR** is installed.

### Syntax

*String* = *object* .**SelectPDFFileForOCR** ( [ *PDFPath* ])

The **SelectPDFFileForOCR** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>PDFPath</i>	Character string, Initial path of the file selector, optional
<i>String</i>	Character string, OCR text made on the document

## Remarks

- If *PDFPath* is an empty string or is omitted, the file selector opens in the user's "Documents" folder
- if Tesserract-OCR is not installed, displays an error message and returns an empty string

---

## ShowPDF method

Displays or redisplay the current page of the loaded document

### Syntax

*object* .**ShowPDF**

The **ShowPDF** method **syntax** includes the following:

Element	Description
<i>Object</i>	The <i>object</i> placeholder represents a PDFReader control

---

## Properties

In addition to the usual properties of almost any control (Align, BackColor, Enabled, Height, Width, Top, Left, Visible, ToolTipText etc ...) **PDFReader** has Properties specific to the handling of PDF files

---

### DisplayedPage property

Returns or sets the number of the page displayed within the loaded document.

#### Syntax

*object* .DisplayedPage [= *number* ]

The *object* placeholder represents a PDFReader control


#### Values

*number* is an Integer

#### Remarks

- This property cannot be modified if no PDF document is loaded and then takes the value 0
  - *number* can take as minimum value only 1 and maximum the page number of the document. If a number is assigned outside this range, the value of *number* is set to 1 if less, to the number of pages if greater.
- 

### IsPDFButtonVisible property

Returns or defines whether the button  allowing to invoke a file selector to change a PDF document is visible or not

#### Syntax

*object* .IsPDFButtonVisible [= *Value* ]

The *object* placeholder represents a PDFReader control

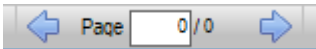
#### Values

*Value*: The property values are as follows:

Value	Description
true	The button is visible
false	The button is hidden

---

### IsStatusBarVisible property

Returns or sets whether the status bar  for navigating the document is visible or not

#### Syntax

*object* .IsStatusBarVisible [= *Value* ]



The *object* placeholder represents a PDFReader control

## Values

*Value:* The property values are as follows:

Value	Description
true	The status bar is visible
false	The status bar is hidden

---

## IsToolbarVisible property

Returns or sets whether or not the toolbar document text is visible



for zooming and extracting

## Syntax

*object* .**IsToolbarVisible** [= *Value* ]

The *object* placeholder represents a PDFReader control

## Values

*Value:* The property values are as follows:

Value	Description
true	The toolbar is visible
false	The toolbar is hidden

---

## Zoom property

Returns or sets the number of the page displayed within the loaded document.

## Syntax

*object* .**Zoom** [= *number* ]

The *object* placeholder represents a PDFReader control

## Values

*number* is a Double

## Remarks

- This property cannot be modified if no PDF document is loaded and then takes the value 100
  - The maximum zoom value is that where a screen pixel corresponds to a point in the document
-

## TesseractPath property

Returns or sets the number of the page displayed within the loaded document.

### Syntax

*object*.**TesseractPath** [= *string* ]

The *object* placeholder represents a PDFReader control

### Values

*string* is the installation path of Tesseract-OCR

---

## OCRLanguage property

Returns or sets the number of the page displayed within the loaded document.

### Syntax

*object*.**OCRLanguage** [= *string* ]

The *object* placeholder represents a PDFReader control

### Values

*string* is the default language for character recognition

### Remarks

- This property cannot be modified if no PDF document is loaded and then takes the value 100
- The maximum zoom value is that where a screen pixel corresponds to a point in the document

# Events

## PDFReader control specific events

---

### PageChanged event

Occurs when the **DisplayedPage** property of a PDFReader control is changed by the user or the application.

#### Syntax


**Private Sub** *object* **\_PageChanged (** *PageViewed As Integer***)**

The syntax for the **PageChanged** event includes the following:

Element	Description
<i>object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>PageViewed</i>	Integer. Number of the newly displayed page

---

### PDFLoaded event

Occurs when a file has just been successfully loaded into the control by the **Load** or **SelectPDFFile** method or by clicking on the button  on the user interface

#### Syntax

**Private Sub** *object* **\_PDFLoaded (** *FileName As String***,** *FilePath As String***)**

The **PDFLoaded** event **syntax** includes the following:

Element	Description
<i>object</i>	The <i>object</i> placeholder represents a PDFReader control
<i>FileName</i>	String. Name of the loaded PDF file
<i>File Path</i>	String. Path where the loaded PDF file is located

---