



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LECTURE: INTRODUCTION VISUAL BASIC IDE

- THE INTEGRATED DEVELOPMENT ENVIRONMENT
- DIFFERENT PROJECT TYPES
- REVIEW OF ALGORITHMS
- FORMS APPLICATION
- TOOLBOX AND OBJECTS IN IT
- INTRODUCING PROPERTIES AND METHODS
- ONE SIMPLE EXAMPLE, HELLO VB!



WHAT IS AN IDE

- IDE stands for “Integrated Development Environment”
- Integrated because the IDE’s generally have everything we need to do our work. This includes:
 - Source Code Editors: This is the text editor where we can write our code
 - Local build Automation: these are tools that automate repetitive tasks. This can include compiling our code with the press of a button, rather than going into PowerShell to compile the code.
 - Debugger: Having access to a built-in debugger has great advantages. You can go through your code step by step and when you find the bug, you can edit the source code directly rather than going from a separate system.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

- Visual Studio is a Microsoft IDE, that is different from Visual Studio Code. VS Code is a text editor, While Visual Studio is a IDE.
- <https://visualstudio.microsoft.com/>



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

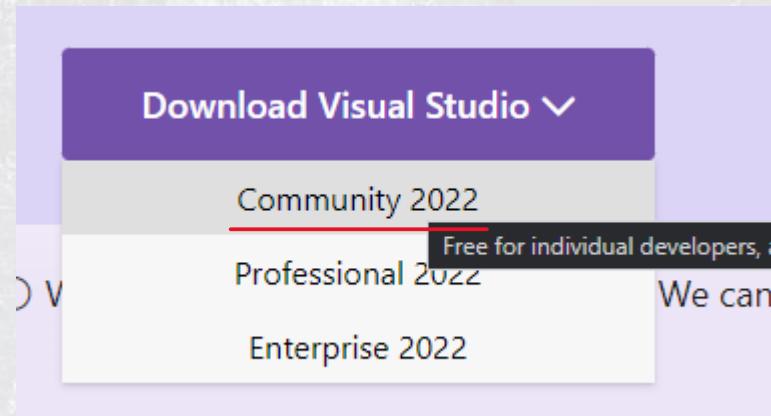
The screenshot shows the Microsoft Visual Studio download page. At the top, there's a navigation bar with links for Microsoft, Visual Studio, Developer Tools, Downloads, Buy, Subscriptions, and Free Visual Studio. On the right side of the nav bar are links for All Microsoft, Search, and Sign in. Below the nav bar, a large purple button says "Download Visual Studio". To its right, three options are listed: "Community 2022" (free), "Professional 2022" (for individual developers, academic institutions, and government organizations), and "Enterprise 2022". Below these options, there are three cards: "Visual Studio" (Windows), "Visual Studio for Mac" (Mac), and "Visual Studio Code" (Windows, macOS, Linux). Each card has a "Learn more" link and a "Download Visual Studio" or "Download Visual Studio for Mac" button. A footer at the bottom of the page says "Wondering which tool is best for you? We can help." and includes a "Feedback" link.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO





INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

The screenshot shows the Microsoft Visual Studio download page. At the top, there's a banner with a man and a woman looking at a laptop screen. Below the banner, the text "Thank you for downloading Visual Studio" is displayed, along with a link to "click here to retry" if the download doesn't start. On the left, a sidebar lists options like "Install Visual Studio", "Getting started", "Personalize", "Start a tutorial", ".NET", "C++", and "Python". A red diagonal line points from this sidebar towards a warning message at the bottom. The main content area includes sections for "New to Visual Studio?", "Install Visual Studio", and "Getting started with Visual Studio IDE". A footer bar at the bottom contains a warning about the executable file being potentially harmful, with "Keep" and "Discard" buttons, and a "Feedback" button.

Microsoft | Visual Studio | Developer Tools | Downloads | Buy | Subscriptions | Free Visual Studio

All Microsoft | Search | Sign in

Thank you for
downloading
Visual Studio

Your download will start shortly. If your download does not begin, click here to retry

New to Visual Studio?

Install Visual Studio

Getting started with Visual Studio IDE

This type of file can harm your computer. Do you want to keep VisualStudioSetup.exe anyway? **Keep** **Discard**

To develop any type of app or learn a language, you'll be working in the Visual Studio Integrated Development Environment (IDE). Beyond code editing, Visual Studio IDE brings together graphical designers, complex code

Show all X Feedback



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

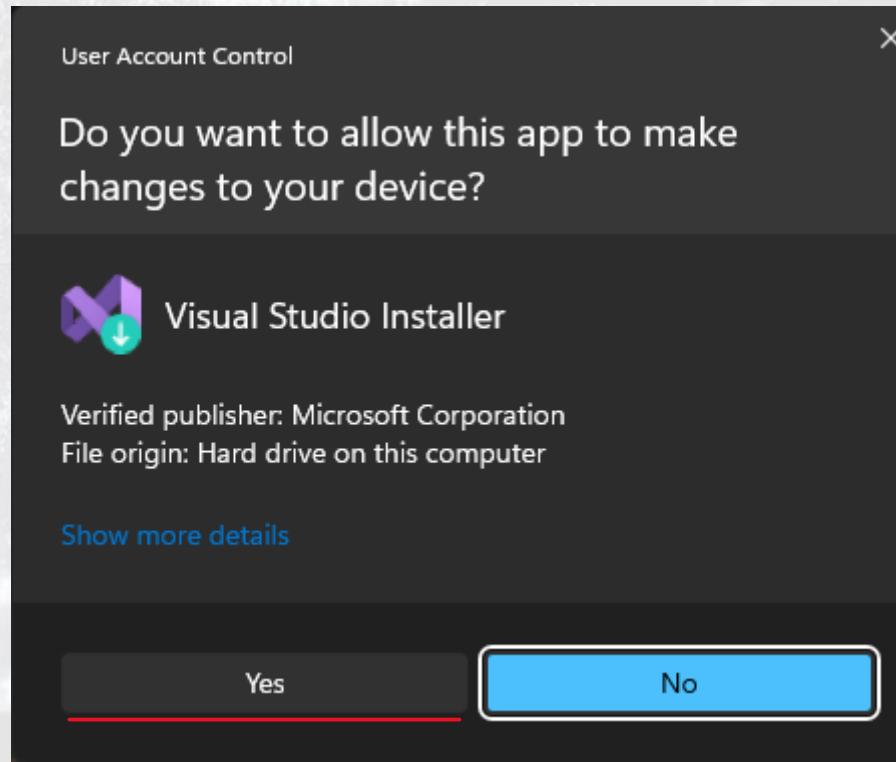




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

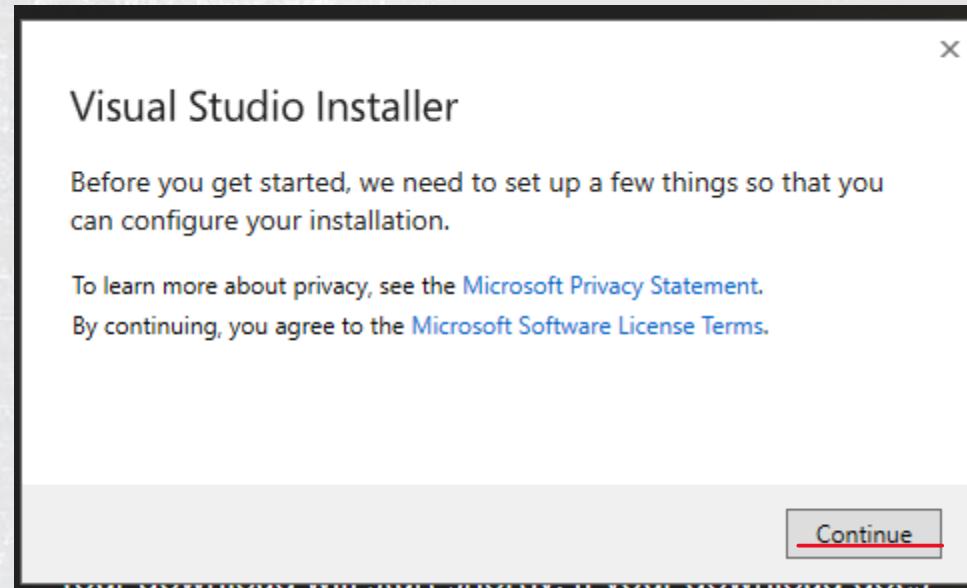




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO





INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

ASP.NET and web development
Build web applications using ASP.NET Core, ASP.NET, HTML/JavaScript, and Containers including Docker supp...

.NET desktop development
Build WPF, Windows Forms, and console applications using C#, Visual Basic, and F# with .NET and .NET Frame...

Universal Windows Platform development
Create applications for the Universal Windows Platform with C#, VB, or optionally C++.

Desktop development with C++
Build modern C++ apps for Windows using tools of your choice, including MSVC, Clang, CMake, or MSBuild.

Data storage and processing
Connect, develop, and test data solutions with SQL Server, Azure Data Lake, or Hadoop.

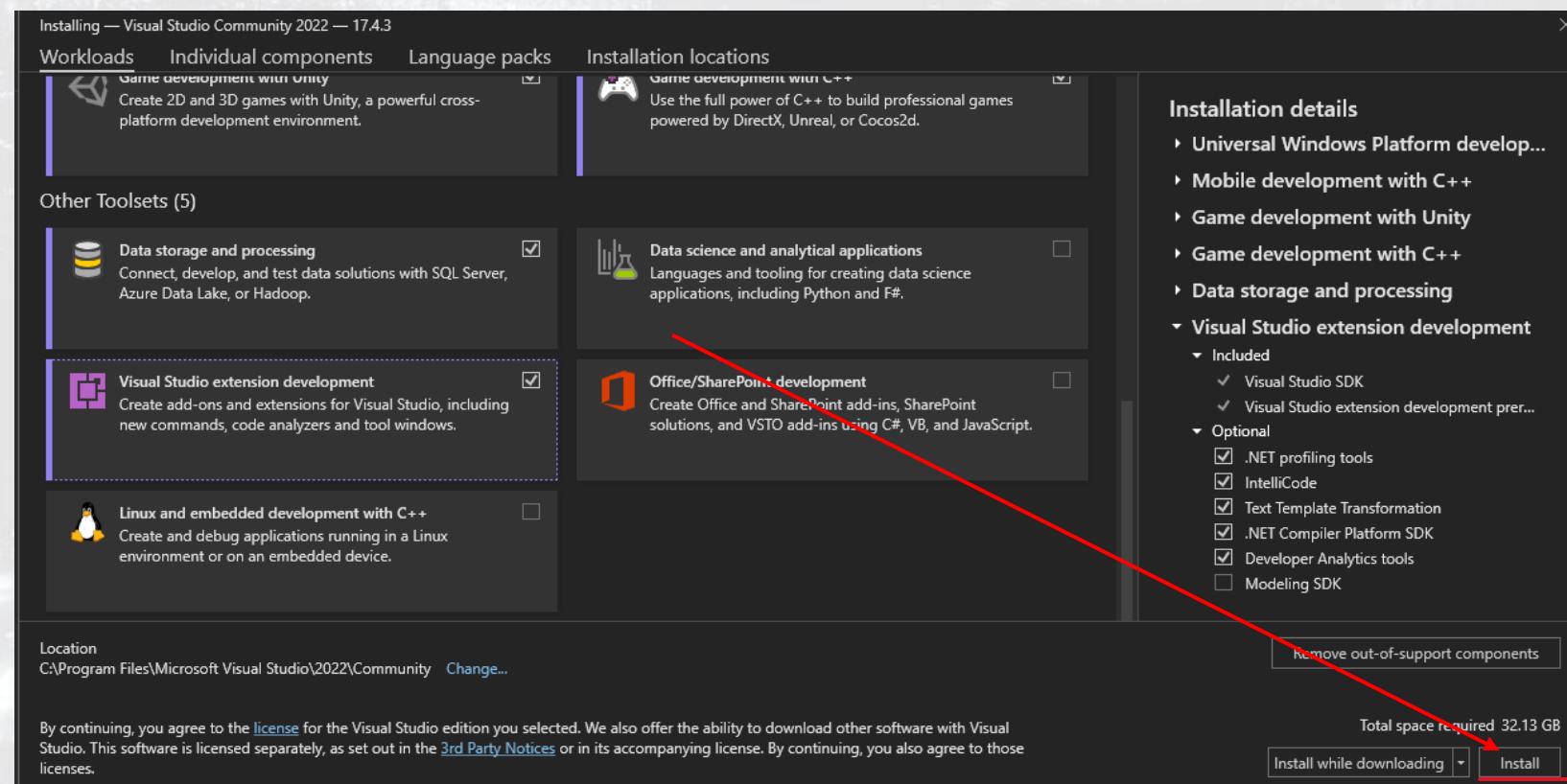
Visual Studio extension development
Create add-ons and extensions for Visual Studio, including new commands, code analyzers and tool windows.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

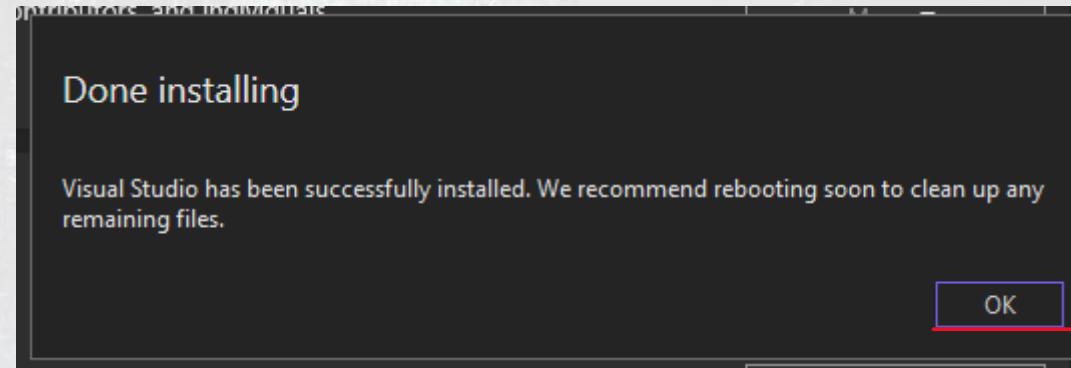




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INSTALLING VISUAL STUDIO

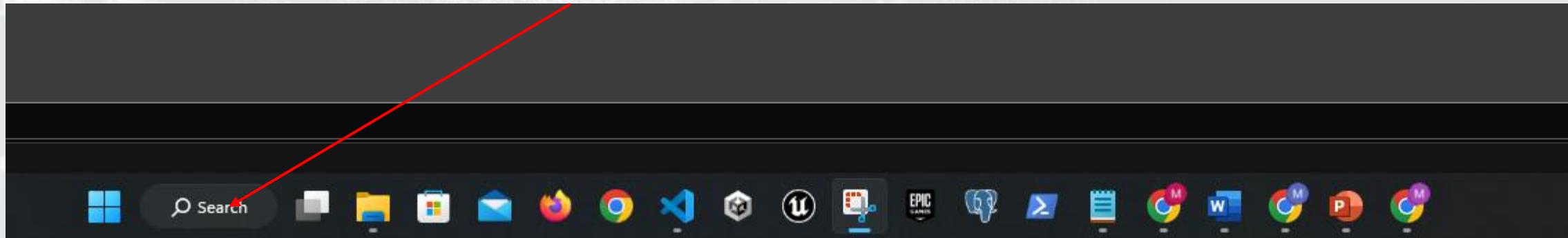




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO

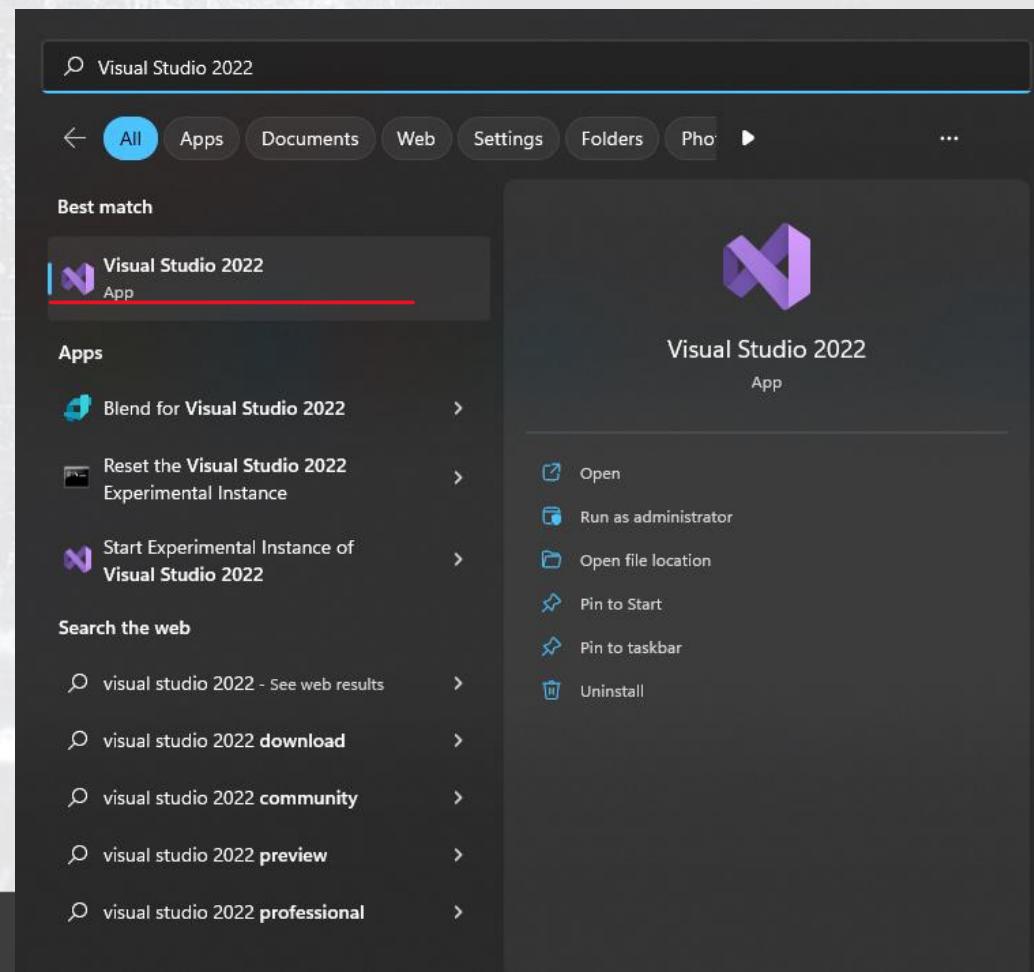




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO

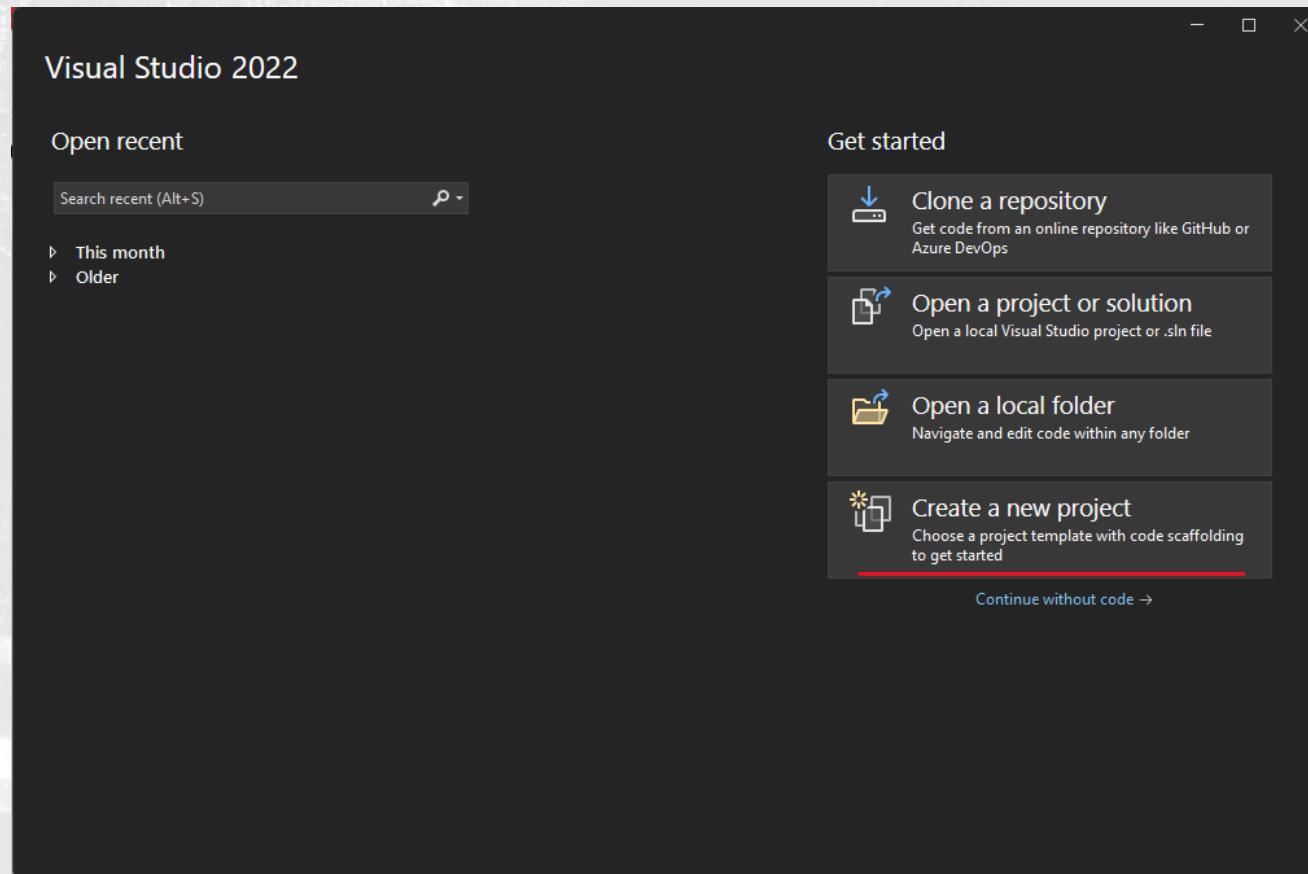




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO





INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO

Create a new project

Recent project templates

A list of your recently accessed templates will be displayed here.

Search for templates (Alt+S)

All languages All platforms All project types

- All languages
- C#
- C++
- F#
- JavaScript
- Python
- Query Language
- TypeScript
- Visual Basic**
- XAML

a command-line application that can run on .NET on Windows,
macOS Windows Console

Blazor Server App

A project template for creating a Blazor server app that runs server-side inside an ASP.NET Core app and handles user interactions over a SignalR connection. This template can be used for web apps with rich dynamic user interfaces (UIs).

C# Linux macOS Windows Blazor Cloud Web

ASP.NET Core Web API

A project template for creating an ASP.NET Core application with an example Controller for a RESTful HTTP service. This template can also be used for ASP.NET Core MVC Views and Controllers.

C# Linux macOS Windows Cloud Service Web WebAPI

Class Library

A project for creating a class library that targets .NET or .NET Standard

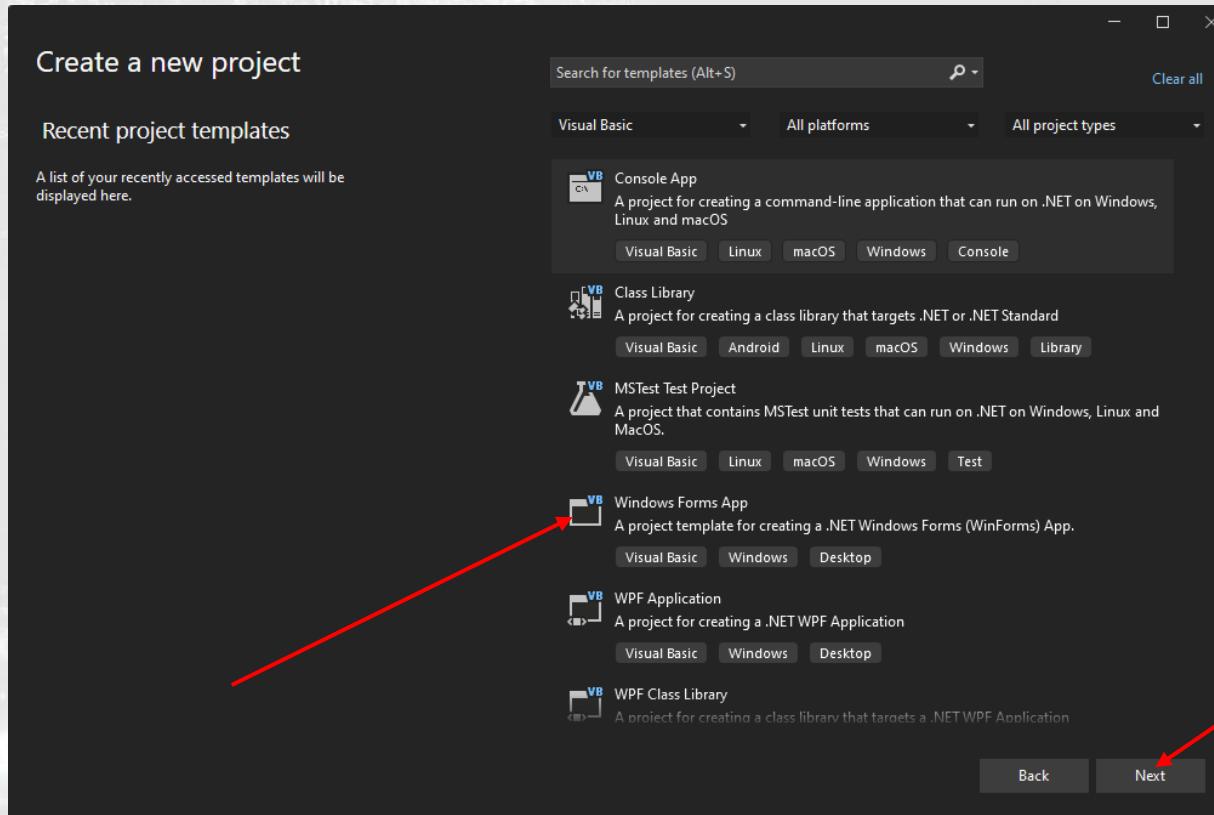
Back Next



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO

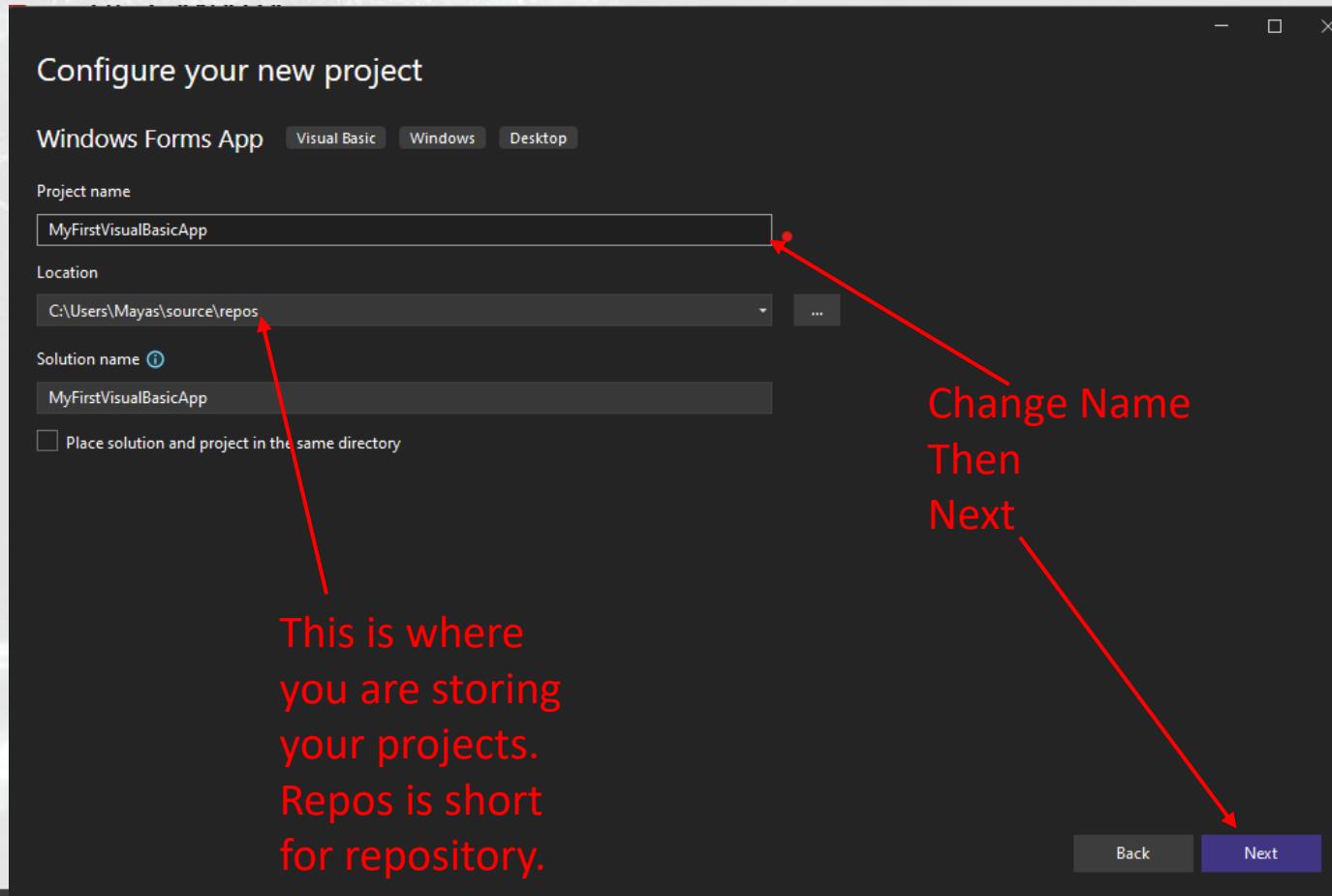




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO

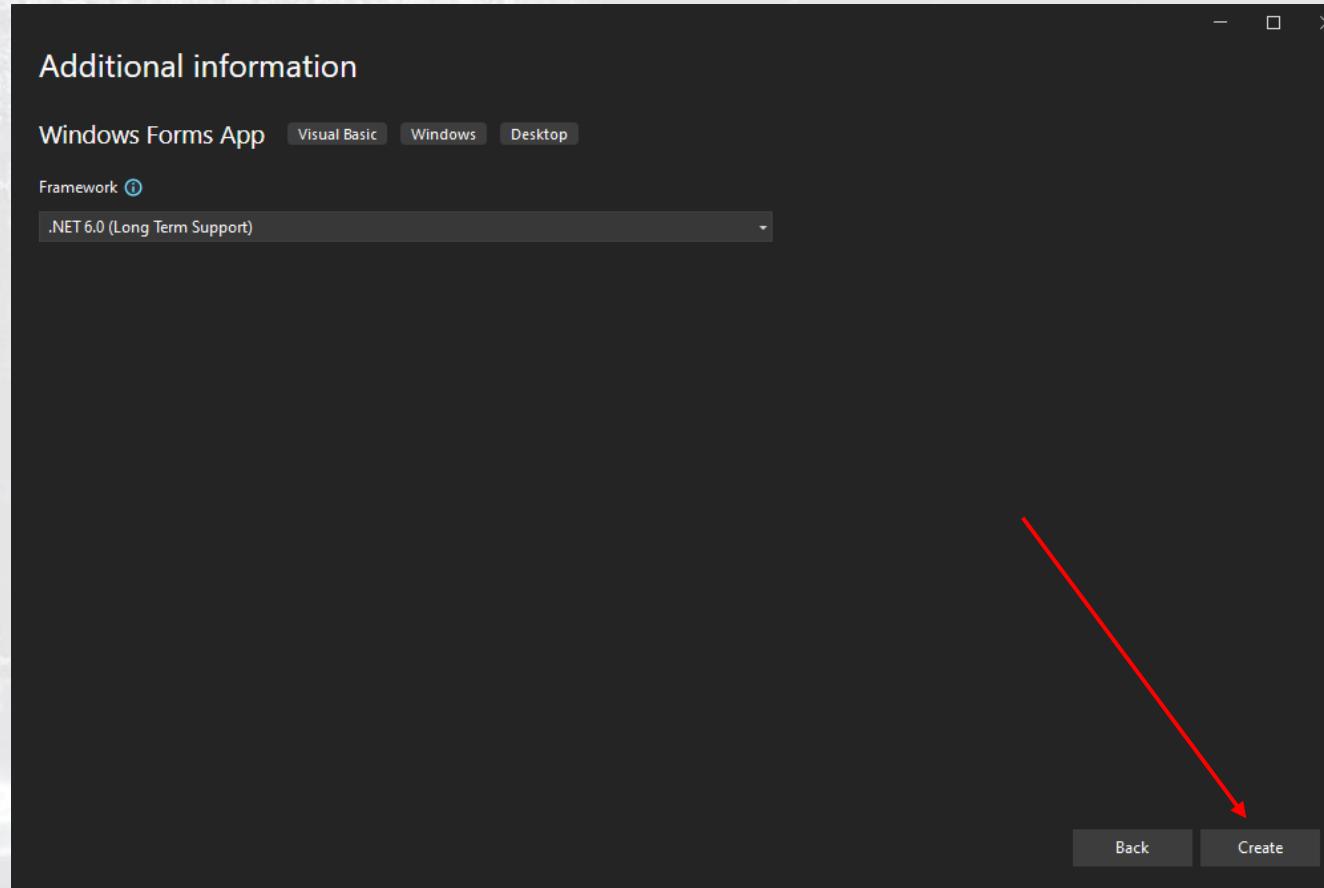




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO

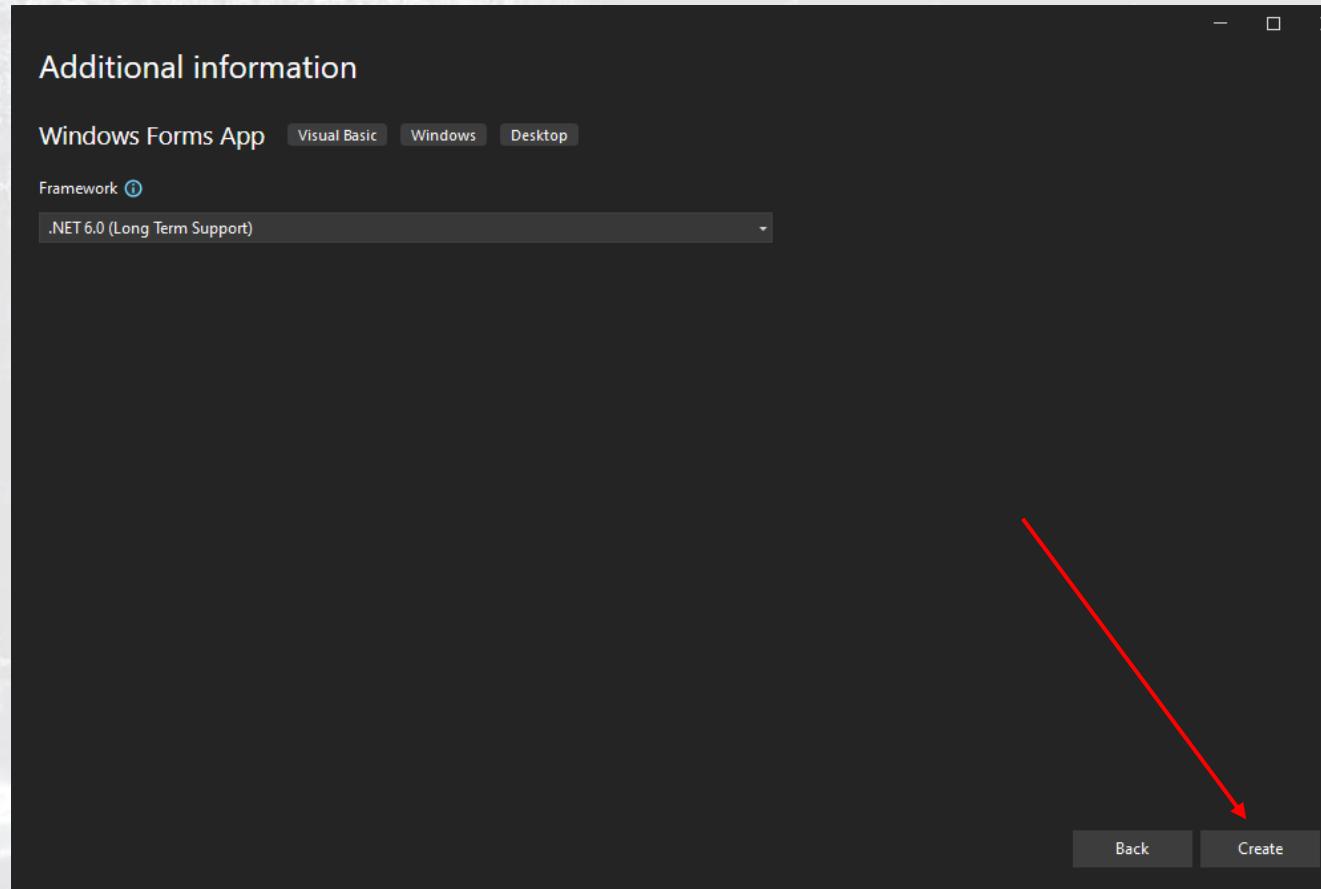




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPENING VISUAL STUDIO





INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

USING VISUAL STUDIO

The screenshot shows the Visual Studio IDE interface. In the center, there's a window titled "Form1.vb [Design]". Inside this window, the text "Click to Run Code" is displayed. To the right of this window, a yellow arrow points upwards towards the "Debug" dropdown menu in the top toolbar, which contains options like "Any CPU", "Run", and "Stop".

On the right side of the interface, the "Solution Explorer" window is open, showing a hierarchical tree structure:

- Solution MyFirstVisualBasicApp (1 of 1 project)
 - MyFirstVisualBasicApp
 - Dependencies
 - My Project
 - VB ApplicationEvents.vb
 - Form1.vb
 - VB Form1.Designer.vb
 - Form1.resx

A blue arrow points from the text "This is your project" to the "MyFirstVisualBasicApp" node in the Solution Explorer. A green arrow points from the text "This is the form you are working on" to the "Form1.vb" node.

At the bottom left, a yellow arrow points from the text "Double click to see your code" to the bottom edge of the "Form1.vb [Design]" window.

At the bottom right, there are several status bar items: "Ready", "Add to Source Control", "Select Repository", and a refresh icon.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

USING VISUAL STUDIO

The screenshot shows the Visual Studio IDE interface. The title bar reads "MyFirstVisualBasicApp". The menu bar includes File, Edit, View, Git, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help, and Search (Ctrl+Q). The toolbar has icons for file operations like Open, Save, and Print.

The main editor area displays the code for "Form1.vb [Design]". The code is:

```
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ' Write code here to run on form load
    End Sub
End Class
```

A red arrow points from the text "This is Your Form Tab" to the tab bar where "Form1.vb [Design]" is selected. A yellow arrow points from the text "This is your code tab" to the code editor area. A blue arrow points from the text "Write code here to run on form load" to the code block.

The Solution Explorer on the right shows the project structure:

- Solution: MyFirstVisualBasicApp (1 of 1 project)
 - MyFirstVisualBasicApp
 - Dependencies
 - My Project
 - VB ApplicationEvents.vb
 - Form1.vb
 - Form1.Designer.vb
 - Form1.resx



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

USING VISUAL STUDIO

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

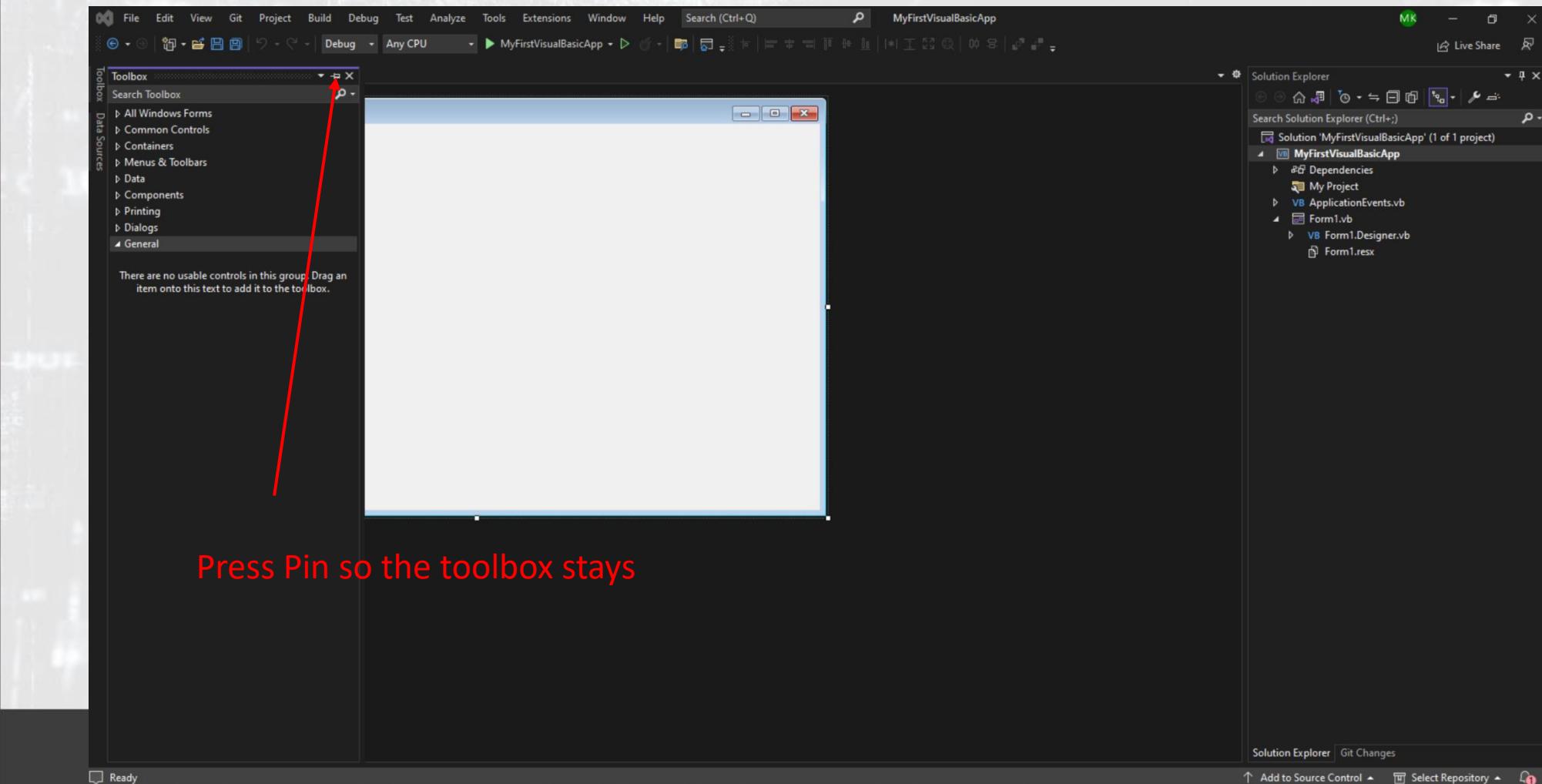
The screenshot shows the Visual Studio 2022 interface. On the left, the 'View' menu is open, displaying various options like 'Code', 'Designer', and 'Toolbox'. The 'Toolbox' option is highlighted with a red arrow pointing to it. To its right, the main workspace shows a project named 'MyFirstVisualBasicApp' in the Solution Explorer. The code editor window is empty. The status bar at the bottom indicates 'Ready'.

Click View >> Toolbox



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

USING VISUAL STUDIO



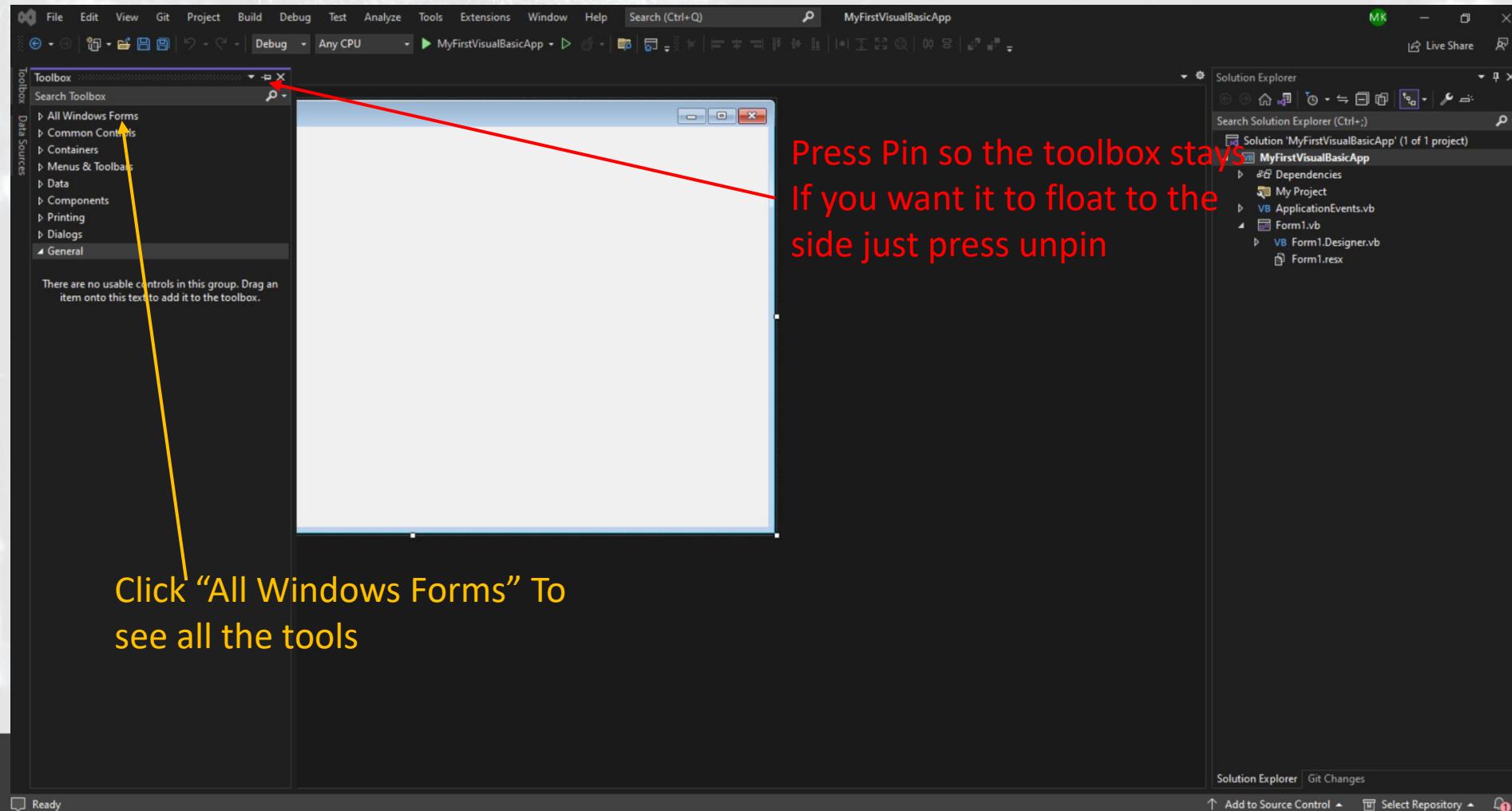
475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

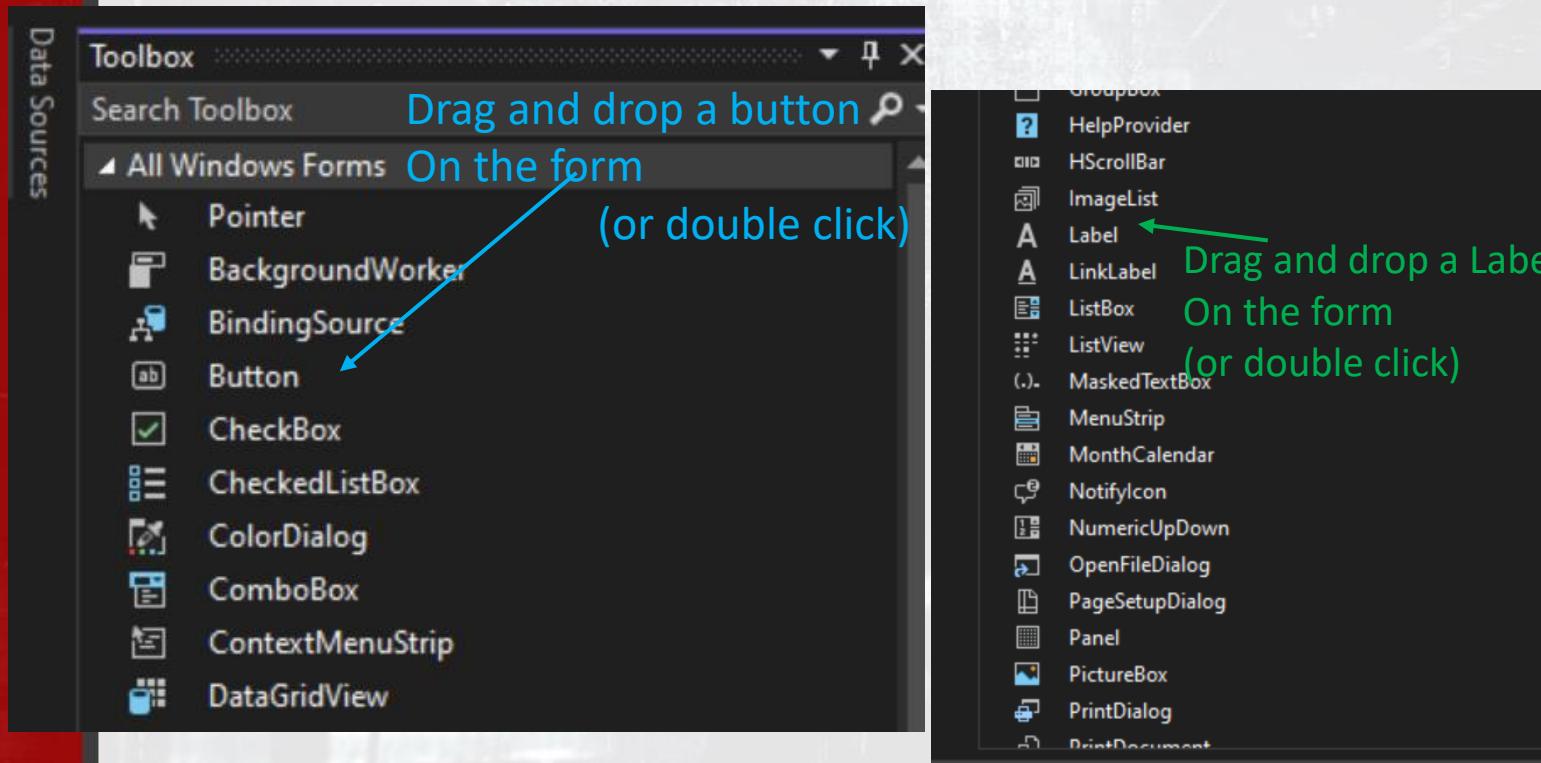
USING VISUAL STUDIO



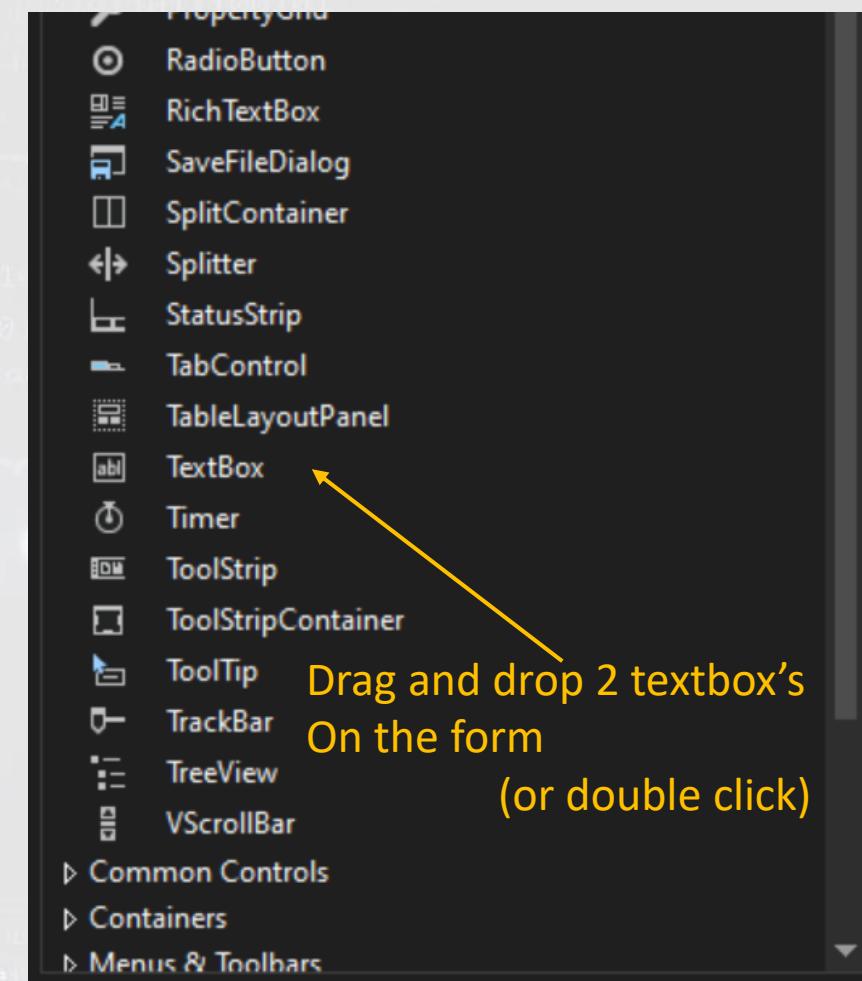


INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

USING VISUAL STUDIO



475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA





INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

USING VISUAL STUDIO

1) Right Click on Button select properties

2) Change Button1 to Sum

3) Single click on Label1

4) Change text to Num1

3) Single click on Label2

4) Change text to Num2

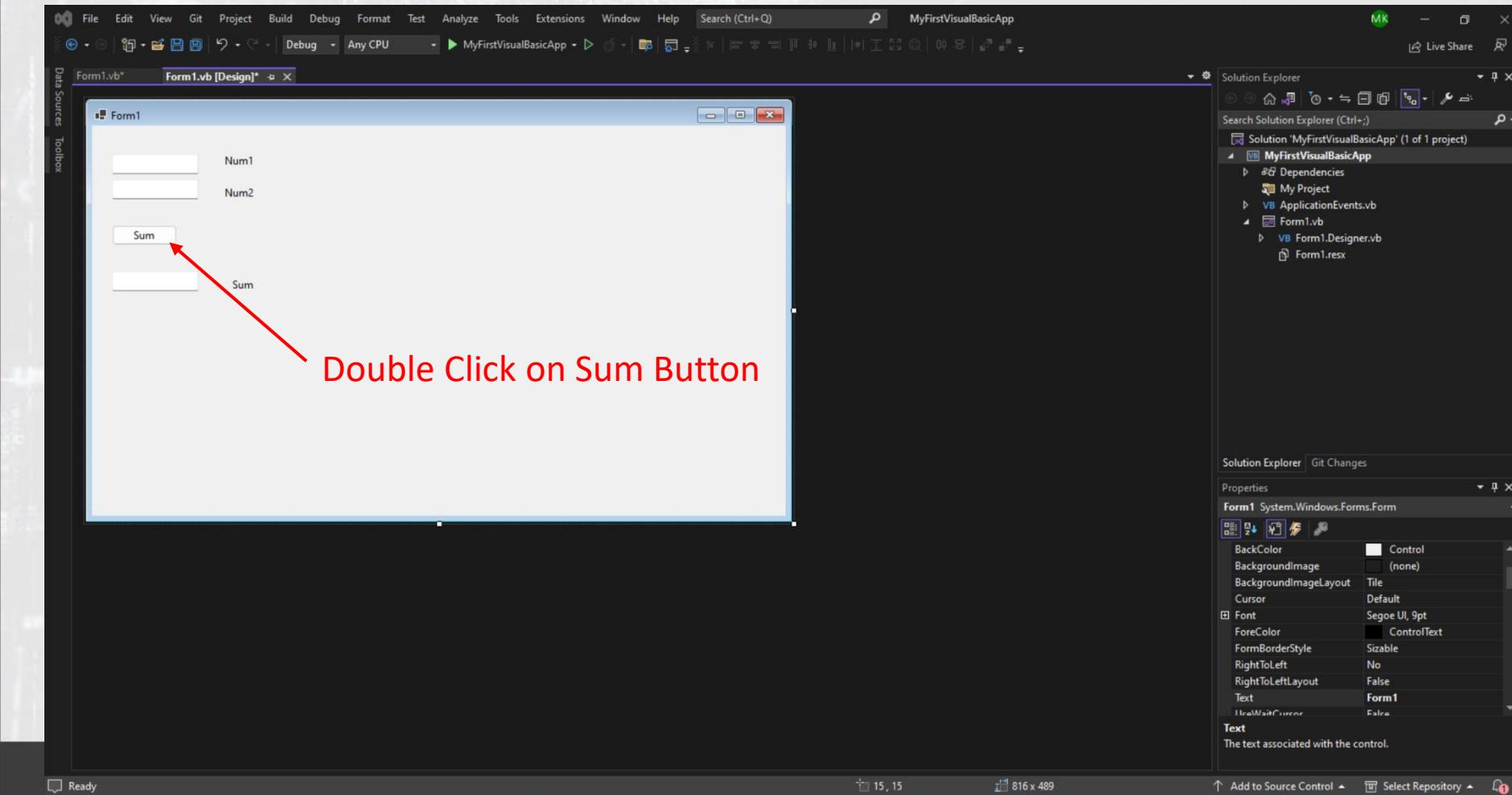
Properties window showing the 'Text' property of Button1 set to 'Button1'.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

USING VISUAL STUDIO

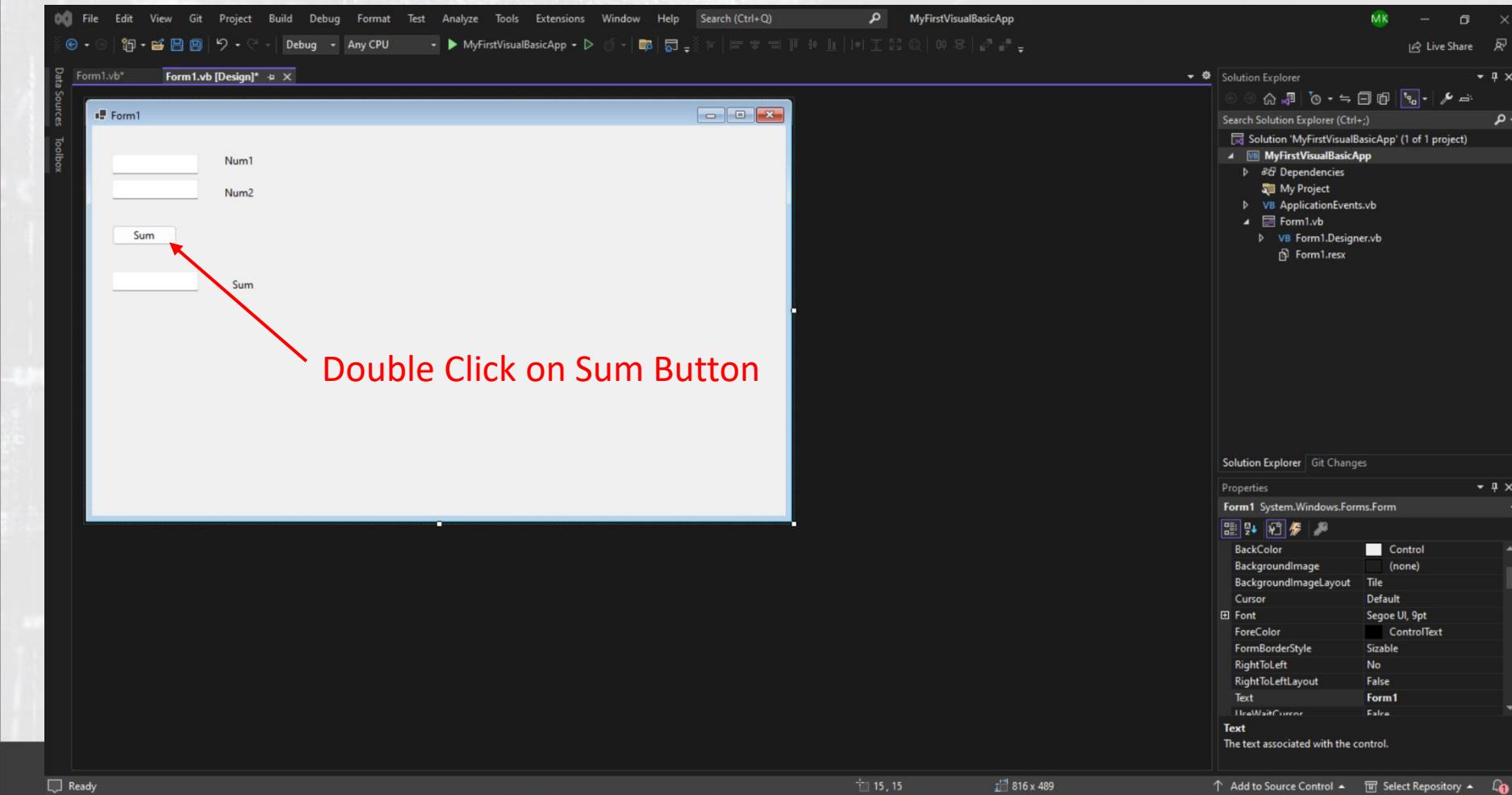




INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

USING VISUAL STUDIO





USING VISUAL STUDIO

The screenshot shows a code editor in Visual Studio displaying C# code for a Windows Form application. The code defines a class named Form1 with two methods: Form1_Load and Button1_Click. The Form1_Load method handles the MyBase.Load event. The Button1_Click method handles the Button1.Click event and contains logic to add two integers from TextBox1 and TextBox2 and display the result in TextBox3. A red box highlights the code within the Button1_Click method.

```
1 reference
1  Public Class Form1
2      0 references
3          Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
4              End Sub
5
6      0 references
7          Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
8              Dim num1 As Integer
9              Dim num2 As Integer
10             num1 = Convert.ToInt32(TextBox1.Text)
11             num2 = Convert.ToInt32(TextBox2.Text)
12             Dim sum As Integer
13             sum = num1 + num2
14             TextBox3.Text = sum.ToString()
15         End Sub
16     End Class
```

Write your code in the button1_click function. This is an event function built into Windows forms



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OPERATORS REFERENCE



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ASSIGNMENT OPERATORS

- = assignment
- $X = 5$
- $X = \text{"Happy"}$



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARITHMETIC OPERATORS

- + add
- - subtract
- * multiply
- / division
- Mod Modulus



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SHORTHAND OPERATORS

- $x += y$ is shorthand for $x = x + y$
- $x -= y$ is shorthand for $x = x - y$
- $x /= y$ is shorthand for $x = x / y$
- $x *= y$ is shorthand for $x = x * y$



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOGICAL OPERATORS

- logical operators
- **And** AND ($x = 5$ And $y = 10$)
- **Or** OR ($x = 5$ Or $x = 10$)
- **Not** NOT (**Not** x)



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

COMPARISON OPERATORS

- = equal
- <> not equal
- > greater than
- < less than
- >= greater than or equal to
- <= less than or equal to



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VARIABLES



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ASSIGNMENT OPERATORS

- = assignment
- X = 5
- X = “Happy”

```
Dim MyNumber As Integer = 55
Dim MyString As String = "Hello World"
Dim MyBool As String = True
```



VARIABLES

Dim allows us to declare and allocate memory for a new variable.

Dim x or Dim x as Integer (or string or boolean)

- When we don't give a value to a variable it is called "unassigned". This can create unpredictable behavior in Visual Basic

Dim x as Integer = 5

- is giving a value to x

Public x as Integer = 5

- Public is less commonly used; it is global scoped which means it can be accessed in most places in our program. It is preferable to use Dim in many cases to prevent scope issues

VB is a *strongly typed* language which means we must declare the specific type like integer, string or Boolean. It will sometimes assume the type, but it is best to declare the type you intend to use.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VARIABLE NAMES

- Good
 - camelCase
 - good_name
- Bad
 - \$badName
 - 5 badName
 - & badName
 - Bad-name



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VARIABLE

Add 3 new buttons to your form difference (subtract), product (multiply) and quotient (divide) use the same three Textboxes you used to for sum (add).

1) Input Difference = 20, 5

Output = 15

2) Input product = 20, 5

Output = 100

3) Input quotient = 20, 5

Output = 4



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SEMICOLON



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SEMICOLON

At the end of a line of code, you will never see “;”
This will cause issues. Do NOT use semi colons



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

IDE

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

WHAT IS YOUR IDE TRYING TO TELL YOU?

You may see a red underline; this is like spell check in things like Microsoft Word. This is more important than a spelling error because it means your program won't run

```
Dim MyString~happy As String
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

WHAT IS YOUR IDE TRYING TO TELL YOU?

When you see the red underline, hover over the red line and it will give you suggestions on how to fix the error, though it might not always be the best suggestion, read it you can click on the BC30205 and it will send you to documentation.

```
Dim MyString~happy As String
```

```
Dim MyString~happy As String
```

[?] (local variable) MyString As Object

BC30205: End of statement expected.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OBJECTS PROPERTIES AND METHODS



PROPERTIES

- In Windows forms we use built in objects.
- TextBox1 is and object, Button2 is an object.
- These objects come with different properties.
- If I say; TextBox1.Text, I am accessing the text value in TextBox1
- You can do the same with Button1.Text. This may be less valuable in most cases but as an exercise try to change the text value of the button to “Wow!” when you click on it.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

PROPERTIES

- Button text change:

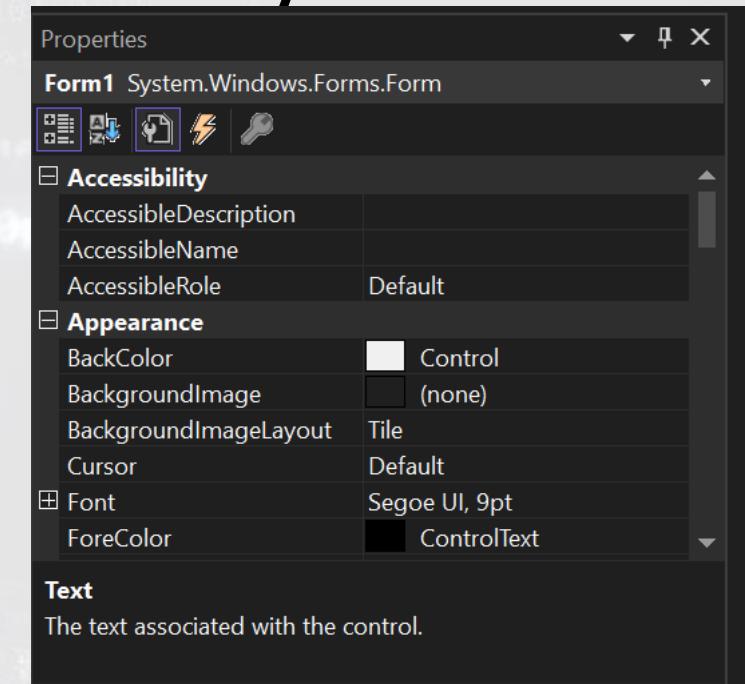
0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Button1.Text = "Wow!"
End Sub
```



PROPERTIES

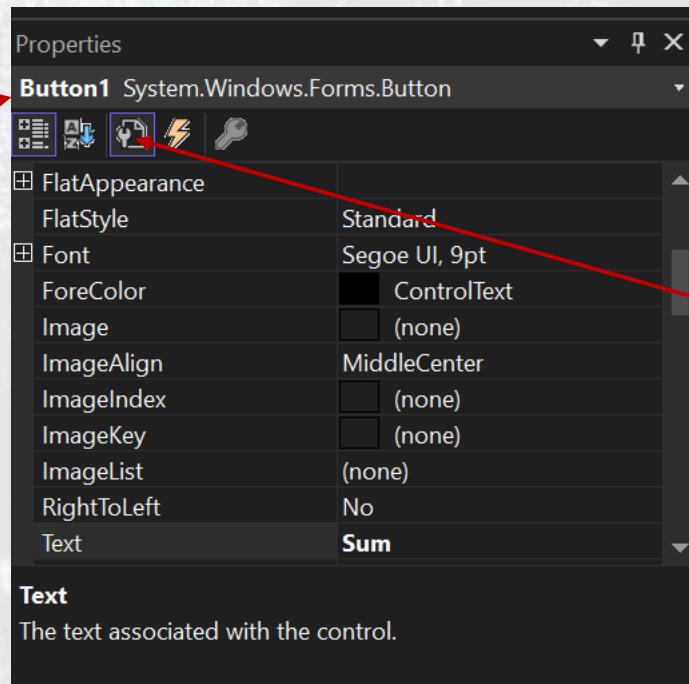
- The properties menu can be found at the bottom right of your screen. We used this already to change the labels and button text





PROPERTIES

This is the object we are working on. You can see it's Button1. If you click on another object like TextBox1 it will show up there



This is where we can find the properties associated with the object we have selected. When clicked on, we can change things like Text, Color, Images.

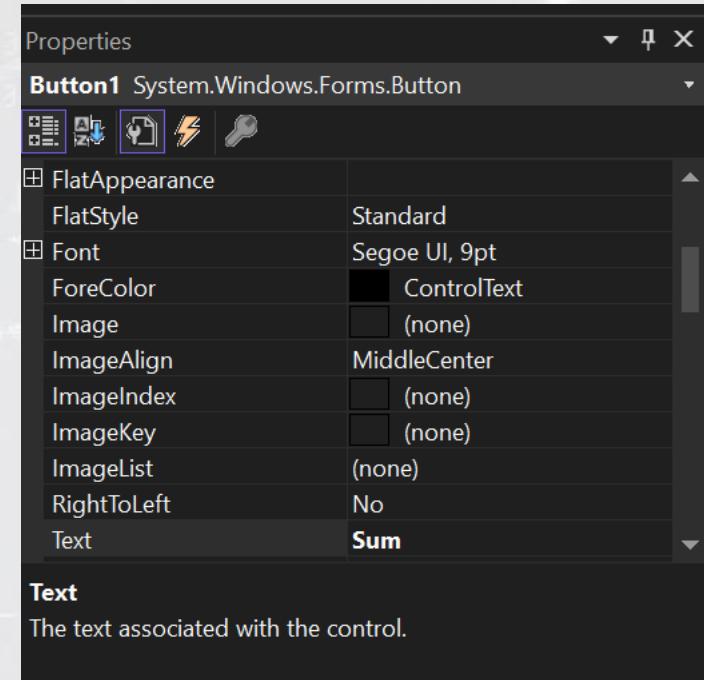


INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

PROPERTIES

Exercise:
Change the color of your Button to
Blue in the properties menu.

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

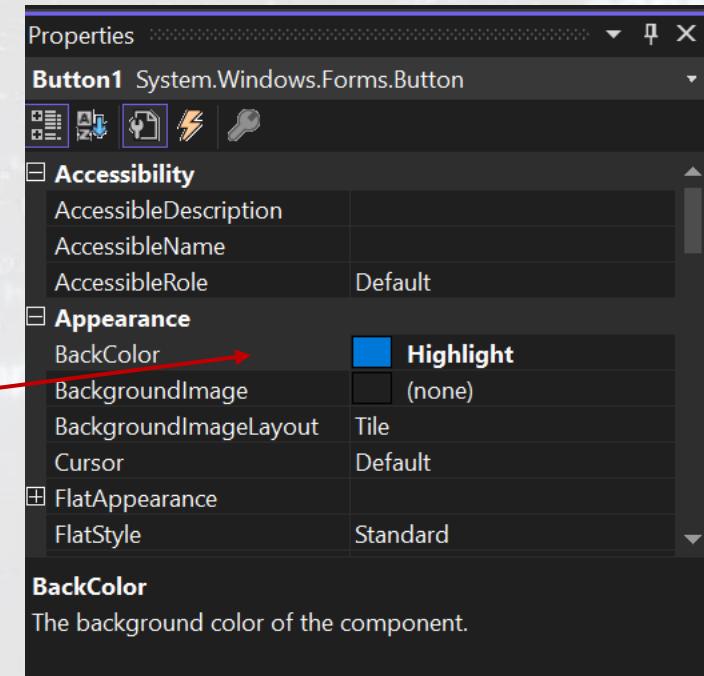




PROPERTIES

Answer:

Click on the column beside BackColor,
And choose the color you want.





PROPERTIES

Now try to do this programmatically:

Hint 1: If you can't figure it out use google.

Hint 2: If google doesn't help, you have to use another object that comes with windows forms called color.

Write:

Color.TheColorYouWant

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    ' Your code here
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

PROPERTIES

Answer:

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Button1.BackColor = Color.Blue
End Sub
```

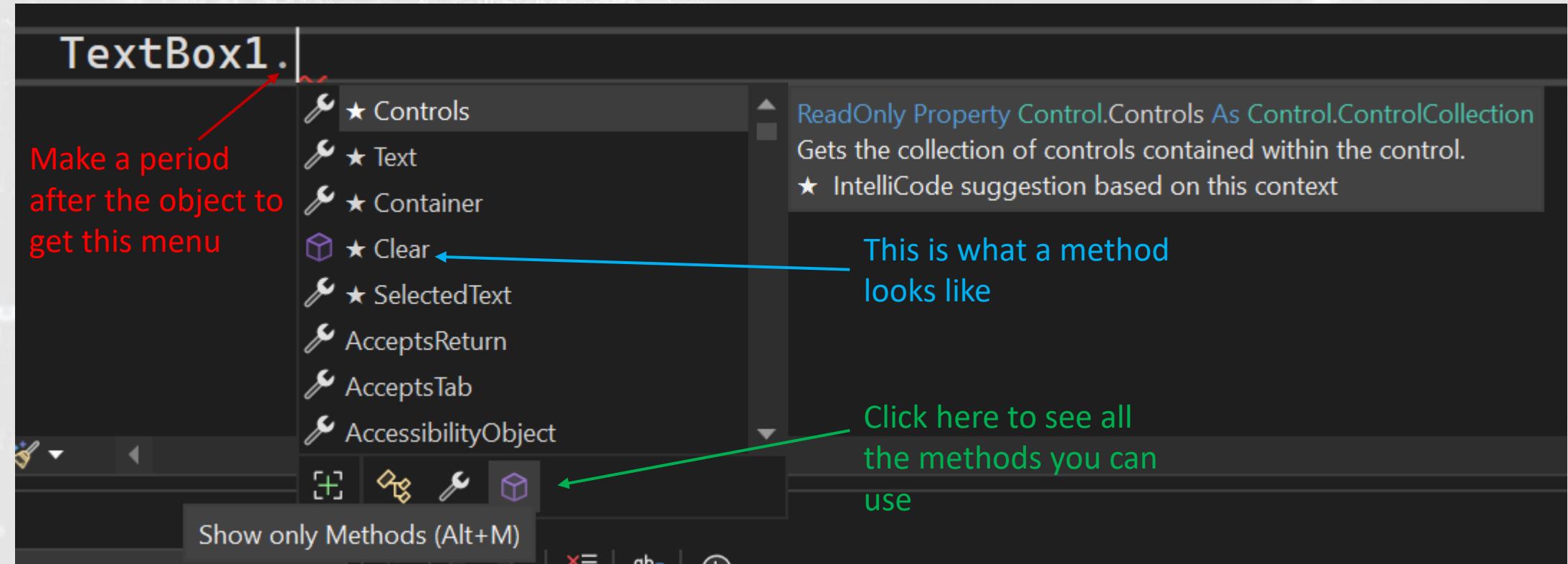


FUNCTIONS/METHODS

- Objects don't just come with properties, they also come with functions (methods)
- Just like you can have `texbox1.text` you can have `TextBox1.Clear()`. This will clear the textbox. Give it a try!
- Make a button that clears the textboxes in your form.



FUNCTIONS/METHODS





FUNCTIONS/METHODS

- Other Items like variables have methods you can add. Some common ones are:
- ToLower()
- ToString()
- Convert.ToInt____() you can fill in what you want to convert



FUNCTIONS/METHODS

```
Dim MyVariable As Integer = 55
MyVariable.ToString()
```

Hover on the
function to see
what it does

Function Integer.ToString() As String (+ 5 overloads)

Converts the numeric value of this instance to its equivalent string representation.

Returns:

The string representation of the value of this instance, consisting of a negative sign if the value is negative, and a sequence of digits ranging from 0 to 9 with no leading zeroes.

This shows what is
returned from the
function



FUNCTIONS/METHODS

```
Dim MyVariable As String = "It's a beautiful day"  
MyVariable.Substring(1)
```

You can see what parameters are required. This one needs a start index. I started at 1 which is t

Function String.Substring(startIndex As Integer) As String (+ 1 overload)

Retrieves a substring from this instance. The substring starts at a specified character position and continues to the end of the string.

Returns:

A string that is equivalent to the substring that begins at startIndex in this instance, or String.Empty if startIndex is equal to the length of this instance.

Exceptions:

ArgumentOutOfRangeException

This returns everything from the start index to the end.
You can read more in the documentation



FUNCTIONS/METHODS

```
Dim MyVariable As String = "It's a beautiful day"  
MyVariable.Substring(1)
```

Print this in a text
box to see what
you get

Function String.Substring(startIndex As Integer) As String (+ 1 overload)

Retrieves a substring from this instance. The substring starts at a specified character position and continues to the end of the string.

Returns:

A string that is equivalent to the substring that begins at startIndex in this instance, or String.Empty if startIndex is equal to the length of this instance.

Exceptions:

ArgumentOutOfRangeException



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

INPUT/OUTPUT



BASIC INPUT/OUTPUT

- Common Input/Outputs are:
 - Textbox (I/O)
 - RichTextbox (I/O)
 - MessageBox.Show() (I/O)
- Textbox are generally used for one line of text going in or out.
- RichTextbox are generally used for multiple lines of text going in or out.
- Messageboxes are commonly used to alert users, They can be used for inputs in specific circumstances like returning different variations of Yes No



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

BASIC INPUT/OUTPUT

- When using both Textboxes and Richtextboxes, treat it like a variable if you want to add more data to the same box:
- `TextBox1.Text = MyString`
- `TextBox1.Text += MySecondString`



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

MORE ADVANCED I/O

There are more Inputs and outputs like different ways to display tabular data and graphs. We will take at these later in the course.



I/O EXERCISE

- 1) Make a New Project like we did in the early slides.
- 2) Call it 2_InputAndOutput
- 3) Add the textboxes labels and button you think you need.
- 4) Take in two strings into variables. One called hello, which takes in “Hello,” one called world which takes in “ world” and concatenate them. Display them in MessageBox.Show() on a button click
- 5) Take in a string “William Blake Said: ” concatenate with another string: “He who binds to himself a joy Does the winged life destroy; But he who kisses the joy as it flies Lives in eternity's sun rise.” Display in a RichTextbox on button click



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

I/O EXERCISE OUTPUT

Input Output

Exercise 1:

Hello, Greeting

World Place

Concatenate MessageBox

Exercise 2:

William Blake Author

Quote

He who binds to himself a joy Does
the winged life destroy; But he who
kisses the joy as it flies Lives in
eternity's sun rise.

Concatenate RichTextBox

William Blake Said: He who binds to
himself a joy Does the winged life
destroy; But he who kisses the joy as it
flies Lives in eternity's sun rise.

Author + Quote



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

I/O EXERCISE CODE

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim Hello As String = TextBox1.Text
    Dim World As String = TextBox2.Text
    MessageBox.Show(Hello + World)
End Sub
```

0 references

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    Dim Author As String = TextBox3.Text
    Dim Quote As String = RichTextBox1.Text
    RichTextBox2.Text = Author + " Said: " + Quote
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

COMMENTS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

COMMENTS

- Comments are declared one sing quote ‘
- You can use short cuts (control + k + c) to block out a line or any group you highlighted
- You can uncomment by (ctrl + k + u)



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

DATA TYPES



DATA TYPE IN GENERAL

Character

single

a, b, c

'a' 'b' 'c' '%' '7' '|'

string

cat (constant)

"cat"

Numbers

Integer

1,2,3

Real

0, 1 || false, true

2.75 != 27.5 != 275.0

Boolean



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VB COMMON PRIMITIVE DATA TYPES

- Integer
- Double
- Long
- String
- Char
- Boolean

For all data types and memory requirements read:

<https://learn.microsoft.com/en-us/dotnet/visual-basic/language-reference/data-types/>



VB DATA TYPES

- Visual Basic is a strongly typed language; this means we must define the data types we are using. It makes it easy when reading code and prevents type errors.
- When we take in a “5”, which is a character and a 5 which is a number, they are different. We can define them as different when we create our variables.



VB DATA TYPES

1. Make a new project called 3_DataTypes
2. On your form add labels, textboxes and buttons
3. Make a two separate buttons that adds two numbers. One will add integer numbers, one will add numbers that are characters(or strings)
4. Input Integers = 5, 5
Output = 10
5. Input Characters= “5”, “5”
Output = “55”



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VB DATA TYPES

- Even though Visual Basic is a strongly typed language; It is not as strict as other languages as C++ or even JavaScript.
- If you compare “5” to 5 it will say they are equal in value. It compares the value not the type and value



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VB DATA TYPES - OUTPUT

Data Types

Exercise 1:

<input type="text" value="5"/>	Integer 1
<input type="text" value="5"/>	Integer 2
<input type="button" value="Integer Add"/>	
<input type="text" value="10"/>	Int Sum

Exercise 2:

<input type="text" value="5"/>	Char 1
<input type="text" value="5"/>	Char 2
<input type="button" value="Character Add"/>	
<input type="text" value="55"/>	Char Sum



VB DATA TYPES - CODE

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim IntNum1 As Integer = TextBox1.Text
    Dim IntNum2 As Integer = TextBox2.Text
    TextBox3.Text = IntNum1 + IntNum2
End Sub
```

0 references

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    Dim CharNum1 As String = TextBox4.Text
    Dim CharNum2 As String = TextBox5.Text
    TextBox6.Text = CharNum1 + CharNum2
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

DATA TYPES NUMBERS



NUMBERS

- Visual Basic has a multiple types of numbers
 - Integer (int32)
 - Double
 - Decimals
 - Long
 - Sbyte
 - Single
 - UInteger
 - Ulong
- We will cover the two most commonly used, Integer and Double, It will be up to you to read further and decide when to use which type



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

NUMBERS - INTEGERS

- Integer (int32) are 4bytes number that can be from negative to positive. This is the spectrum -2,147,483,648 through 2,147,483,647**
- Integers can not have a decimal point. So 44.2 will round down to 44 and 44.5 will round up to 45

** is from <https://learn.microsoft.com/en-us/dotnet/visual-basic/language-reference/data-types/>



NUMBERS - INTEGERS

- Doubles are 8bytes in size and can go from negative to positive.
- -1.79769313486231570E+308 through -
4.94065645841246544E-324 † for negative values;**
- 4.94065645841246544E-324 through
1.79769313486231570E+308 † for positive values**
- Doubles can have a decimal point; 44.2 will be 44.2

** is from <https://learn.microsoft.com/en-us/dotnet/visual-basic/language-reference/data-types/>



NUMBERS - INTEGERS

- When it comes to using other types you have to think what will be the most efficient on memory. Will you need to allocate an 8 byte double when using whole numbers that go from 1-100? Or should you use an Integer which takes bytes? Or maybe an Sbyte might be better? Depends on if you will need to expand the program to have bigger numbers. Be sagacious and decern what is best when.

** is from <https://learn.microsoft.com/en-us/dotnet/visual-basic/language-reference/data-types/>



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

DATA TYPES STRINGS



STRINGS

- There are two different ways to make strings in VB:
 1. Use double quotes. This is great work for strings that don't need to have data added in the middle. If you need to concatenate, take two strings and add them "Hello, " + "World"
 2. Make a Template Literal. This is where you add the dollar sign in front of the first ". Then you nest the variables in {}. This allows you to format your strings with ease.
- Next page will have demonstrations



STRINGS

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim Hello As String = TextBox1.Text
    Dim World As String = TextBox2.Text
    TextBox3.Text = Hello + World ← Regular String being concatenated
    TextBox4.Text = $"Hello}, {World}"
End Sub
```

Template Literal



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

STRINGS

Data Types

Strings:

Hello	Hello
World	World
Concatenate	
HelloWorld	Without Template Literal
Hello, World	With Template Literal

Regular String being concatenated

Template Literal



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

STRINGS EXERCISE 1

Make a prompt that takes a first name, last name, and age.

Get it to print in the console in one line.

Hello FirstName LastName, you are Age years old.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

STRINGS EXERCISE 2

- Make a form that fills in the blanks:

Hi! Last week I went _____. My friend _____ and I had so much _____. There were _____ and _____. When we saw it we felt really _____. You should come next time!



STRINGS (ARRAYS OF CHARACTERS)

- Each part of a string has an index number

myCode = "Code"

0123

- myCode(3) is not equal to d
- it is e
- Variable.length is used to calculate the number of characters in a string including spaces
- () Is called subscript operator (myCode address + (char size * index))
- The subscript operator calculates the amount of bits or bytes between two data types stored in an array to find the address.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

STRINGS EXERCISE 3

- What is the value:
- Let bday = “Happy Birthday”
- MessageBox.Show(bday(7))



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

STRINGS EXERCISE 3

- What is the value:
- Let bday = “Happy Birthday”
- MessageBox.Show(bday(7))
- Answer:
- i



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

DATA TYPES

BOOLEANS



BOOLEANS

- Booleans are true or false statements.
- You can use Booleans in your code as variables

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim myBool As Boolean = True
    MessageBox.Show(myBool)
End Sub
```

- This will print true in the message box



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

BOOLEANS

- Change the Boolean in the below image so it prints out false:

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim myBool As Boolean = True
    MessageBox.Show(myBool)
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

BOOLEANS

- Because Booleans are true or false, there are two operator categories we can use to check Booleans
 1. Comparison operator you can compare numbers, strings and Booleans and return true and false
 2. Logical operators can be used to add logic to a Boolean statement



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

COMPARISON OPERATORS

- = equal
- <> not equal
- > greater than
- < less than
- >= greater than or equal to
- <= less than or equal to



COMPARISON OPERATOR

How will these be displayed in the message box:

1.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim myInt As Integer = 55
    Dim myInt2 As Integer = 40
    MessageBox.Show(myInt = myInt2 + 15)
End Sub
```

2.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim mystring As String = "55"
    Dim myInt As Integer = 55
    MessageBox.Show(myInt <> mystring)
End Sub
```

3.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim MyInt1 As String = 1
    Dim MyInt2 As Integer = 55
    MessageBox.Show(MyInt1 < MyInt2)
End Sub
```

4.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim MyInt1 As String = 1
    Dim MyInt2 As Integer = 55
    MessageBox.Show(MyInt1 > MyInt2)
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOGICAL OPERATORS

- **And** AND ($x = 5$ And $y = 10$)
- **Or** OR ($x = 5$ Or $x = 10$)
- **Not** NOT (Not x)



LOGICAL OPERATOR

How will these be displayed in the message box:

1.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim MyInt1 As Integer = 1
    Dim MyInt2 As Integer = 55
    MessageBox.Show((MyInt1 > MyInt2) And (MyInt1 <> 42))
End Sub
```
2.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim MyInt1 As Integer = 1
    Dim MyInt2 As Integer = 55
    MessageBox.Show((MyInt1 > MyInt2) Or (MyInt1 <> 42))
End Sub
```
3.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim MyInt1 As Integer = 1
    Dim MyInt2 As Integer = 55
    MessageBox.Show(Not (MyInt1 > MyInt2) And Not (MyInt1 = 1))
End Sub
```
4.

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim MyInt1 As Integer = 1
    Dim MyInt2 As Integer = 55
    MessageBox.Show(Not (MyInt1 > MyInt2) And (MyInt1 = 1))
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

DATA TYPES

DATA WITHOUT DEFINED DATA



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

NOTHING

- X = Nothing;
 - The above will print a default value associated with the type:
 - Dim x as integer = Nothing will print out 0
 - Dim x as String = Nothing will print out __ (imagine the underscores are nothing)
 - Dim x as Boolean = Nothing will print out False



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

DEBUGGING



WHAT IS A BUG?

- Grace Hopper, who made the first compiler, is attributed with coining the term “Debugging”. This was when she had found a moth in one of the computers in the 1940's.
- The term was commonly used 50 years prior to Grace by Thomas Edison in relation to technical errors in machines.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

01001

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

TWO COMMON TYPES OF BUGS

1. Syntactical bugs
2. Logical bugs



SYNTACTICAL

- Syntactic bugs are caused by the programmer writing code in the wrong way. This often a small mistake like misspelling something or misplacing if statement endings. Most IDE's have built in features to tell you when you made a mistake, but it might not always catch the mistake.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOGICAL BUGS?

- Logical bugs are generally harder to find. Logical bugs are based on errors in the logic. There are tools that allow us to find these bugs such as:
 1. Reading error messages
 2. The Debugger



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

WHAT IS DEBUGGING?

- Debugging is a process we can use to go through our programs and find why it crashes, outputs wrong or doesn't do what we want.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

READING ERRORS

- Error messages are basically other programmers communicating with us that we have done incorrect.



READING ERRORS

The screenshot shows the Microsoft Visual Studio IDE with a VB.NET project named "MyFirstVisualBasicApp". The code editor displays the following code:

```
1 reference
Public Class Form1
    Public MyNumber As Integer
    0 references
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        End Sub
    0 references
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim MyInt1 As String = Nothing
        Dim MyInt2 As Integer = 55
        MessageBox.Show(MyInt1.ToLower())
        End Sub
    0 references
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        End Sub
End Class
```

A yellow arrow points from the text "This is telling us what line the error is on" to the line number 10 in the code editor, which contains the line: `MessageBox.Show(MyInt1.ToLower())`. A red arrow points from the text "Error message: you may not always understand what it means, so copy the message and paste it into google" to the error message window titled "Exception Thrown". The message reads: "System.NullReferenceException: 'Object reference not set to an instance of an object.'".

Call Stack:

- MyFirstVisualBasicApp.dll!MyFirstVisualBasicApp.Form1.Button1_Click(Object sender, System.EventArgs e) Line 10 [External Code]

Autos:

Name	Type	Value
Me	(MyFirstVisualBasicApp.Form1, Text: Data Types)	
MyInt1	Nothing	
MyInt2	Integer	55

Call Stack:

- Add to Source Control
- Select Repository



READING ERRORS

The issue is .ToLower()
doesn't work because
the string doesn't hold
a value

```
1 reference
1 Public Class Form1
2     Public MyNumber As Integer
3     0 references
4     Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
5
6 End Sub
7
8 0 references
9 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
10    Dim MyInt1 As String = Nothing
11    Dim MyInt2 As Integer = 55
12    MessageBox.Show(MyInt1.ToLower())
13
14 End Sub
15
16 End Sub
17
18 End Class
```

Exception Thrown

System.NullReferenceException: 'Object reference not set to an instance of an object.'

MyInt1 was Nothing.

Show Call Stack | View Details | Copy Details | Start Live Share session

Exception Settings

Break when this exception type is thrown

Call Stack

Name

MyFirstVisualBasicApp.dll!MyFirstVisualBasicApp.Form1.Button1_Click(Object sender, System.EventArgs e) Line 10
[External Code]

Autos

Name

Me

MyInt1

MyInt2

Type

MyFirstVisualBasicApp...
Nothing
String
Integer

Value

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output

Add to Source Control Select Repository



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

READING ERRORS

- Reading errors is one of the easiest ways to determine what is wrong with code.
- This feedback is invaluable and is often overlooked by beginners because it can be hard to decipher.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

READING ERRORS

- What to do with an error you are unfamiliar with or don't understand?
- Copy it and paste it into a search engine.
- Generally, stack overflow will pop up and someone will have found the answer.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

THE DEBUGGER

- The debugger slows the program down frame by frame so you can see what is happening
- Let's look at an example of adding the numbers from 1 -100



THE DEBUGGER

The screenshot shows a Microsoft Visual Studio code editor window for a Visual Basic application named "MyFirstVisualBasicApp". The code is written in VB.NET. A yellow arrow points from the text "First Click beside the line you want to stop at. A red dot will appear and stay there till you click on it again" to the line number 9, which is highlighted in yellow. The code is as follows:

```
VB MyFirstVisualBasicApp
1 reference
1    Public Class Form1
2        Public MyNumber As Integer
3        Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
4
5            End Sub
6
7            0 references
8            Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
9                Dim sum As Integer = 0
10               For index = 1 To 100
11                   TextBox1.Text = sum + index
12               Next
13           End Sub
14       End Class
```

First Click beside the line you want to stop at. A red dot will appear and stay there till you click on it again



THE DEBUGGER

```
VB MyFirstVisualBasicApp
    1 reference
1   Public Class Form1
2     Public MyNumber As Integer
3     0 references
4     Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
5       End Sub
6
7       0 references
8       Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
9         Dim sum As Integer = 0
10        For index = 1 To 100
11          sum = sum + index
12        Next
13      End Sub
14    End Class
```

When you hover over a variable it will show what value it currently has. Index is at 0 because this is the start of our loop



THE DEBUGGER

The screenshot shows the Microsoft Visual Studio IDE during a debugging session. The code editor displays a VB.NET script:

```
VB MyFirstVisualBasicApp
    1 reference
1    Public Class Form1
2        Public MyNumber As Integer
3        0 references
4        Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
5            End Sub
6
7            0 references
8            Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
9                Dim sum As Integer = 0
10               For index = 1 To 100
11                   TextBox1.Text = sum + index
12                   Next
13               End Sub
14           End Class
```

The debugger interface includes:

- Call Stack**: Shows the current call stack entry: MyFirstVisualBasicApp.dll!MyFirstVisualBasicApp.Form1.Button1_Click(Object sender, System.EventArgs e) Line 11 [External Code].
- Variables**: The **Autos** window shows variable values:

Name	Type
MyFirstVisualBasicApp.Form1.TextBox1.Text	System.Windows.Forms.TextBox
Microsoft.VisualBasic.CompilerServices.CompilerServices	Object
Me	Object
TextBox1	System.Windows.Forms.TextBox
TextBox1.Text	String
index	Integer
- Breakpoints**: A red arrow points from the text "You can also find variable values down here" to the **Value** column in the Autos window.
- Diagnostics Session**: A separate window on the right shows performance metrics: 2.796 seconds total, 2.792s for the session, and graphs for Process Memory (MB), CPU (% of all processors), and CPU Usage.

You can also find variable values down here



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

THE DEBUGGER

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

When you want to go to the next line of text, click here or F11 (F11 may not work on some laptops)

```
1 reference
1    Public Class Form1
2        Public MyNumber As Integer
3        0 references
4        Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
5            End Sub
6
7        0 references
8        Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
9            Dim sum As Integer = 0
10           For index = 1 To 100
11               TextBox1.Text = sum + index
12               Next < 2ms elapsed
13           End Sub
14       End Class
```

Stack Frame: MyFirstVisualBasicApp.Form1.Button1_Click

Diagnostic Tools

- Diagnostics session: 2 seconds (2.796 s selected)
2.792s
- Events
- Process Memory (MB)
- CPU (% of all processors)

Solution Explorer Git Changes



THE DEBUGGER

```
VB MyFirstVisualBasicApp
1 reference
1   Public Class Form1
2     Public MyNumber As Integer
3     0 references
4     Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
5       End Sub
6
7     0 references
8     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
9       Dim sum As Integer = 0
10      For index = 1 To 100
11        TextBox1.Text = sum + index
12      Next
13    End Sub
14  End Class
```

We have gone through the loops 9 times and sum hasn't changed!?! Why is this? we can see that we are adding index and sum but not changing the value of sum. We should try sum += index then print after we leave the loop

Autos

Name	Value	Type
Me	{MyFirstVisualBasicApp.Form1, Text: Data Types}	MyFirstVisualBasicApp...
TextBox1	{Text = "9"}	System.Windows.Forms...
TextBox1.Text	"9"	String
index	9	Integer
sum	0	Integer

Call Stack

Name
MyFirstVisualBasicApp.dll!MyFirstVisualBasicApp.Form1.Button1_Click(Object sender, System.EventArgs e)
[External Code]



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

THE DEBUGGER

```
0 references
7  Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
8      Dim sum As Integer = 0
9      For index = 1 To 100
10         sum += index
11     Next
12     TextBox1.Text = sum
13 End Sub
```

There we go! Remember to click on the red dot again otherwise your program will keep stoping on the designated line



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

THE DEBUGGER

- An analogy of debugging is Animation. When we slow down animation, we can see each frame like in this short film...
- https://www.youtube.com/watch?v=NLqiY7sJAh0&ab_channel=PedroPaulino
- 4:02 – 4:41



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS

If statements are used to direct the flow of our programs. They allow the computer to make decisions based on specific conditions.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENT ANATOMY

- If statements in VB look like this:

```
If True Then  
|  
|  
End If
```



ELSE IF STATEMENT

```
Dim beverage As String = TextBox1.Text
If beverage = "coffee" Then
    MessageBox.Show("We have coffee!")
ElseIf beverage = "tea" Then
    MessageBox.Show("We have tea!")
ElseIf beverage = "soda" Then
    MessageBox.Show("We have Soda!")
Else
    MessageBox.Show("Sorry we dont have that, You get water")
End If ← End If closes every if block
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS COMPARISON



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

COMPARISON OPERATORS

- = equal
- <> not equal
- > greater than
- < less than
- >= greater than or equal to
- <= less than or equal to



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS

COMPARISON OPERATORS

GREATER, LESS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

GREATER THAN, LESS THAN

```
Dim x As Integer = 22
Dim y.. As Integer = 55
If x = 21 Then
    MessageBox.Show("x is 21")
ElseIf x < 20 Then
    MessageBox.Show("x is less than 20")
ElseIf x > 25 Then
    MessageBox.Show("x is greater than 25")
Else
    MessageBox.Show("x is not 21 and is 20 or greater and less than 25")
End If
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS

COMPARISON OPERATORS

EQUALITY



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

EQUALITY

```
Dim beverage As String = TextBox1.Text
If beverage = "coffee" Then
    MessageBox.Show("We have coffee!")
ElseIf beverage = "tea" Then
    MessageBox.Show("We have tea!")
ElseIf beverage = "soda" Then
    MessageBox.Show("We have Soda!")
Else
    MessageBox.Show("Sorry we dont have that, You get water")
End If
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS COMPARISON OPERATORS INEQUALITY



INEQUALITY

Inequality operator is basically a less than operator and greater than operator put together to indicate that something is not equal.

```
Dim x As Integer = 22
Dim y As Integer = 55
If x <> y Then
    MessageBox.Show("x is not equal to y")
Else
    MessageBox.Show("x is equal to y")
End If
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENTS COMPARISON LOGICAL OPERATORS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOGICAL OPERATORS

- **And** AND ($x = 5$ And $y = 10$)
- **Or** OR ($x = 5$ Or $x = 10$)
- **Not** NOT (Not x)



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOGICAL OPERATORS

```
Dim x As Integer = 22
Dim y As Integer = 55
If x <> y And Not (x <> 22) Or y > 65 Then
    MessageBox.Show("It is True")
Else
    MessageBox.Show("It is False")
End If
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

EXERCISE



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENT EXERCISE 1

Take in one number, check if it is even or odd

Case 1) Input = 55

Output = odd

Case 2) input = 22

output = even



IF STATEMENT EXERCISE 2

Determine if a number evenly divisible by 3 or evenly divisible by 5. When the number is evenly divisible by 3 print "fizz". When the number is evenly divisible by 5 print "buzz". When it is evenly divisible by both print "FizzBuzz". In all other circumstances print "No fizz No buzz"

Case1) Input = 25

 output = Buzz

Case 2) Input = 30

 output = FizzBuzz

Case 3) Input = 12

 output = Fizz

Case 4) Input = 11

 output = No fizz No buzz



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

IF STATEMENT EXERCISE 3

Make A Full Calculator:



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS

- The two most commonly used loops are
 - For loops
 - While loops
- For loops are counter based
- While loops are condition based
- There are variations of these two loops loops that have similar functionality, but are used in different situations



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS FOR LOOP



LOOPS - FOR

- For loops are counter based loops that allow us to iterate a fixed number of times
- For loops in visual basic have a variable defined, then a counter condition

```
For index = 1 To 10
|
|
Next
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS - FOR

You can add steps if you need to skip numbers

```
For index = 1 To 10 Step 2
    RichTextBox1.Text += index.ToString() + ", "
Next
```



LOOPS – FOR BROKEN DOWN

- index = 1 // initialize variable as our counter
- index to 5 // set condition
- Next // increment counter by one (or steps indicated)

```
For index = 1 To 5
    RichTextBox1.Text += index.ToString() + ", "
Next
```



LOOPS – FOR BROKEN DOWN

index = 1

RichTextBox.Text += index.ToString() + "," //prints out 1,

Next // increments by 1

RichTextBox.Text += index.ToString() + "," //prints out 2,

Next // increments by 1

RichTextBox.Text += index.ToString() + "," //prints out 3,

Next // increments by 1

RichTextBox.Text += index.ToString() + "," //prints out 4,

Next // increments by 1

RichTextBox.Text += index.ToString() + "," //prints out 5,



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS FOR LOOP EXERCISES



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS - FOR

Exercise 1:

- Make a program that sums all the numbers from 1 – 100 in the console
- Output = 5050



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS - FOR

Exercise 3 :

- User inputs two numbers
- The program will print and sum all the even numbers between the user's input.
- Input = 1, 9
- Output = 2, 4, 6, 8
- “The sum of all the even numbers is 20”



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS - FOR

Exercise 4 :

- Determine if a number is prime number
- Input = 1399
- Output = It is True Your number is prime
- Input = 22
- Output = It is False Your number is prime
- Input = 2
- Output = It is True Your number is prime
- Input = 1
- Output = It is False Your number is prime



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS WHILE



LOOPS - WHILE

- While loops are a condition-based loop
- While loops can be turned into loop that count, but its best if we specifically use the while loop for conditions that have no numerically defined ending
- Because while loops are based on a condition, they can run forever if the condition isn't met. This may crash whatever platform you are running (Unity will do this, web browsers generally have safety measures in place to prevent this.)
- In windows forms you can just press the stop button if you get into an infinite loop.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

LOOPS - WHILE

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

While True

End While



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS - WHILE

This would be a for loop as a while loop

```
Dim i As Integer = 0
While i < 10
    RichTextBox1.Text += i.ToString() + ", "
    i += 1
End While
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS

WHILE LOOP

EXERCISES



LOOPS - WHILE

Exercise 1

- Research the random number generator for Visual Basic. Set up a while loop that while true it generates a random number and prints the number onto a RichTextBox. The while loop will stop running when the number 62 is generated by the Random number generator.
- Input = 42 as the stop
- Output = All random numbers generated, including 42



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS - WHILE

Exercise 1

```
Randomize()
Dim myRandom As Integer = 0
RichTextBox1.Text = ""
While myRandom <> 42
    myRandom = (Rnd() * 100) + 1
    RichTextBox1.Text += myRandom.ToString() + ", "
End While
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

RANDOM NUMBER GENERATOR



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

RANDOM NUMBER GENERATOR

Random number generators are in almost all programming languages. Random number generators are not truly random but pseudo random. This means that even though a number appears random it has a mathematical formula that can be broken down to accurately predict the number being generated.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

01001

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

RANDOM NUMBER GENERATOR

- The way we get a random number in Visual Basic is by using the Below Code.

```
Dim myRandom As Integer = 0
Randomize()
myRandom = Int((100 * Rnd()) + 1)
RichTextBox1.Text = myRandom.ToString()
```



RANDOM NUMBER GENERATOR

- On its own the rnd() returns a single data type that looks like this:
 - 0.7055475

```
Dim myRandom As Single = 0
myRandom = Rnd()
RichTextBox1.Text = myRandom.ToString()
```

- More about single data type:
 - <https://learn.microsoft.com/en-us/dotnet/visual-basic/language-reference/data-types/single-data-type>



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

RANDOM NUMBER GENERATOR

- 0.7055475 Is the default number that will show up if you do not include the Randomize() function. Try running your program without Randomize(), every time you start the program, it will start with the same number.



RANDOM NUMBER GENERATOR

- If you make your variable type as an Integer, it will automatically make the number 1 or 0. This is because integers will round up or down
 - 0.7055475 becomes 1
 - 0.28956246 becomes 0

```
Dim myRandom As Integer = 0
myRandom = Rnd()
RichTextBox1.Text = myRandom.ToString()
```



RANDOM NUMBER GENERATOR

- Multiply by 100 and you will get a number between 0 and 99 or you can add 1 and it will be between 1 and 100

```
Dim myRandom As Integer = 0
myRandom = Rnd() * 100
RichTextBox1.Text = myRandom.ToString()
```

```
Dim myRandom As Integer = 0
myRandom = (Rnd() * 100) + 1
RichTextBox1.Text = myRandom.ToString()
```



RANDOM NUMBER GENERATOR

- You can add Randomize at the start of your program and it will randomize for the whole run time. I put it in the same function for the sake of this picture.

```
Randomize()
Dim myRandom As Integer = 0
myRandom = Rnd() * 100
RichTextBox1.Text = myRandom.ToString()
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS

EXIT AND CONTINUE



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

EXIT AND CONTINUE

- Exit – terminates the loop
- Continue – terminate the iteration of the loop



BREAK AND CONTINUE IMAGES

- What Will this print out?

```
For index = 1 To 10
    If index = 5 Then
        Continue For
    End If
    If index = 7 Then
        Exit For
    End If
    RichTextBox1.Text += index.ToString() + ", "
Next
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

LOOPS

NESTED LOOPS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

NESTED LOOPS

```
For i = 1 To 10
    For j = 1 To 10
        RichTextBox1.Text += "* "
    Next
    RichTextBox1.Text += vbCrLf
Next
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS



ARRAYS

- An array is a collection of similar data elements stored at contiguous memory locations. It is one of the simplest data structures. Each data element can be accessed directly by using an index.
- Arrays in Visual Basic are by default static arrays, which means they cannot change their size.



ARRAYS BROKEN DOWN

- Let's say you have a collection of heights you want to compare
- You could take in one height at a time and look at it then compare it against the last. This could work maybe once but if you need to look at the same data and do something else this would be painful.
- Another thing you could do is store each height in a variable which we will look at on the next page.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS

ARRAYS BROKEN DOWN



ARRAYS BROKEN DOWN

- If we store each height in a variable, we end up with this:

```
Dim Height1 As Integer = 55
Dim Height2 As Integer = 100
Dim Height3 As Integer = 101
Dim Height4 As Integer = 98
Dim Height5 As Integer = 44
```

- Which isn't great...



ARRAYS BROKEN DOWN

- Array indexes start from 0 and increment by 1

0 1 2 3 4 5

↓ ↓ ↓ ↓ ↓ ↓

```
Dim myArray() As Integer = {55, 100, 101, 98, 44, 120}
```

This is like strings, which are just arrays of characters.



ARRAYS BROKEN DOWN

- You can think of arrays as a collection of variable data but stored in a single variable. We access each individual variable by using the subscript operator “()” and a number called an index.

```
Dim myArray() As Integer = {55, 100, 101, 98, 44, 120}
```

```
MessageBox.Show(myArray(2))
```

Index Subscript Operator



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS BROKEN Down

- On line one we initialize the variable heights using a subscript operator. We can initialize our array with our data or add data to an array.

```
Dim myArray() As Integer = {55, 100, 101, 98, 44, 120}
MessageBox.Show(myArray(2))
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS BROKEN DOWN

- To initialize an array with no data we must define the amount of data we want:

```
Dim myArray(5) As Integer
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS BROKEN DOWN

In class Exercise 1:

Make an array of five numbers, add all the numbers together. (no loops just use the indexes of array)

Input = 5, 1, 22, 4, 3

Output = 35



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS AND LOOPS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS AND LOOPS

- For loops and arrays are best friend
- Why do you think this is?



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS AND LOOPS

- For loops and arrays are best friend because when we loop over an array, we can use the counter in our for loops to track the index of the array. This is why for loops in Visual Basic auto populate “index”



ARRAY LENGTH PROPERTY

In visual basic there is an array.length property. This works ok with loops, but you must minus 1 so it doesn't go out of range.

```
For index = 0 To (myArray.Length - 1)
    MessageBox.Show(myArray(index))
```

[Next](#)



ARRAY LENGTH PROPERTY

- Its better to use array.GetUpperBound(0). This function returns the index of the last element of an array:

```
For index = 0 To myArray.GetUpperBound(0)
    MessageBox.Show(myArray(index))
Next
```

- The 0, is the amount of dimensions our array is.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS AND LOOPS

In Class Exercise 1:

Create an array of 10 unique numbers from 0 - 20, count how many odd numbers, even numbers (that aren't zero) and zeros there are in the array.

Input = 1, 5, 8, 0, 19, 4, 6, 15, 10, 13

Output = “There are 5 odd, 4 even and 1 zero”



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS AND LOOPS

Exercise 2:

Create an array of 100 random numbers, then calculate how many odd numbers, even numbers (even excluding 0) and zeros there are in the array.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS AND LOOPS

Exercise 3:

Use the same array in exercise 2 to check for the largest number, smallest number, add the whole array together, and find the average.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

MULTIDIMENSIONAL ARRAYS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

MULTIDIMENSIONAL ARRAYS

- A multidimensional arrays are arrays in an array.
- A great example of this is an array of strings.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

MULTIDIMENSIONAL ARRAYS

This is an example of an array of strings

```
10 Dim Months() As String = {"January", "February", "March", "April", "May",
    "June", "July", "August", "September", "October", "November",
    "December"}
```



MULTIDIMENSIONAL ARRAYS

- This will iterate over the month array and each letter of each month. Give it a try!

```
9
10 Dim Months() As String = {"January", "February", "March", "April", "May",
11     "June", "June", "July", "August", "September", "October", "November",
12     "December"}
13 For index = 0 To Months.GetUpperBound(0)
14     For innerIndex = 0 To Months(index).Length - 1
15         RichTextBox1.Text += Months(index)(innerIndex) + ", "
16     Next
17     RichTextBox1.Text += vbCrLf
18 Next
```



MULTIDIMENSIONAL ARRAYS

- You could also use a for each loop here, but I wanted to keep it simple, so I used Months(index).length. GetUpperBound() doesn't work on strings

```
9
10 Dim Months() As String = {"January", "February", "March", "April", "May",
11   "June", "June", "July", "August", "September", "October", "November",
12   "December"}
13
14 For index = 0 To Months.GetUpperBound(0)
15   For innerIndex = 0 To Months(index).Length - 1
16     RichTextBox1.Text += Months(index)(innerIndex) + ", "
17   Next
18   RichTextBox1.Text += vbCrLf
19 Next
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAY

Exercise 3:

Create a program that has an array of names when the user enters a name, the program calculates how many times the name is repeated.

Ex. John is input

The output will be:

Name John:2 times

If there is only one name you will have to output:

Name John: 1 time



ARRAY

Exercise 4

Convert a string to camel case:

Input = “Be kind to people”

Output = “beKindToPeople”

Look up:

1. Split()
2. Substring()
3. ToUpper()
4. ToLower()
5. Join()



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBPROGRAMS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBPROGRAMS

- Modular Programming is a software design technique where you break each part of a program into individual separate modules.
- Subprograms (also known as procedures, methods subroutine and functions) are blocks of code set aside in a reusable bundle.
- You have already been using subprograms in this course. These are built in functions.
- There are three basic types of subprograms:
 1. Predefined
 2. Event driven (While these are still predefined, this can be considered a separate type of predefined)
 3. User Defined



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBPROGRAMS PREDEFINED



SUBPROGRAMS – PREDEFINED

- Predefined subprograms are procedures that come with our language or from a library.
- Predefined subprograms are subprograms written by other programmers.
- Some examples are:
 - Join()
 - Substring()
 - ToString()



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBPROGRAMS PREDEFINED EVENTS

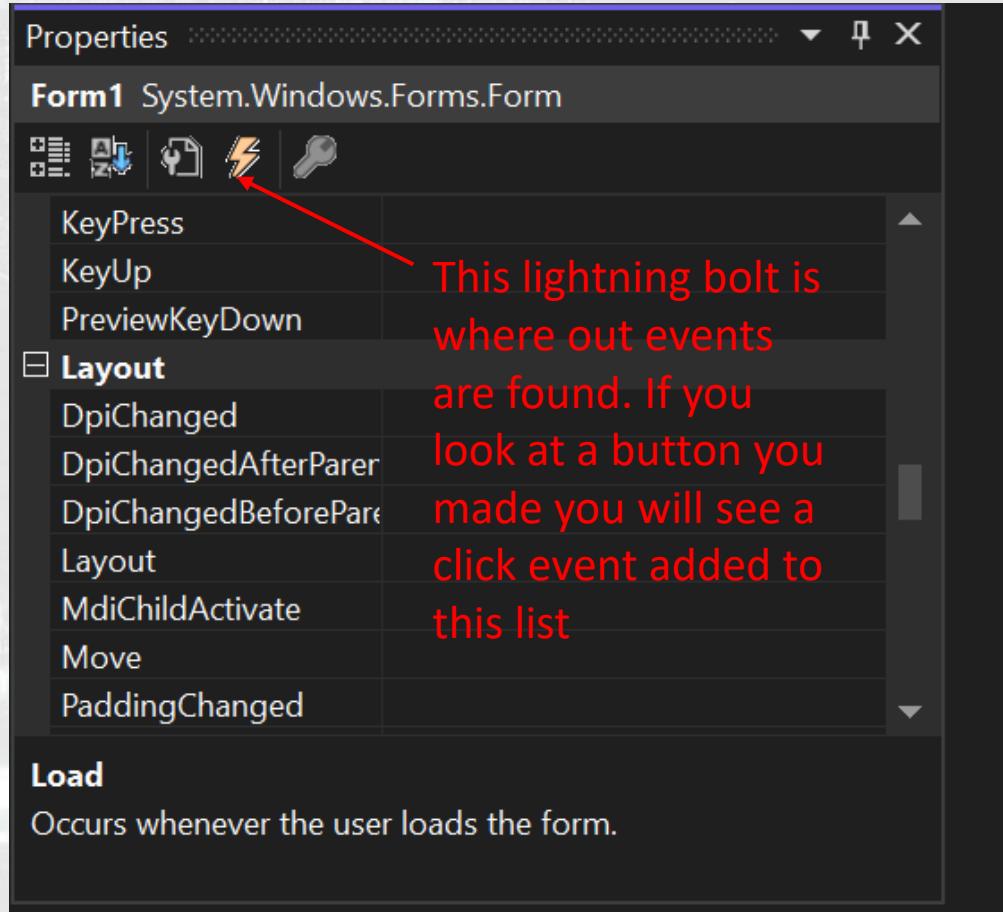


SUBPROGRAMS – PREDEFINED EVENT

- Event subprograms are procedures that allow us to interact with the environment.
- These are common events:
 - Mouse
 - Keyboard
 - Window
 - Input
- There are more you can see in windows forms on the next page:



SUBPROGRAMS – PREDEFINED EVENT





INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBPROGRAMS

USER DEFINED



SUBPROGRAMS – USER DEFINED

- User defined subprograms are the procedures that we will be focusing on for this portion of the course.
- User defined subprograms at the base level are just code blocks we may want to repeat.
- Think about how tedious it would be if every time we wanted to convert a number to a string, we had to write out the code to do this.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS – VISUAL BASIC

- There are two types of subprograms in Visual Basic
 - 1. Functions
 - in VB functions return a value.
 - 2. Subroutine or Sub:
 - Subroutines do not return values.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SUBROUTINE



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SUBROUTINES

- A subroutine is a block of code that we can use over and over in different places.
- A subroutine would be used for something like drawing a Circle, Printing data to the console. There is no data that is passed from the subroutine back into our main program.



VISUAL BASIC SUBROUTINES

2 references

```
Sub Print()
```

```
    RichTextBox1.Text += "Hello, Sub! "
```

```
End Sub
```

This Is a subroutine

called Print() I gave
it that name.

0 references

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    Print()
```

```
End Sub
```

I have “Called” print
on form load

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
    Print()
```

```
End Sub
```

I have “Called” print
on Button Click



VISUAL BASIC SUBROUTINES

- When we “Call” a function this means we are going to be running it on that specific line.
- This function will run twice, Once on the form load, and another time when I click the button.
- It will add to the previous data in the RichTextBox.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

01001

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SUBROUTINES

Subroutines:

Run Sub

Hello, Sub!

This ran on form load

Subroutines:

Run Sub

Hello, Sub! Hello, Sub!

This ran on the
button press and
added to the form
load run



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC PARAMETERS AND ARGUMENTS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SUBROUTINES

- What if we want to change the value of our “Hello, sub!” ?
- What is we want to use the same function to write “Hello, Form Load!” and “Hello, Button Click!”
- How do you think we can do that?



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SUBROUTINES

- We can input different data into our subroutines by defining parameters and inputting arguments.
- Parameters are variables we define so we can pass data into the function
- Arguments are what we define when we call the subroutine.



VISUAL BASIC SUBROUTINES

2 references

```
Sub Print(action As String)
    RichTextBox1.Text += $"Hello, {action}! "
End Sub
```

0 references

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Print("Form Load")
End Sub
```

These are arguments

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Print("Button Click")
End Sub
```



VISUAL BASIC SUBROUTINES

- Exercise 1:
- Make a subroutine called sum that has two integer parameters, add the two integers together and print to a RichTextBox
- Call the subroutine on the form load and a button click and fill in the arguments
- Input form load= 10, 5
- Output = “Form load sum arguments $10 + 5 = 15$ ”
- Input button click = 55, 20
- Output = “Button click arguments $55 + 20 = 75$ ”



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SUBROUTINES

- Exercise 2:
- Make the product (multiplication), quotient (division) and difference (Subtraction) Subroutines just like sum and add them to the form load and button click
- Input form load = 10, 5
- Output = "Form load product arguments $10 * 5 = 50$ "
- Input button click = 55, 20
- Output = "Button click product arguments $55 * 20 = 1100$ "
- Input form load = 10, 5
- Output = "Form load quotient arguments $10 / 5 = 2$ "
- Input button click = 20, 55
- Output = "Button click quotient arguments $55 / 20 = 1100$ "
- Input form load = 10, 5
- Output = "Form load difference arguments $10 - 5 = 5$ "
- Input button click = 20, 55
- Output = "Button click difference arguments $55 - 20 = 2.75$ "



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBROUTINES SCOPE



VISUAL BASIC SCOPE

- What is scope?
- Scope can be thought of as what can be seen and accessed in specific locations.
- When we refer to scope in programming, we are referring to the level of access a variable or function has.
- Some levels of Scope:
 - Block
 - Function (procedural)
 - Module
 - Namespace
- Different names for variables based on scope
 - Global
 - Local
 - protected



VISUAL BASIC SCOPE

- We can think about the level of access ability like Canadian politics.
- The Federal Government has influence over Provincial and municipal
- Provincial has influence over municipal but very little over Federal
- Municipal has influence over itself but has little say in Provincial or Federal affairs.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

VISUAL BASIC SCOPE

3 Levels of Government

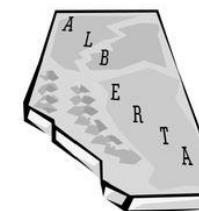
Federal

We elect Members of Parliament



Provincial

We elect Members of the Legislative Assembly



Municipal

We elect a Mayor and councilors



VISUAL BASIC SCOPE

- This is like scope. Global variable can influence everything which means that if we aren't careful, it can cause issues with what goes on locally. Local variables can influence very little outside of its own locality.
- Local variables are generally a better choice.



VISUAL BASIC SCOPE

```
1 reference
Public Class Form1
    Dim myModulaVar As Integer = 10000 ← This is a Modular scope
    0 references
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim myFunctionVar As Integer ← 55 This is procedural scope. It cannot be
        For index = 1 To 10 accessed outside of the procedure it is in
            Dim myBlock As Integer = 22 ←
        Next
        TextBox1.Text = myBlock ← This is a Block scope. It can only be
        End Sub accessed in the current block like the for
        0 references
        Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
            TextBox1.Text = myFunctionVar ← This is procedural scope. It wont work
            End Sub here
            0 references
            Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
                TextBox1.Text = myModulaVar ← This is a Modular scope it works!
                End Sub
            End Class
```



VISUAL BASIC SCOPE

- Now it looks like the modular scope is the winner because it works, but that isn't the case.
- In some circumstances it is good, but its preferable to use a local variable. Here are some reasons
 - We prevent name conflict. You will notice that the for loops all come with index. This is a block variable so if there are no for loops inside our for loops, we can use the index name for all of our loops. In some languages we could be reassigning the variable which would cause even more issues.
 - It saves memory. When we use a variable in a function or block it releases the memory as soon as it is done. If we use a more global scope, it will keep the memory allocated till the program has stopped running .



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBROUTINES STACK



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

SUBROUTINES - STACK

- What is stack?
- Stack is a place in RAM where data gets stored in contiguously. This means that we “stack” one piece of data on another.
- Think of a tennis ball tube or pringles.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

SUBROUTINES - STACK

- The first ball in is the last one out or the last one in is the first one out.
- (Last In First Out -> LIFO)
- This is stack memory. The first piece of data in is the last piece of data out.



475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA



SUBROUTINES - STACK

- When we run a subroutine, it gets stored in stack memory, when the subroutine finishes, the memory is released.
- So, when you run a subprogram, it does take up memory, this can be useful. We can use a technique called recursion to take advantage of the data being stored, Which we will touch on later.
- It can also cause issues, and that is when we run out of stack space. This is called a Stack Overflow. It's like having too many tennis balls in a tube.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS

VISUAL BASIC



VISUAL BASIC - FUNCTIONS

- What is the difference between functions and subroutines?
- As we have seen, a subroutine executes a process, that is all. But what if we wanted to keep and use that value? How can we do that?
- With functions!



VISUAL BASIC - FUNCTIONS

- Functions *return* values that we can use after our function has stopped.

2 references

```
Function Sum(x As Integer, y As Integer)
    Return x + y
End Function
```

Here is the function that returns the sum of x and y

0 references

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Sum(55, 20)
End Sub
```

This will run but nothing substantial will happen

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim myVariableSum As Integer = Sum(55, 20)
    RichTextBox1.Text += myVariableSum.ToString()
End Sub
```

This will pass a value to our variable; We could pass it directly to the RichTextbox if we wanted



VISUAL BASIC - FUNCTIONS

- Functions *return* values that we can use after our function has stopped.

2 references

```
Function Sum(x As Integer, y As Integer)
|   Return x + y
End Function
```

We define Function rather than sub

0 references

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
|   Sum(55, 20)
End Sub
```

0 references

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
|   Dim myVariableSum As Integer = Sum(55, 20)
|   RichTextBox1.Text += myVariableSum.ToString()
End Sub
```



VISUAL BASIC - FUNCTIONS

- Make a one function called arithmetic.
- Create three parameters
 1. Operator (string)
 2. Number1 (Integer)
 3. Number2 (Integer)
- Pass in (+ or - or * or /) as your string arguments and 2 numbers for integers, do the relevant arithmetic based on the operator input and return the value.
- Input = *, 5 , 10 --> Output = 50
- Input = +, 10, 55 --> Output = 65
- Input = -, 42, 2 --> Output = 40
- Input = /, 100, 20 --> Output = 5



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS

EARLY RETURNS



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS– EARLY RETURNS

- When using functions, you can only return once.
This means that once you leave the function you can't go back.



FUNCTIONS– EARLY RETURNS

- This last return for “Water” acts like an else statement. If we put in coffee, it will return coffee and that will be the end.

```
Function Order(beverage As String)
    If beverage = "coffee" Then
        Return "coffee"
    ElseIf beverage = "tea" Then
        Return "tea"
    ElseIf beverage = "soda" Then
        Return "soda"
    End If
    Return "Water"
End Function
```



FUNCTIONS – EARLY RETURNS

```
Function Order(beverage As String)
    ...
    If beverage = "coffee" Then
        Return "coffee"
    End If
    If beverage = "tea" Then
        Return "tea"
    End If
    If beverage = "soda" Then
        Return "soda"
    End If
    Return "Water"
End Function
```



FUNCTIONS— EARLY RETURNS

- This can be useful in events if the user doesn't input the correct data

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
    If TextBox1.Text = "" Then
        MessageBox.Show("Please input data into the text box")
        Return
    End If
    MessageBox.Show(TextBox1.Text)
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS

EXERCISES



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS

Exercise:

- Look back at the 100 random number array we made.
- Move the find largest, find smallest, sum, average, and the (even, odd, zero counter) into separate subprograms.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS

Exercise:

- Make a Mini-ATM.
- Using a function and subroutine, use the function to return the amount withdrawn as a messagebox. Use the subroutine to print the balance in a message box. Print the balance when after you withdraw
- Output is on the next page, but feel free to use your own design



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS – ATM - 1

ATM:



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS – ATM - 2

ATM:

You have \$10000 left

Check Balance

OK

Input Amount to Withdraw:

Withdraw

The screenshot shows a Windows-style application window titled "ATM". Inside the window, there is a message box with the text "You have \$10000 left" and an "OK" button. Below the message box, there is a button labeled "Check Balance". At the bottom of the window, there is a label "Input Amount to Withdraw:" followed by a text input field. To the left of the input field, there is a button labeled "Withdraw".



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS – ATM - 2

The image displays two side-by-side screenshots of an ATM user interface. Both screenshots show a main screen with a 'Check Balance' button and an 'Input Amount to Withdraw:' field containing '100'. Below the field is a 'Withdraw' button.

In the left screenshot, a modal dialog box is open, displaying the message 'You have \$9900 left' and an 'OK' button. A red arrow points from the text 'These happen successively after one button click' to the bottom of this dialog box.

In the right screenshot, another modal dialog box is open, displaying the message 'Here is your \$100' and an 'OK' button. A red arrow points from the same text 'These happen successively after one button click' to the bottom of this dialog box.

ATM:

Check Balance

Input Amount to Withdraw:
100

Withdraw

You have \$9900 left

OK

ATM:

Check Balance

Input Amount to Withdraw:
100

Withdraw

Here is your \$100

OK

These happen successively after one button click



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS

RECURSION



FUNCTIONS— RECURSION

- What is recursion?
- Functions can be used just like a loop:

```
2 references
Function MyFunc(x As Integer)
    If x = 0 Then
        Return x
    End If
    RichTextBox1.Text += x.ToString() + ", "
    Return MyFunc(x - 1)
End Function
0 references
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MessageBox.Show(MyFunc(10))
End Sub
```



FUNCTIONS— RECURSION

- This as used in the example below is inefficient for just printing numbers because of 2 things:

```
2 references
Function MyFunc(x As Integer)
    If x = 0 Then
        Return x
    End If
    RichTextBox1.Text += x.ToString() + ", "
    Return MyFunc(x - 1)
End Function
0 references
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MessageBox.Show(MyFunc(10))
End Sub
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS— RECURSION

1. The function looping into itself is slower than a regular iterative loop.
2. When we call a function, it uses up stack memory. For just printing something is a waste of space.



FUNCTIONS— RECURSION

- What looping in a function is good for, is when we have a problem that has similar sub problems.
- Take factorials
- $!5 = (1) * (2) * (3) * (4) * (5)$
- $!4 = (1) * (2) * (3) * (4)$
- $!3 = (1) * (2) * (3)$
- $!2 = (1) * (2)$
- $!1 = (1)$



FUNCTIONS— RECURSION

- $!5 = (1) * (2) * (3) * (4) * (5)$
- $!4 = (1) * (2) * (3) * (4)$
- $!3 = (1) * (2) * (3)$
- $!2 = (1) * (2)$
- $!1 = (1)$
- If I want to solve factorial 5, I must solve factorial 4 at the same time



FUNCTIONS— RECURSION

- $!5 = (1) * (2) * (3) * (4) * (5)$
- $!4 = (1) * (2) * (3) * (4)$
- $!3 = (1) * (2) * (3)$
- $!2 = (1) * (2)$
- $!1 = (1)$
- If I want to solve factorial 5, I must solve factorial 4 at the same time



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

FUNCTIONS— RECURSION

- Factorials using iteration:
 - Make a loop to solve the factorial problem



FUNCTIONS— RECURSION

- Factorials using iteration:
 - Make a loop to solve the factorial problem

```
Function Factorial(x As Integer)
    Dim fact As Integer = 1
    For index = 1 To x
        fact *= index
    Next
    Return fact
End Function
```



FUNCTIONS— RECURSION BROKEN DOWN

```
1 reference
Function Factorial1()
| Return 1
End Function

1 reference
Function Factorial2()
| Return Factorial1() * 2
End Function

1 reference
Function Factorial3()
| Return Factorial2() * 3
End Function

1 reference
Function Factorial4()
| Return Factorial3() * 4
End Function

1 reference
Function Factorial5()
| Return Factorial4() * 5
End Function

1 reference
Function Factorial6()
| Return Factorial5() * 6
End Function

0 references
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
| MessageBox.Show(Factorial6())
End Sub
```

If we want to get factorial 6,
all we have to do is multiply
6 by factorial 5



FUNCTIONS— RECURSION

- Factorials using Recursion:
 - Try to implement a single recursive function that calculates factorials

```
2 references
Function Factorial(x As Integer)
    If x = 1 Then
        Return x
    End If
    Return Factorial(x - 1) * x
End Function
```

```
0 references
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MessageBox.Show(Factorial(5))
End Sub
0 references
```



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS

OBJECT ARRAYS



ARRAYS OF OBJECTS

- So far, we have looked at arrays of primitive datatypes; Strings, characters and numbers
- How can we make more complex arrays?
- One way is by using arrays of objects
- Because windows forms comes with a library of objects, we can use these objects to build our array



ARRAYS OF OBJECTS

- We are going to be building an airline reservation system which will use a picture box array.
- In our airline reservation system, each seat will be represented by one of two images, when empty the image will have an empty seat, when full the image will have a reserved seat.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OBJECT ARRAYS

PICTUREBOX (PB)



ARRAYS OF OBJECTS - PB

- First, we must figure out how to dynamically add our picture boxes
- A few steps to figure this out are:
 - Figure out how to instantiate a new object in visual basic
 - Add the properties when you instantiate the object
 - Figure out how to add a new object to the form dynamically
 - Hint
`(me.controls.someCoolFunctionThatAddsThingsToTheForm(ThingIWantToAdd))`
 - If you have issues, go to google and try to find how to add a picturebox into a windows form dynamically



ARRAYS OF OBJECTS - PB

- Adding a picture box:

```
Dim MyBox(20) As PictureBox

Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MyBox = New PictureBox() With {
        .ImageLocation = "C:\Users\Mayas\Pictures\VB\pictureBox.png",
        .Size = New Size(1200, 800),
        .Location = New Point(10, 10),
        .Name = "PictureBox2",
        .Visible = True,
        .Enabled = True
    }
    Me.Controls.Add(MyBox)
End Sub
```

This is where we
are instantiating
the
PictureBoxObject



ARRAYS OF OBJECTS - PB

- Adding a picture box:

```
Dim MyBox(20) As PictureBox

Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MyBox = New PictureBox() With {
        .ImageLocation = "C:\Users\Mayas\Pictures\VB\pictureBox.png",
        .Size = New Size(1200, 800),
        .Location = New Point(10, 10),
        .Name = "PictureBox2",
        .Visible = True,
        .Enabled = True
    }
    Me.Controls.Add(MyBox)
End Sub
```

Inside the curly braces is where we add the properties



ARRAYS OF OBJECTS - PB

- Adding a picture box:

```
Dim MyBox(20) As PictureBox

Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MyBox = New PictureBox() With {
        .ImageLocation = "C:\Users\Mayas\Pictures\VB\pictureBox.png",
        .Size = New Size(1200, 800),
        .Location = New Point(10, 10),
        .Name = "PictureBox2",
        .Visible = True,
        .Enabled = True
    }
    Me.Controls.Add(MyBox)
End Sub
```

Me is a reference to the form1, to translate it would be form1.controls.Add(MyBox)
This adds MyBox to form1



ARRAYS OF OBJECTS - PB

```
Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    MyBox = New PictureBox() With {
        .ImageLocation = "C:\Users\Mayas\Pictures\VB\pictureBox.png",
        .Size = New Size(1200, 800),
        .Location = New Point(10, 10),
        .Name = "PictureBox2",
        .Visible = True,
        .Enabled = True
    }
    Me.Controls.Add(MyBox)
End Sub
```

Picture box with a picture of the code instantiating the picture box



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS OF OBJECTS - PB

- Now turn your picture box into an array of pictureboxes
- Make the array display as 4 by 5 on your form.
- Display the empty seat





```
5 Dim BussinessClass(20) As PictureBox
6 Dim myImageLocationPrefix As String = "C:\Users\Mayas\Pictures\VB\
7 Dim EmptySeatImg As String = "airlineSeatEmpty.png"
8 Dim FullSeatImg As String = "airlineSeat.png"
9 Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
10    Dim xLocation As Integer = 10
11    Dim yLocation As Integer = 10
12    For index = 0 To 19
13        BussinessClass(index) = New PictureBox() With {
14            .ImageLocation = myImageLocationPrefix + EmptySeatImg,
15            .Size = New Size(200, 200),
16            .Location = New Point(xLocation, yLocation),
17            .Name = "PictureBox" + index.ToString(),
18            .Visible = True,
19            .Enabled = True,
20            .Cursor = Cursors.Hand
21        }
22        Me.Controls.Add(BussinessClass(index))
23        xLocation += 210
24        If (index + 1) Mod 4 = 0 Then
25            xLocation = 10
26            yLocation += 210
27        End If
28        Next
29    End Sub
```

0 references



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS OF OBJECTS - PB

- Research how to add the AddHander, AddressOf, and the associated function.
- One event handler will work for all pictureboxes.
- Check that it works, click on a picture and have a message box pop up saying “It works!!”



ARRAYS OF OBJECTS - PB

1 reference

```
Private Sub PictureBoxClick(sender As Object, e As EventArgs)
    MessageBox.Show("It worked!")
End Sub
```

```
For index = 0 To 19
    BussinessClass(index) = New PictureBox() With {
        .ImageLocation = myImageLocationPrefix + EmptySeatImg,
        .Size = New Size(200, 200),
        .Location = New Point(xLocation, yLocation),
        .Name = "PictureBox" + index.ToString(),
        .Visible = True,
        .Enabled = True,
        .Cursor = Cursors.Hand
    }
    AddHandler BussinessClass(index).Click, AddressOf PictureBoxClick
    Me.Controls.Add(BussinessClass(index))
    xLocation += 210
    If (index + 1) Mod 4 = 0 Then
        xLocation = 10
        yLocation += 210
    End If
```

Next



ARRAYS OF OBJECTS - PB

- The sender parameter is the pictureBox object you clicked on being passed to the function, You can see this by changing the “It works!!” to sender.Name
- You can use the sender directly, but its easier to put it into a pictureBox variable. You’ll see the difference when you try to access the properties of the picture box using the sender vs the Picture box variable.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS OF OBJECTS - PB

- When the picture box is clicked on, change the seat from empty to full, then when it's clicked on again, change it from full to empty



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

OBJECT ARRAYS NAMES



ARRAYS OF OBJECTS

- You will need an array of rich textboxes to display the names of the people who have booked the flight.
- Make a textbox to take in first name and last name.
- When the user clicks on a picture and the first and last name are not filled in, it will not allow them to make a reservation.
- When they input their names then click to submit, the names show up in the rich textbox



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

ARRAYS OF OBJECTS

PictureBox

Input first name and last name

First Name First

Last Name Last

PictureBox

First name and last name go into the richtextbox

First Name First

Last Name Last

First Name First

Last Name Last

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA

ARRAYS OF OBJECTS

- Make a second array of 10 by 10 (100) for economy.
You can add this on a new.



INSTITUTE OF
TECHNOLOGY
DEVELOPMENT
OF CANADA

01001

475 GRANVILLE STREET, VANCOUVER, BC, V6C 1T1
PHONE: +1(604)558-8727, +1(604)409-8200
TOLL FREE: +1(888) 880-4410
FAX: +1(888) 881-6545
WEB: WWW.ITDCANADA.CA
EMAIL: STUDYING@ITDCANADA.CA



FROM THE ITD CANADA TEAM!!!