# **Downloading Visual Studio**

Created by Haleh Amanieh



### 1. Go to visualstudio.Microsoft.com



 On the dropdown menu under Visual Studio, choose Community
 2019

Note: Visual Studio for Mac does not support creating Windows Forms nor coding in Visual Basic





4. Follow the steps to install the program

#### Visual Studio Installer

Installed Available

## 5. ... (Wait)

Almost there...

. . . . .

#### Developer News

Announcing .NET 5.0 Preview 7 The post Announcing .NET 5.0 Preview 7 appear... Thursday, July 23, 2020

ASP.NET Core Updates in .NET 5 Preview 7

The post ASP.NET Core Updates in .NET 5 Previe... Thursday, July 23, 2020

Exciting new updates to the Git experience in Visual Studio

The post Exciting new updates to the Git...

Thursday, July 23, 2020

View more online...

Need help? Check out the <u>Microsoft Developer</u> <u>Community</u> or reach us via <u>Visual Studio Support</u>.

Installer Version 2.6.2037.624



## 6. Click Launch

## Visual Studio 2019

Open rec

Yesterday
 Older

ent		Get sta	rted
: (Alt+S)	<b>-</b> م	⇒	Clone a repository Get code from an online repository like GitHub or Azure DevOps
		ስ	Open a project or solution Open a local Visual Studio project or .sln file
		2	Open a local folder Navigate and edit code within any folder
		•	Create a new project
			Continue without code $ ightarrow$

7. You can choose to create a new project or open a project.

## Sign in to Visual Studio

Visual Studio will help you plan projects, collaborate with your team, and manage your code online from anywhere.

#### Learn more

Sign <u>i</u>n

All Acc

16.0.30320.2

Sign in to start using your Azure credits, publish code to a private Git repository, sync your settings, and unlock the IDE.

ounts Sign in to yo	ur account
Sign in to yo	ur account
	Visual Studio
D16.6	Microsoft Sign in
	Email, phone, or Skype
	No account? Create one!
	Can't access your account?
	Sign-in options
	Back Next

## **Visual Studio**

#### Community 2019

License: 30 day trial (for evaluation purposes only) Your evaluation period has ended.

Your evaluation period has ended. Please sign in to unlock the product.



## 8. You will need to sign in with your Microsoft account

#### \_ Create a new - م Search for templates (Alt+S) project All <u>p</u>latforms Visual Basic All project types Recent project templates Console Visual Basic Windows Windows Forms App (.NET Windows Forms App (.NET Framework) VB Visual Basic A project for creating an application with a Windows Forms (WinForms) user Framework) interface Visual Basic Windows Desktop Class Library (.NET Framework) Library Visual Basic Windows Class Library (.NET Core) A project for creating a class library that targets .NET Core. Library Linux macOS Visual Basic Windows **JYB** Unit Test Project (.NET Framework) A project that contains MSTest unit tests. Test Visual Basic Windows **T**Y<sup>B</sup> xUnit Test Project (.NET Core) Back <u>N</u>ext

9. After signing in, scroll until you find "Windows Forms App (.NET Framekwork)." Select it and click "Next"

		-		>
Configure your new project				
Windows Forms App (.NET Framework) Desktop Visual Basic Windows				
Project <u>n</u> ame				
WindowsApp1				
Location				
C:\Users\haman\source\repos *				
Solution name 🕕				
WindowsApp1				
Place solution and project in the same <u>directory</u>				
Eramework				
.NET Framework 4.7.2 🔹				
	<u>B</u> ack	<u>C</u> reat	e	

10. Here you can choose the Project name, solution name, and the location in which the files for that project will be stored.



## 11. You can now begin creating your project!

# Creating a project

This document walks the reader through the creation of a program in which clicks on a button is counted and displayed on the form.



## 1.Start with your Windows Form.



2. In the "Properties" section, you can change features of the form such as the color, size, and text.

🗄 🛃 🖗 🏓	
RightToLeft	No
RightToLeftLayout	False
Showlcon	True
ShowInTaskbar	True
Size	816, 489
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	Hello World!
TopMost	False

e text associated with the control.

Solution Explorer Team Explorer	
Properties	- ‡ x
Form1 System.Windows.Forms.Form	-
🗄 👥 🖓 🕖 🥬	
(ApplicationSettings)	▲
(Name)	frmHelloWorld
AcceptButton	(none)
AccessibleDescription	
AccessibleName	
AccessibleRole	Default
AllowDrop	False
AutoScaleMode	Font
AutoScroll	False
	0, 0
	0, 0
AutoSize	False 🗸
(Name) Indicates the name used in code to identify the	object.
	🛧 Add to Source Control 🔺 🖣
	Properties Form1 System.W
Categories	Alphabetize

2a. The properties can be organized according to the category the property falls into or they can be alphabetized.

So	lution Explorer Team Explorer			
Pr	operties		×	
Fe	orm1 System.Windows.Forms.Form		- ·	
0	1 💤 🐔 🌽 🥒			
Ŧ	(ApplicationSettings)			
Ŧ	(DataBindings)			
	(Name)	frmHelloWorld	_	
	AcceptButton	(none)		
	AccessibleDescription			
	AccessibleName			
	AccessibleRole	Default		
	AllowDrop	False		
	AutoScaleMode	Font		
	AutoScroll	False		
Ŧ	AutoScrollMargin	0, 0		
Ŧ	AutoScrollMinSize	0, 0		
	AutoSize	False		
()	lame)			
In	dicates the name used in code to identify the	object.		
		🛧 Add to Source Control 🔺 🏼 🧧	0	

3. Change the Name of the form. The name is what you will use to refer to the form in your code.

Names of objects follow a convention. Notice the name starts with "frm" which is used only when naming a form.

You can find a list of naming conventions at codingbehavioranalysts.org/visualb asic



4. Change the text of the form. The text is what will be displayed on the form when the program is running.



5. Now we can add objects to our form. Click "Toolbox."



6. Displayed in the toolbox are "Common Controls." These are objects you can add to your form that users can interact with or that can interact with your code. Let's start by adding a button.



Data Sources

7. You can double click "button" and a button will appear on your form. Or you can click "button" one time and click and drag your button onto your form. You can make the button any size you want and place it anywhere on your form.

Properties		<b>- - -</b>
btnHelloWorld System.Windows.Forms	Button	- 2
11 🐏 🐑 🖑 🎾		
(ApplicationSettings)		
🕀 (DataBindings)		
(Name)	btnHelloWorld	
AccessibleDescription		
AccessibleName		
AccessibleRole	Default	
AllowDrop	False	2
Anchor	Top, Left	
AutoEllipsis	False	
AutoSize	False	
AutoSizeMode	GrowOnly	
BackColor	Control	· · · · ·
BackgroundImage	(none)	<b>_</b>
(Name) Indicates the name used in code to ident	ب. ify the object.	2
	↑ Add to Source Co	ontrol 🔺 🖣

8. Click on the button that is now on your form to view its properties. Just like with the form's properties, you can change the button's properties. Change the name to include the naming convention "btn."

Properties	<b>→ ₽ ×</b>
btnHelloWorld System.Windows.Forr	ms.Button <del>-</del>
H 💤 🖗 🥖	
RightToLeft	No
∃ Size	229, 134
TabIndex	0
TabStop	True
Tag	
Text	Hello World!
TextAlign	MiddleCenter
TextImageRelation	Overlay
UseCompatibleTextRendering	False
UseMnemonic	True
UseVisualStyleBackColor	Тгие
UseWaitCursor	False
Visible	True
Taut	
The text associated with the control.	
	🛧 Add to Source Control 🔺 🖣

9. Change the text that appears on the button. You can change it to be whatever you want.



ropenies			
btnHelloWorld System.Windows.Forms.Butto	n	<b>-</b> 2	11. I've also changed the font type
H 💤 🖗 🌽			and size, just for fun.
Enabled	True	A	
E FlatAppearance			
FlatStyle	Standard	2	
B Font	Yu Mincho, 14.25pt		1
ForeColor	ControlText	Ν.	
GenerateMember	True		
Image	(none)	2	
ImageAlign	MiddleCenter		
ImageIndex	(none)		
ImageKey	(none)		
ImageList	(none)		
E Location	245, 112		
Locked	False	Hello World!	
- · · ·			
Font			
The font used to display text in the control.			ае
	↑ Add to Source Co		Hello World!





12. Add a label on to your form.

Properties			×	
IblCounter System.Windows.Forms.Label		-	2	
11 🐏 🖗 🥖				
(Name)	IblCounter			
AccessibleDescription				
AccessibleName			ы.	
AccessibleRole	Default			
AllowDrop	False			2
Anchor	Top, Left			
AutoEllipsis	False			
AutoSize	True			
BackColor	Control			:
BorderStyle	None			
CausesValidation	True		<b>–</b>	
(Name) Indicates the name used in code to identify the	object.			
	↑ Add to Source Co	ntrol 🔺 🛛 🐴		

13. Re-name the label. I named it IblCounter because the naming convention for labels is "Ibl" and because it will display the number of times the button is clicked.

We won't change the text that the label displays in the properties section. Instead, we will change the text using code.



14. Double click the form to open the tab in which you can code.



15. The application pre-loads some code for you. Because you double clicked the Form, it coded the sub in which you can write code that is executed when the Form loads.



16. We want the label to not display any text when the Form loads. First, type the label's name "IblCounter." Visual basic is very user friendly and gives you suggestions, shown here.

	Γ	1 reference
1	Ę	Public Class frmHelloWorld
		0 references
2	Ę	Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
3 🖌		lblCounter.Text = ""
4		End Sub
5		
6		End Class
7	Ľ	

17. To change the Text property of the label, add ".Text" at the end of the label's name. We want the text to be blank, so we code "=" and "".
You can change the text of the label to be whatever you want by adding "=" and the text in quotes.



18. Go back to the "design" tab and double click the button that we created earlier. Now there is a Sub in which we can create code for when the button is clicked.

orm1.v	b* +¤	х	Form1.vb [Design]*		
∞ My fi	rst pro	ject	- 🔩 frmHelloWorld	-	କ
		1 re	eference		
	1 6	⊒Pu	blic Class frmHelloWorld		
	2				
			0 references		
	3 [	<u> </u>	Private Sub Form1 Load(sender As Object, e As EventArgs) Handles MyBase.Load		
			lblCounter.Text = ""		
1	5		End Sub		
	5	H			
	1		0 references		
	7 8	5:	Private Sub btnHelloWorld Click(sender As Object, e As EventArgs) Handles btnHelloWorld.Clic	k	
	3		Static Dim intX As Integer		
	9				
10	a 📕		End Sub		
		En			
			a £1033		
1.	2				

19. Create a variable that will count the number of times the button is clicked. "Static" will preserve the value assigned to the variable "intX." "Dim" is short for dimension and is used to declare a variable. "intX" is the name of our variable, we can name it whatever we want, but notice the use of "int" from naming conventions. Finally, "As Integer" is included to tell the computer that this variable is used to store numbers.

or	m1.vb*	÷	×F	Form1.vb [Design]*	
/B	My first	proj	ect	🗸 🔩 frmHelloWorld 🗸 🗸	ଦ୍ଧ
			1 ref	ference	
		Ę	Pub	blic Class frmHelloWorld	
				0 references	
		Ē		Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load	
				lblCounter.Text = ""	
			-	End Sub	
				0 references	
		Ē	] [	Private Sub htnHelloWorld_Click(sender As Object, e As EventArgs) Handles btnHelloWorld.Click	
				Static Dim intX As Integer	
				intX += 1	
	10			End Sub	
	11		End	d Class	
	12				

20. We want to count the number of times the button is pressed. To do that, we will write code that adds 1 to the variable intX. The code "intX += 1" is short for intX = intX + 1. Either one will do.

.vb -	⊨ × Fo	orm1.vb [Design]	
first p	project	- 🔩 frmHelloWorld -	, Q
	ference		
	📮 Pub	olic Class frmHelloWorld	
		0 references	
	ė.	Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load	
		lblCounter.Text = ""	
5		End Sub	
		0 references	
		Private Sub btnHelloWorld Click(sender As Object, e As EventArgs) Handles btnHelloWorld.Click	k
		Static Dim intX As Integer	
		intX += 1	
10		lblCounter.Text = intX_ToString	
11	• • • •	End Sub	
10 -	End		
12	End		
	vb - first   1 2 3 4 5 6 7 8 9 10 11 11 12	wb         ⇒         ×         Fi           first project         1 re           1         □Pub           2         1           3         □           4         5           5         1           7         □           8         10           11         □           12         End	vb *   form1.vb [Design] first project I reference 1 Provide Sub form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load 4 IblCounter.Text = "" End Sub 6 Oreferences 7 Private Sub btnHelloWorld_Click(sender As Object, e As EventArgs) Handles btnHelloWorld.Click 8 Static Dim intX As Integer 9 intX += 1 10 IblCounter.Text = intX.ToString End Sub 12 End Sub

21. Now we want to display the value of intX on the form with the label. Labels display text, called Strings. Because intX is an integer, and not a string, we have to convert it to a string using ".ToString"

C <sup>il</sup> → De	bug -	Any CPU - Start - Star	
Form1.vb	⊸ × Fo	orm1.vb [Design]	
🕫 My firs	t project	🚽 🔩 frmHelloWorld	- Ø <sub>6</sub>
	1 refe	erence lic Class frmHelloWorld Oreferences Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load   lblCounter.Text = "" End Sub	
6 7 8 9 10 11 12	End	Oreferences Private Sub htnHelloWorld_Click(sender As Object, e As EventArgs) Handles btnHelloWorld.Clic Static Dim intX As Integer intX += 1 lblCounter.Text = intX.ToString End Sub Class	ck

22. Finally, we can execute the code to see if it works. Press the Start button at the top.

– 🗆 🗙

23. You will see the programexecuted. Notice that the label isblank when the form loads.

Hello World!



23. Click the button! The number of times it is clicked with be displayed in the label.



