

*Lecture 9: Python
Programming 1*

Objectives

- ***The student should be familiar with the Python Programming Environment.***
- ***learn about different data types in Python***
- ***Learn about Python statements***
- ***The student should be able to build a simple Python Program.***

Programming Languages

- Programming language is the code for writing the instructions the computer will follow.
 - Programmers will often refer to their program as *computer code*.
- Process of writing an algorithm in a programming language often called *coding*.

Programming Languages

- High-level language
 $c = a + b$
- This needs to be translated into machine language that the computer can execute.
- *Interpreters* convert programs written in a high-level language into the machine language of some computer.
- Python needs an interpreter for execution
- VS Community 2017 is an Integrated Development Environment (IDE) which includes all necessary tools for developing Python Programs
 - Code Editor
 - Interpreter
 - Libraries
 - Debugger

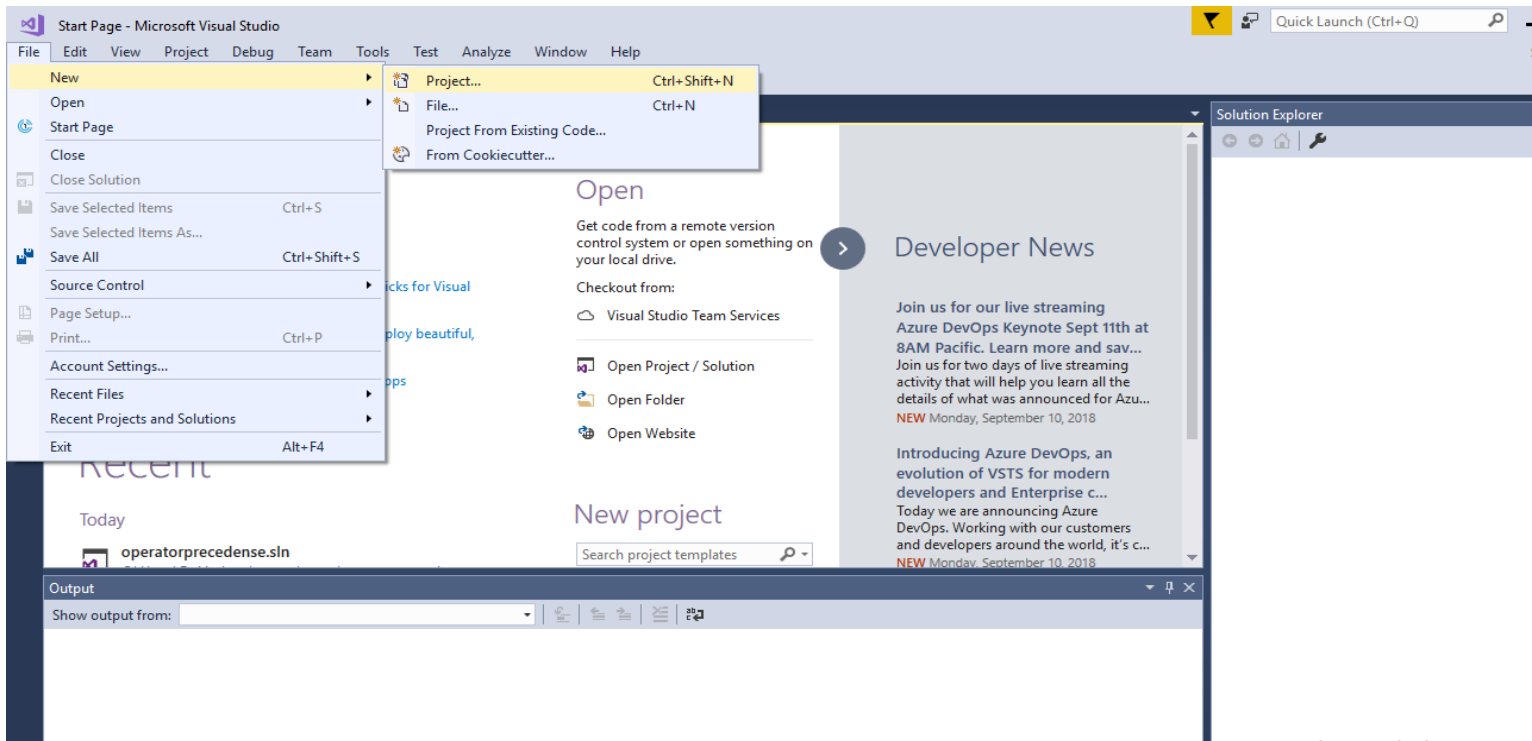
Python installation

- 1. Go to Download page on the web site:
<https://visualstudio.microsoft.com/vs/features/python/>
and click Download Visual Studio Community 2017.**
- 2. When the download is completed, double-click the file
and follow the instructions to install it.**

How to create a new program?

1. After installation open Visual Studio 2017

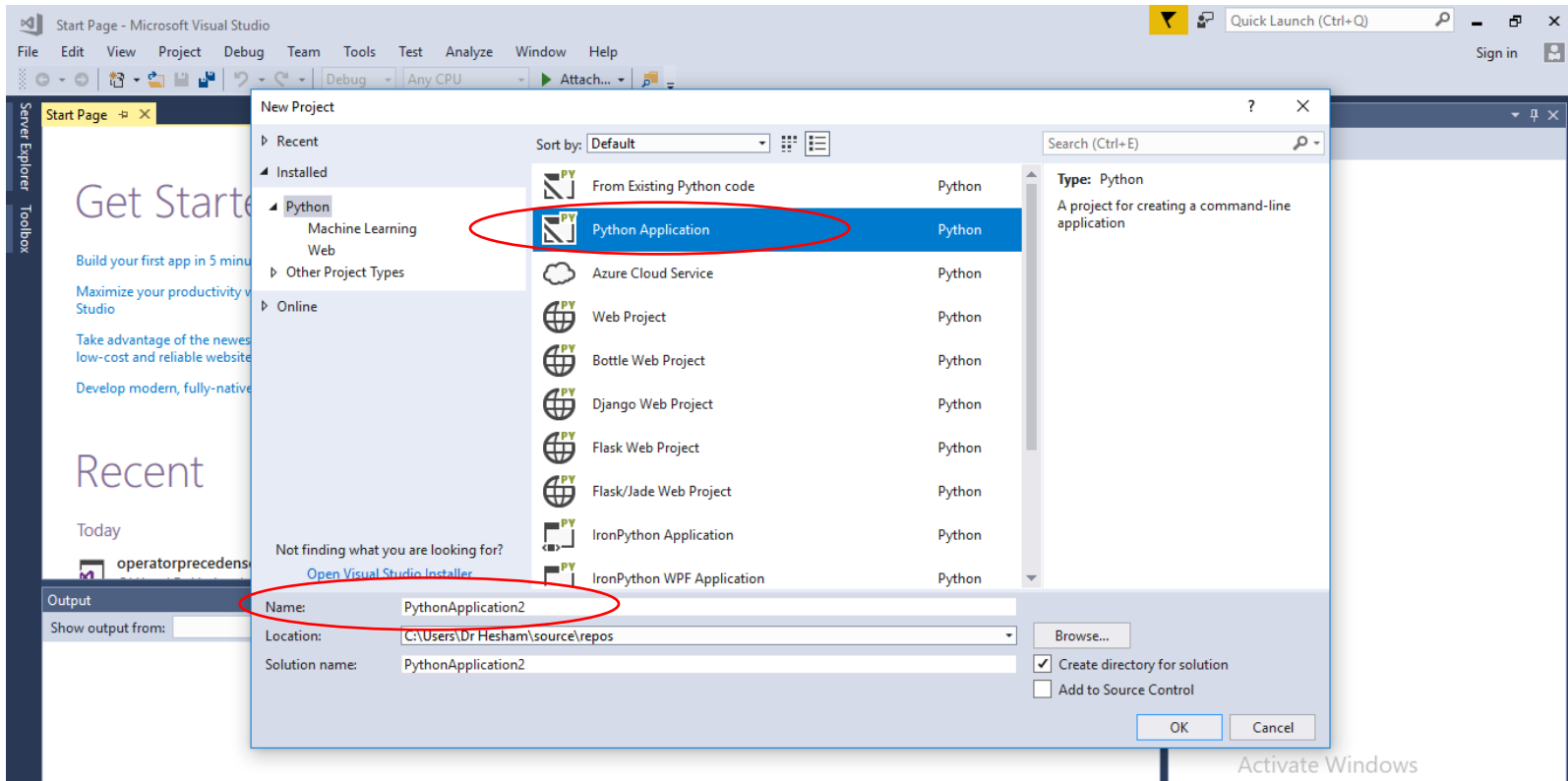
2. Select: File > New > Project



How to create a new program?

3. Choose Python Application

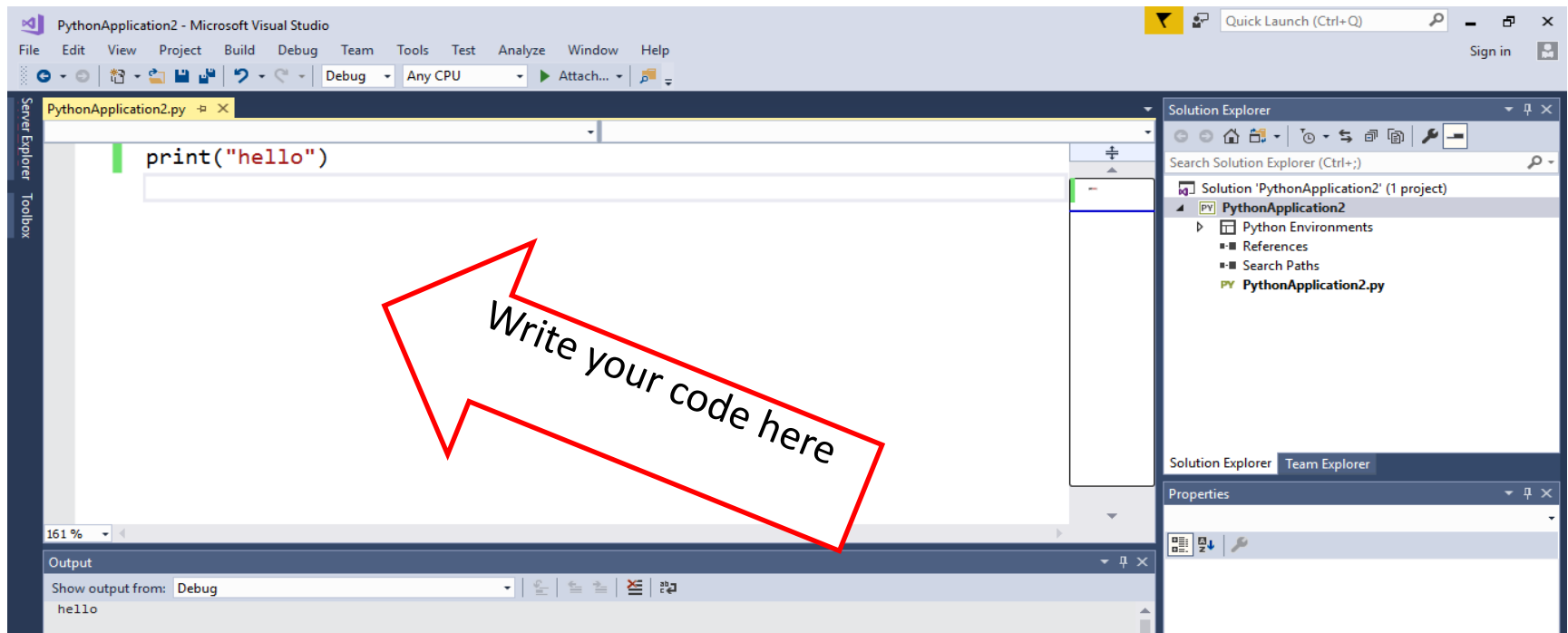
4. Provide a name for your project



How to create a new program?

5. Click ok and the VS will create all necessary files for your work

6. Start writing your code



Python Programming

```
num1 = 3
```

```
num2 = 5
```

```
sum = num1+num2
```

```
print (sum)
```

Elements of Programs

Code

```
print(3+4)  
print(3, 4, 3+4)  
print()  
print(3, 4)  
print(3 + 4)  
print("The answer is", 3+4)
```

Output screen

```
7  
3 4 7  
  
3 4  
7  
The answer is 7
```

Elements of Programs

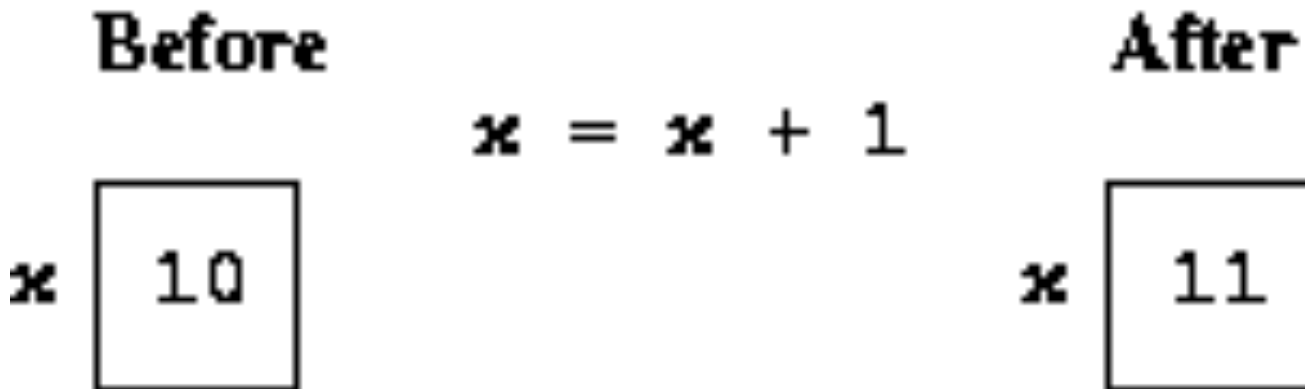
- **Variables**
 - Each have a unique name, termed as identifier (e.g. Temp, fahrenheit)
 - Every identifier must begin with a letter or underscore (“_”), followed by any sequence of letters, digits, or underscores.
 - Identifiers are case sensitive.
 - We cannot use a keyword as variable name

Keywords in Python programming language

False	class	finally	is	return
None	continue	for	lambda	try
True	def	from	nonlocal	while
and	del	global	not	with
as	elif	if	or	yield
assert	else	import	pass	
break	except	in	raise	

Assignment Statements

- *Variables are like a box we can put values in.*
- *When a variable changes, the old value is erased and a new one is written in.*



Assignment Statements

- *Variables can be reassigned as many times as you want!*

```
myVar = 0  
print( myVar)  
myVar = 7  
print( myVar)  
myVar = myVar + 1  
print( myVar)
```

Output screen



```
0  
7  
8
```

Assignment Statements

```
a, b, c = 5, 3.2, "Hello"  
print( a)  
print( b)  
print( c)
```

Output screen

```
5  
3.2  
Hello
```

Numeric Data Types

- *Python has a special function to tell us the data type of any value.*

```
print (type (3))  
print (type (3.1))  
print ( type (3.0))  
myInt = 32  
print ( type (myInt))
```

Output screen

```
<class 'int'>  
<class 'float'>  
<class 'float'>  
<class 'int'>
```


Numeric Data Types

- ***Operations on int produce int, operations on float produce float (except for /).***

```
print(3.0+4.0)
```

```
print( 3+4)
```

```
print( 3.0*4.0)
```

```
print( 3*4)
```

```
print( 10.0/3.0)
```

```
print(10/3)
```

```
print( 10 // 3)
```

```
print( 10.0 // 3.0)
```

Output screen

```
7.0
```

```
7
```

```
12.0
```

```
12
```

```
3.333333333333335
```

```
3
```

```
3.0
```

Inside a Python Program

```
x = int(input("Enter a number "))  
print(x)
```

- *x is an example of a variable*

Output screen



Enter a number 1

1

Inside a Python Program

```
y = float(input("Enter a number "))  
print(y)
```

- *y is an example of a variable*

Output screen

Enter a number 3.6

3.6

Python Operator Precedence

Highest Precedence	()	Parentheses
	**	Exponentiation
	* / %	Multiplication, Division & Mod
Lowest Precedence	+ -	Addition & Subtraction

What is the result of evaluating
Print(3 + 6 * (5 + 49) / 3 ** 3)

(3 + 6 * 54 / 3 ** 3)

(3 + 6 * 54 / 27)

(3 + 324 / 27)

(3 + 12)

15

print(27 / 3 % 2 * 42)**

27/3%2*16

9%2*16

1*16

16

**Operators of the same
precedence are evaluated from
left to right**

Assignment Statements

- ***Simple Assignment***

- ***<variable> = <expr>***

- variable is an identifier, expr is an expression***

- ***The expression on the RHS is evaluated to produce a value which is then associated with the variable named on the LHS.***

Inside a Python Program

```
x = 3.9 * x * (1 - x)  
print(x)
```

- *This is called an assignment statement*
- *The part on the right-hand side (RHS) of the “=” is a mathematical expression.*
- ** is used to indicate multiplication*
- *Once the value on the RHS is computed, it is stored back into (assigned) into x*

Example Program: **Temperature Converter**

- ***Analysis – the temperature is given in Celsius, user wants it expressed in degrees Fahrenheit.***
- ***Specification***
 - ***Input – temperature in Celsius***
 - ***Output – temperature in Fahrenheit***
 - ***$9/5(\text{input}) + 32$***

Example Program: Temperature Converter

- **Design**

- **Input, Process, Output**
- **Prompt the user for input (Celsius temperature)**
- **Process it to convert it to Fahrenheit using $F = 9/5(C) + 32$**
- **Output the result by displaying it on the screen**

Example Program: **Temperature Converter**

- ***Pseudocode:***
 - ***Input the temperature in degrees Celsius (call it celsius)***
 - ***Calculate fahrenheit as $(9/5)*celsius+32$***
 - ***Output fahrenheit***
- ***Now we need to convert this to Python!***

Example Program: **Temperature Converter**

```
#convert.py
# A program to convert Celsius temps to Fahrenheit
celsius = float(input("What is the Celsius temperature? "))
fahrenheit = (9/5) * celsius + 32
print("The temperature is ",fahrenheit,"degrees Fahrenheit.")
```

Once we write a program, we should test it!

What is the Celsius temperature? 0

The temperature is 32.0 degrees Fahrenheit.

Using the Math Library

- *In python a number of mathematical operations can be performed with ease by importing a module named “math” which defines various functions which makes our tasks easier*
- *To use a library, we need to make sure this line is in our program: `import math`*
- *Importing a library makes whatever functions are defined within it available to the program.*

Math Library

math.factorial(x) ***Return x factorial***

math.exp(x) ***Return e**x.***

math.pow(x, y) ***Return x raised to the power y***

math.sqrt(x) ***Return the square root of x.***

math.cos(x) ***Return the cosine of x radians***

Using the Math Library

- ***Let's write a program to compute the roots of a quadratic equation!***

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

- ***The only part of this we don't know how to do is find a square root... but it's in the math library!***

Using the Math Library

- To access the sqrt library routine, we need to access it as *math.sqrt(x)*.
- Using this dot notation tells Python to use the sqrt function found in the math library module.
- To calculate the root, you can do
`discRoot = math.sqrt(b*b - 4*a*c)`

Using the Math Library

```
# A program that computes the real roots of a quadratic equation.
# Illustrates use of the math library.
# Note: This program crashes if the equation has no real roots.
import math # Makes the math library available.
print("This program finds the real solutions to a quadratic")
print()
a = float(input("Please enter the coefficients a: "))
b = float(input("Please enter the coefficients b: "))
c = float(input("Please enter the coefficients c: "))
discRoot = math.sqrt(b * b - 4 * a * c)
root1 = (-b + discRoot) / (2 * a)
root2 = (-b - discRoot) / (2 * a)
print()
print("The solutions are:", root1, root2 )
```

```
This program finds the real solutions to a quadratic
Please enter the coefficients a: 1
Please enter the coefficients b: 5
Please enter the coefficients c: 6
The solutions are: -2.0 -3.0
```