

# Python!!!

Code Girls 2020-21



# What is Python?

- A popular high level programming language
  - Known for its simple syntax that is similar to the English language
- web development (server-side),
- software development
- mathematics
- system scripting



# Downloading Python

- Go to <https://www.python.org/downloads/>
  - Click the big download button!
  - Follow the instructions

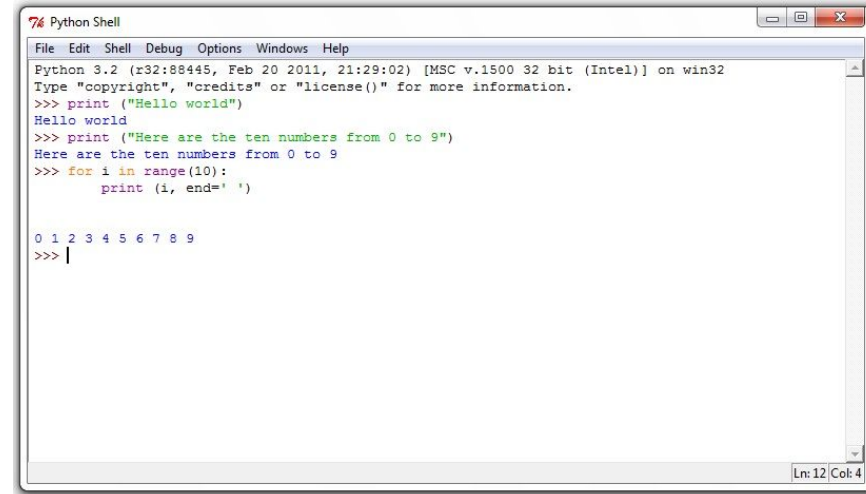
OR

- Go to [repl.it](https://repl.it) and use Python



# Using IDLE

- Opening IDLE will open the Python Shell
  - You can use this to try out some short blocks of code
- In the Shell, go to **File** → **New File**, and it will open up a module where you can code and save your programs
- Go to **Run** → **Run Module** to test your code, and it will run in the Shell

A screenshot of the Python Shell window. The window has a title bar that says "Python Shell" and a menu bar with "File", "Edit", "Shell", "Debug", "Options", "Windows", and "Help". The main text area contains the following text:

```
Python 3.2 (r32:88445, Feb 20 2011, 21:29:02) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> print ("Hello world")
Hello world
>>> print ("Here are the ten numbers from 0 to 9")
Here are the ten numbers from 0 to 9
>>> for i in range(10):
    print (i, end=' ')
0 1 2 3 4 5 6 7 8 9
>>> |
```

The status bar at the bottom right shows "Ln: 12 Col: 4".



# Basic Syntax: Lines and Tabs

Unlike other languages like Java and HTML/CSS/JS, there are no symbols like braces {} or semicolons ; that separate or group lines of code.

- Lines are used to separate each statement
  - Cannot put two statements that perform different tasks on the same line
- Tabs are used to group lines of code
  - Lines that are indented the same amount are grouped in the same block

```
print(1)
print(1 + 1)
print(3 * 1 + 2)
print(3 * (1 + 2))

if 2 > 1:
    print("One is the loneliest number")
else:
    print('Two is the lonliest number?')
```



# Basic Syntax: Reserved Words

Python has some reserved keywords that can only be used in certain contexts and not as variables

and	exec	not
assert	finally	or
break	for	pass
class	from	print
continue	global	raise
def	if	return
del	import	try
elif	in	while
else	is	with
except	lambda	yield



# Comments

Used to document code! Just like HTML/CSS/JS

- # for single line comments
- Begin and end multi-line comments with “””

The screenshot shows a Python IDE window titled "pyprogram.py - D:\python\pyprogram.py (3.8.1)". The menu bar includes "File", "Edit", "Format", "Run", "Options", "Window", and "Help". The code editor contains the following Python code:

```
# Single line comment

print('Hello Stechies')

'''
Python program to explain
multiple line comment
print('Hello')
'''

"""
Python program to explain
multiple line comment
print('Hello')
"""

print('Hello Stechies')
```

Two callout boxes are present:

- A box labeled "Single-line Comment" points to the line `# Single line comment`.
- A box labeled "Multi-line Comments" points to the triple-quoted string `'''...'''`.

# Variables

Similar to JavaScript, Python has variables as well.

- We don't need to add **var** in front to create a variable anymore
- Simply use the assignment operator (=) to assign values

```
1  # Declare a variable and initialize it
2  f = 0
3  print(f)
4  # re-declaring the variable works
5  f = 'guru99'
6  print(f)
```

you can re-declare the variables, even-after if it is declared once. it works fine

Run Python5.1

```
"C:\Users\DK...
5/PythonCode5/
0
guru99
```





# Math Operators

They work just like regular math! With a few extras.

Name	Symbol
Addition	+
Subtraction	-
Multiplication	*
Division	/
Modulus	%
Exponentiation	**
Floor Division	//

```
Python 3.6.1 Shell
File Edit Shell Debug Options Window Help
>>> x = 5
>>> y = 2
>>> x + y #Addition Operator
7
>>> x - y #Subtraction Operator
3
>>> x * y #Multiplication Operator
10
>>> x / y #Division Operator
2.5
>>> x % y #Modulus Operator
1
>>> x // y #Floor Division Operator
2
>>> x ** y #Exponent Operator: x^y
25
Ln: 19 Col: 4
```



# Print Statements

What the user will see the program output

**print("Hello World!")**

**print(variableName)**

```
*Python Shell*
File Edit Shell Debug Options Windows Help
Python 3.2.3 (default, Apr 11 2012, 07:15:24) [MS
Type "copyright", "credits" or "license()" for mo
>>> print ( "Hello World!")
Hello World!
>>> print ( "Learning Python is great!")
Learning Python is great!
>>> print ( "10+10")
10+10
>>> print ( 10+10 )
20
>>> print ( 10+10-5*20 )
-80
>>> print ( "Hello World!")
```



# Input statements

A way that the user can interact with the program

- **input("Enter a number:")** will print **Enter a number:** and accept user input
- To actually use the input, we have to assign a variable to the input
- **num = input("Enter a number: ")**
- Inputs always default to a string, so the variable **num** won't be actually be a number yet (but we'll deal with that next time!)

```
PythonExample.py x
1 # Python Program - Get String Input from User
2 str = input("Enter any string: ")
3 print(str)
```



# Acknowledgments

Thanks to:

- Ms. Lola Piper for sponsoring our club and guidance