Application Development Orienting

Assignments week 1

Quiz questions, practical assignments and

answers to quiz questions

**Version:** 0.1 **Last updated:** 06 september 2019

Contents

[1 Quiz 3](#_Toc536006402)

[1.1 Question 1 3](#_Toc536006403)

[1.2 Question 2 3](#_Toc536006404)

[1.3 Question 3 3](#_Toc536006405)

[1.4 Question 4 3](#_Toc536006406)

[1.5 Question 5 4](#_Toc536006407)

[1.6 Question 6 4](#_Toc536006408)

[2 Practical assignments 5](#_Toc536006409)

[2.1 Programming Assignment 1: Exploring visual controls 5](#_Toc536006410)

[2.1.1 Screenshots 5](#_Toc536006411)

[2.2 Programming Assignment 2: Simple calculator 7](#_Toc536006412)

[2.2.1 Screenshots 7](#_Toc536006413)

[2.2.2 Case description 7](#_Toc536006414)

[2.3 Programming Assignment 3: Truck management 8](#_Toc536006415)

[2.3.1 Case description 8](#_Toc536006416)

[2.3.2 User interaction 8](#_Toc536006417)

[2.3.3 Screenshots 8](#_Toc536006418)

[3 Quiz answers 9](#_Toc536006419)

# Quiz

Difficulty: C:\Users\874156\Desktop\flatastic-icons-part-1-by-custom-icon-design\png\16x16\star-2_5.png. Estimated time: 30 minutes.

Answers to the quiz-questions can be found in the last section of this chapter.

## Question 1

What is the value of the variable **c** after running this piece of code: 20

int a;

a = 10;

int b;

b = 20;

int c;

c = a + b;

## Question 2

What is the value of the variable **sum** after running this piece of code: 45

int a;

a = 10;

int b;

b = a + 5;

int c;

c = b + 5;

int sum ;

sum = a + b + c;

## Question 3

Can you find 3 errors in the next piece of code (line numbers do not belong to the code, but are added to easily identify the lines with errors):

1 int myShoesize;

2 myShoesize = 44.5; - mistake

3 float myShoesize;

4 myShoesize = 39.5;

5 int double;

6 double = 44;

## Question 4

What is the value of the variable **totalPrice** after running this piece of code: 119.5

float price;

price = 12.95;

int amount = 3;

amount = amount + 4;

amount = amount + 3;

float totalPrice;

float discount;

discount = 10.0;

totalPrice = amount \* price - discount;

## Question 5

What are the values of the variables div, remainder, x and y after running this piece of code:

int a = 97;

int b = 20;

int div = a / b;

int remainder = a % b;

int x = ((a / b) / b) / b;

int y = ((a % b) % b) % b;

Div = 12

Rem = 17

X = 0

Y = 17

## Question 6

What are the values of the variables **a** and **b** after running this piece of code: a = -10 b = -50

int a = 20;

int b = 10;

a = a + b;

b = a + b;

a = a - b;

b = a - b;

# Practical assignments

## Programming Assignment 1: Exploring visual controls

Difficulty: C:\Users\874156\Desktop\flatastic-icons-part-1-by-custom-icon-design\png\16x16\star-2_5.png. Estimated time: 60 minutes.

The assignment covers the following learning goals:

* Understand how to use Windows Forms Application.
* Understand how properties and events function for controls.

### Screenshots

#### Case description

In this assignment, you learn how to build a Windows Forms Application in Visual Studio. You will learn how to edit properties of the controls and use their events.

1. First create a new Windows Forms Application project by choosing “File -> New -> Project”. In case you did not chose C# when you installed Visual Studio, it might be that Visual Studio starts with another programming language. In that case, chose in “Template” for “Visual C#”. At this point, there are many options for the kind of applications (or apps) that you might want to implement in C#. We select the Windows Forms Application.  
   Visual Studio suggest the default name “WindowsFormsApp1” for your app. Change this to a better name, something like “MyFirstWindowsApp”. Be aware that the name should be an identifier.  
   Chose a location (a folder on your hard drive) in which your first app should be stored. For this purpose there is a button with text “Browse” to indicate the location to store this project.  
   Remember where you store it: you might need it for later (step 9).  
   The Solution name (at the bottom of the window) is the same as Name. We keep it as it is.
2. The text on the title-bar of the window is "Form1". Select the window (Form1) by clicking once on it. Now the property-window displays the properties of the window. Change the property Text in "My first program".
3. Add a button on your window by selecting a button in the Toolbox and add it to the window. In the screenshot above, its BackColor is yellow. Give this button a nice color (be creative; what is your favorite color?). Change the text of the button in something like “Who am I?” or “Information about me”.
4. Clicking this button should results in displaying information about you on the screen. You will have to program it. There are 2 ways of choosing an event: one way is illustrated in the lesson (in the Property-window choose for events and then the click-event). Now use the other way. Microsoft has chosen a “most-likely-meant-event” for every control. For a button, it is the click-event.  
   Double-click the button. Visual Studio will direct you to the event-handler for button1\_click. Place the following text in body of this event handler:  
   MessageBox.Show("My name is <your\_name>");   
   Run your app and click the button.
5. The next control (the red one in the above figure) is a trackbar. Place a trackbar on your window and change its Minimum and Maximum in, for instance, 0 and 100. Change its property Value in 20. Run your program to see what happens. Now you can scroll with this trackbar, but nothing happens. Also, you can give the trackbar a nice BackColor.
6. Place a progress bar (in the above picture the lower green/gray rectangle). Also place a label on your window (in the picture it has the text "50"). Normally the font is very small. Can you give this label a bigger font?
7. We would like the progress bar and the text in the label to “follow the trackbar” (if the user scrolls with the trackbar, its value should be displayed in the label and be visual in the progress bar).  
   Make the progress bar’s property Minimum equal to 0 and its Maximum equal to 100 (the same Minimum and Maximum as your trackbar)  
   Doubleclick on the trackbar to let Visual Studio generate the “most-likely-meant-event”-handler:  
   private void trackBar1\_Scroll(object sender, EventArgs e)  
   {  
   }  
     
   Type the code  
    this.progressBar1.Value = this.trackBar1.Value;  
    this.label1.Text = this.trackBar1.Value.ToString();  
   in this trackBar1\_Scroll-method, run your program and see what happens.
8. There are several kind of buttons. You already practiced with a normal button. Next, we will explore radiobuttons. You see three of them on the right side of the window.  
   The most important property of a radiobutton is the property “Checked”. It’s value is a boolean (true or false). When the user selects a radionbutton, its property “Checked” becomes true and you see it visually on the screen (a dot in the circle before the text). But there is more: if you place some radiobuttons on the form, at every moment there cannot be 2 or more selected. If you click on a radiobutton, the clicked radiobutton will become selected and all other radiobuttons will automatically be deselected. Run your app to check it.  
   If you push the first radiobutton, the window's backcolor should become red. This radiobutton has already a nice text. Choose some nice texts for the two other radiobuttons.  
   By the way, so far we did not change the name of any visual control. Since we have only 1 button, there will be hardly any confusion what is meant by “button1”. Part of pragmatics is to choose an appropriate name. Now we have 3 radiobuttons, so please, don’t name them radiobutton1, radiobutton2, radiobutton3, as proposed by Visual Studio. Rename them to something like rbMakeMeRed, rbMakeMe??? and rbMakeMe???.  
   When doubleclicking on a radiobutton, Visual Studio generates the “most-likely-meant-event”-handler CheckedChanged. For the radiobutton rbMakeMeRed, add the code  
    this.BackColor = Color.Red;  
    to it. There are some pre-defined colors. If you type Color, followed by a dot, they will pop up.  
   Do the same for the other radiobuttons.
9. Now you are so proud of your first program that you cannot wait to send it to your family. Maybe you want your picture on the window: just do it (use a picturebox to hold your image. Most probably your image does not have the same size as the picturebox. Ask a fellow student or the teacher how to handle it).  
   After running your app, Visual studio creates a subfolder with the same name as your project. In it you find the subfolder “bin” and in that folder “debug”. There are some files. One of them is the executable (.exe). This executable runs on Windows, even if you do not have Visual Studio. Thus, now you can send this app to your family.

## Programming Assignment 2: Simple calculator

Difficulty: C:\Users\874156\Desktop\flatastic-icons-part-1-by-custom-icon-design\png\16x16\star-4_5.png. Estimated time: 60 minutes.

The assignment covers the following learning goals:

* Use variables in the applications.
* Use and interact with different kinds of controls.

### Screenshots

### Case description

This assignment is about making a calculator app. It involves using the controls textbox, button and label. Additionally, we will look at panels as well.

The functionality of this app is quite straightforward: the user inputs a first number and a second number. Clicking one of the buttons (e.g. the add-button) should result in the corresponding operation applied on the two numbers (e.g. calculating the sum). Finally, the result should be shown on the screen.

Create a new Windows app with a nice name. The first step is adding the two panels to the form (the yellow and green rectangles in the screenshot above). Panels are there to decorate your window and to group things together. You might not be able to see the two panels, since they have the same BackColor as the window. Change their background colours into something nice. The yellow panel is there to group the calculating functionality; the green panel is there to show or hide information about this app.

Start with the yellow panel. Add the necessary buttons, labels and textboxes to make it look like in the screenshot above. Ensure that the buttons, textboxes and label at the bottom have nice names (for instance tbFirstNumber and tbSecondNumber for the textboxes; btnAdd, btnSubtract, btnMultiply and btnDivide for the buttons; lblAnswer for the label at the bottom).

Oops, that’s not good-looking: the buttons do not have the same size and their location is not on the same height. To give them the same size, you can select all of them simultaneously. Change the size in the Property-window. Now all selected controls should have the same size. Also, find a way to align all selected controls in such a way that they all are aligned horizontally (or is it vertically?).

Now program some code in the btnAdd-eventhandler to calculate the sum of the two numbers, as entered in the textboxes. Display the answer in the label at the bottom. You may assume that the user inputted texts in the textboxes, that can be converted to numbers. Do the same for the other buttons. Run your app and test if it works.

We continue with the second part of the app: the green pane. We would like to show/hide some information about this app. Add two buttons to this panel and a textbox.

Normally the text of a textbox is meant to hold one string, but we want to have many strings in it. Select the textbox and check the property Multiline. It’s default value is false, now change it to true. Now you can change the height of the textbox. When you select the property Lines, a button with 3 dots appear. Clicking this 3-dotted-button gives you the possibility to type some lines of text in the textbox.

If you run your app, both buttons and the textbox are visible, because the default value for their visibility is true. Change the values for the property Visible into false for the button with text “hide info” and for the textbox.

Clicking the button with text “show info” should make the textbox and the other button visible, and this button invisible. Clicking the button with text “hide info”: well, you can imagine what should happen. Make it work.

## Programming Assignment 3: Truck management

Difficulty: C:\Users\874156\Desktop\flatastic-icons-part-1-by-custom-icon-design\png\16x16\star-4_5.png. Estimated time: 60 minutes.

The assignment covers the following learning goals:

* Use operators in your app.

### Case description

The owner of a truck company needs help in managing his business. He needs an application to calculate how many trucks are required to transport a certain order. An order consists of a number of boxes that a supermarket orders. The next day, a transport company brings the ordered boxes, but they need to be able to calculate how many trucks are needed. Not all trucks have the same capacity, so the user should be able to specify how many pallets fit in a truck. The user should also be able to specify how many boxes fit on a pallet.

### User interaction

Start a new project and add some controls to it. We start with the rectangles that look like panels, but they are not. A panel decorates the GUI. If you want to do so, but you also want to have a text at the top of it, use a groupbox. A groupbox has the same functionality as a panel, but it has some text on top of it. So first put two groupboxes on the window and then some other controls.

After inputting some data in the textboxes, the user wants to know how many full trucks are needed to deliver the order, how many remaining pallets there are and how many remaining boxes there are. Oops, forgot to add the necessary controls to the window as shown in the above screenshot. Add the necessary controls yourself and make it work.

### Screenshots

# Quiz answers

|  |  |
| --- | --- |
| Question | Answer |
| 1 | 30 |
| 2 | 45 |
| 3 | Line 2 (not possible to assign that value to an int-variable)  Line 3 (there is already a variable with that name)  Line 5 (double is a reserved word, cannot be used as identifier for a variable) |
| 4 | 119.5 |
| 5 | 4, 17, 0, 17 |
| 6 | -10, -50 |