**Self-Learning Module 4 (Week 4)**

1. **Topic: Visual Basic 2010 – Using checked DateTimePicker Control**
2. **Competency**

Understand other different controls to be used in a program.

1. **Objectives**

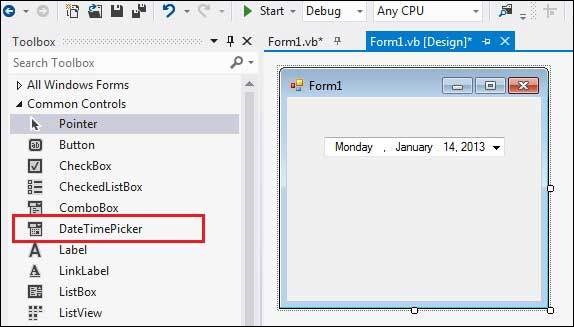
* Use of **DateTimePicker Control** in a program;
* Understand the event and properties of **DateTimePicker Control**;
* Incorporate **DateTimePicker Control** in basic program;

1. **Key Point**

The DateTimePicker control allows selecting a date and time by editing the displayed values in the control.

1. **Core Content**

The DateTimePicker control allows selecting a date and time by editing the displayed values in the control. If you click the arrow in the DateTimePicker control, it displays a month calendar, like a combo box control. The user can make selection by clicking the required date. The new selected value appears in the text box part of the control.



The **MinDate** and the **MaxDate** properties allow you to put limits on the date range.

Properties of the DateTimePicker Control

The following are some of the commonly used properties of the DateTimePicker control −

|  |  |
| --- | --- |
| **Sr.No.** | **Property & Description** |
| 1 | **BackColor**  Gets or sets a value indicating the background color of the DateTimePicker control. |
| 2 | **BackgroundImage**  Gets or sets the background image for the control. |
| 3 | **BackgroundImageLayout**  Gets or sets the layout of the background image of the DateTimePicker control. |
| 4 | **CalendarFont**  Gets or sets the font style applied to the calendar. |
| 5 | **CalendarForeColor**  Gets or sets the foreground color of the calendar. |
| 6 | **CalendarMonthBackground**  Gets or sets the background color of the calendar month. |
| 7 | **CalendarTitleBackColor**  Gets or sets the background color of the calendar title. |
| 8 | **CalendarTitleForeColor**  Gets or sets the foreground color of the calendar title. |
| 9 | **CalendarTrailingForeColor**  Gets or sets the foreground color of the calendar trailing dates. |
| 10 | **Checked**  Gets or sets a value indicating whether the Value property has been set with a valid date/time value and the displayed value is able to be updated. |
| 11 | **CustomFormat**  Gets or sets the custom date/time format string. |
| 12 | **DropDownAlign**  Gets or sets the alignment of the drop-down calendar on the DateTimePicker control. |
| 13 | **ForeColor**  Gets or sets the foreground color of the DateTimePicker control. |
| 14 | **Format**  Gets or sets the format of the date and time displayed in the control. |
| 15 | **MaxDate**  Gets or sets the maximum date and time that can be selected in the control. |
| 16 | **MaximumDateTime**  Gets the maximum date value allowed for the DateTimePicker control. |
| 17 | **MinDate**  Gets or sets the minimum date and time that can be selected in the control. |
| 18 | **MinimumDateTime**  Gets the minimum date value allowed for the DateTimePicker control. |
| 19 | **PreferredHeight**  Gets the preferred height of the DateTimePicker control. |
| 20 | **RightToLeftLayout**  Gets or sets whether the contents of the DateTimePicker are laid out from right to left. |
| 21 | **ShowCheckBox**  Gets or sets a value indicating whether a check box is displayed to the left of the selected date. |
| 22 | **ShowUpDown**  Gets or sets a value indicating whether a spin button control (also known as an up-down control) is used to adjust the date/time value. |
| 23 | **Text**  Gets or sets the text associated with this control. |
| 24 | **Value**  Gets or sets the date/time value assigned to the control. |

Methods of the DateTimePicker Control

The following are some of the commonly used methods of the DateTimePicker control −

|  |  |
| --- | --- |
| **Sr.No.** | **Method Name & Description** |
| 1 | **ToString**  Returns the string representing the control. |

Events of the DateTimePicker Control

The following are some of the commonly used events of the DateTimePicker control −

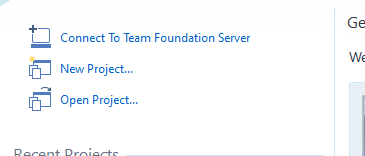
|  |  |
| --- | --- |
| **Sr.No.** | **Event & Description** |
| 1 | **BackColorChanged**  Occurs when the value of the BackColor property changes. |
| 2 | **BackgroundImageChanged**  Occurs when the value of the BackgroundImage property changes. |
| 3 | **BackgroundImageLayoutChanged**  Occurs when the value of the BackgroundImageLayout property changes. |
| 4 | **Click**  Occurs when the control is clicked. |
| 5 | **CloseUp**  Occurs when the drop-down calendar is dismissed and disappears. |
| 6 | **DoubleClick**  Occurs when the control is double-clicked. |
| 7 | **DragDrop**  Occurs when a drag-and-drop operation is completed. |
| 8 | **ForeColorChanged**  Occurs when the value of the ForeColor property changes. |
| 9 | **FormatChanged**  Occurs when the Format property value has changed. |
| 10 | **MouseClick**  Occurs when the control is clicked with the mouse. |
| 11 | **MouseDoubleClick**  Occurs when the control is double-clicked with the mouse. |
| 12 | **PaddingChanged**  Occurs when the value of the Padding property changes. |
| 13 | **Paint**  Occurs when the control is redrawn. |
| 14 | **RightToLeftLayoutChanged**  Occurs when the RightToLeftLayout property changes. |
| 15 | **TextChanged**  Occurs when the value of the Text property changes. |
| 16 | **ValueChanged**  Occurs when the Value property changes. |

**Example**

**Step 1:** Click Microsoft Visual Studio 2010 icon on your desktop.



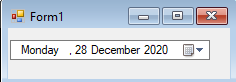
**Step 2:** Click New Project



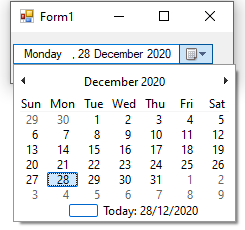
**Step 3:** Type “datetime” in the name as your filename.



**Step 4:** On your toolbox look for DateTimePicker and insert it to your form. And it should like this



Try to run the program and you should be able to use the DateTimePicker.



**Step 5:** Go back to the form and then add another DateTimePicker, a label and a button.

**Step6:** Double click the button then type the codes below;

*Dim date1 As DateTime = DateTimePicker1.Value*

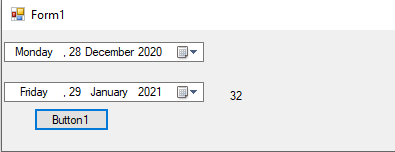
*Dim date2 As DateTime = DateTimePicker2.Value*

*Dim result As TimeSpan = date2.Subtract(date1)*

*Dim days As Integer = result.TotalDays*

*Label1.Text = days*

Run the program then select a different date on the second DateTimePicker, like the image below



The program will display the number of days from date 1 to date 2.

**Explanation:**

We declare 2 variables to hold the value of the DateTimePickers just to make it more shorter when it comes to our codes.

*Dim date1 As DateTime = DateTimePicker1.Value*

*Dim date2 As DateTime = DateTimePicker2.Value*

*Dim result As TimeSpan = date2.Subtract(date1)*

This line has the variable that will hold the difference between 2 dates.

**Worksheet 1**

**I. Topic:** Use *DateTimePicker* in a program

**II. Objectives**

* Incorporate *DateTimePicker* to perform math problems.
* Use *DateTimePicker* to initiate an event.

**III. Activity 1**

Create a program that will display your age on a label using only with one *DateTimePicker.*