LECTURE 1.3
INTRODUCTION TO PYTHON

Lee Wilkins

Python language and coding environment

Python syntax

Data values

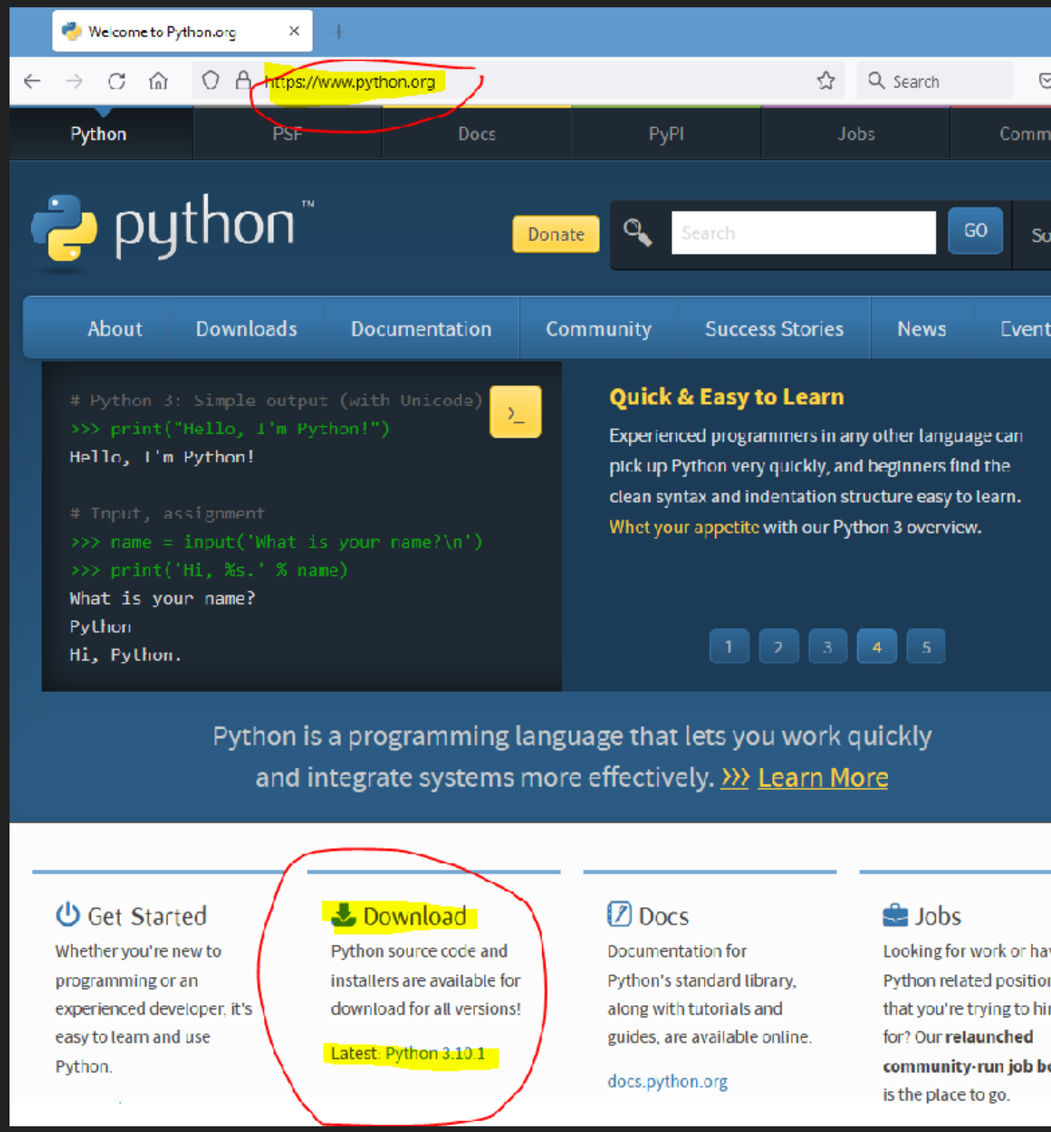
Input

Output

CONTENT

Download/install latest
python interpreter and VSCode
Go to
https://www.python.org/downloads/
For mac OS
https://www.python.org/downloads/macos/

Watch a video on how to set it up with VSCode
https://www.youtube.com/watch?v=cUAK4x_7thA



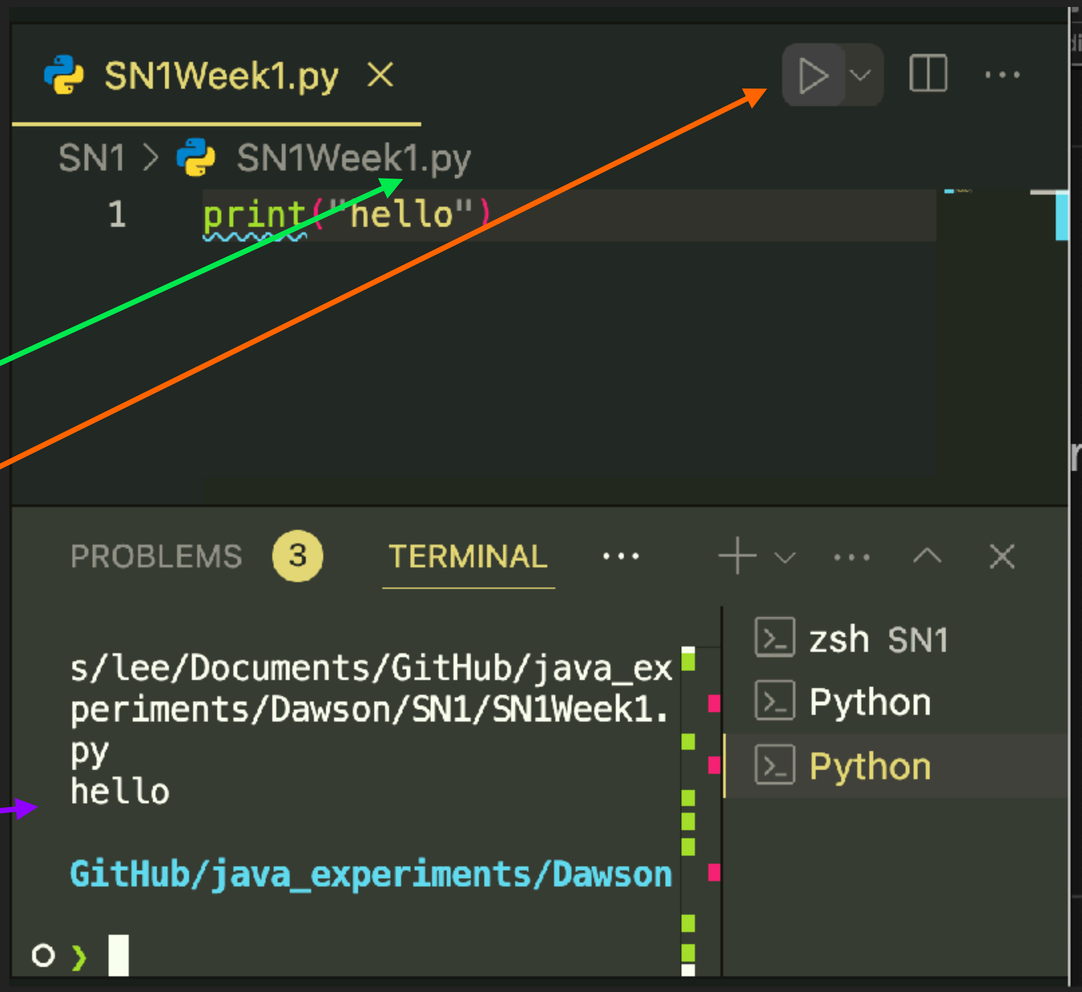
PYTHON DOWNLOAD/INSTALLATION

You can run a Python file basically two ways:

- By finding it in your command line or terminal, and running it like this:

Python mFile.py

- By saving it as a .py file in VSCode and clicking run, it will run in the VSCode terminal



PYTHON EXECUTION

```
~
❯ python
Python 3.11.9 (main, Apr  2 2024, 08:25:04) [Clang 15.0.0 (clang-1500.3.9.4)] on
 darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> print("hello")
hello
>>>
```

You can also run Python in the terminal by typing python

Then writing your program (good for small or quick programs!)

PYTHON EXECUTION

- An IDE (integrated development environment)
- Good to highlight syntax and errors
- has tools to help you debug
- There are other IDEs! Use whatever you like.

VSCODE

INDENTATION, COMMENTS, INPUT, VALUES

PYTHON SYNTAX

Comments are helpful ways to leave human-readable messages in your code.

- Comments can be used to document you rower
- Describe your code
- Add instructions, clarifications, or credits!
- Temporarily remove code for testing

```
# This is a comment, it comments the
entire line until you hit return
```

PYTHON COMMENTS

```python
# This is a comment
print("this is some code")
# print("this is some code that is commented out, so it wont run")

'''
I can comment big blocks of code like this
with man lines
print("and code!")
'''
```

You can comment many things at once by highlighting them and pressing command /, this also works for uncommenting!

PYTHON COMMENTS

Python has built in functionality. You can find them in the docs
https://docs.python.org/3/library/functions.html
If you need help, this is a great place to start!

PYTHON FUNCTIONS

```
print("hello world”)
```

Print is a way to display information in the terminal. You can print a string (some words), an integer (a whole number) or a float (a decimal) (or other things we will talk about later)

Print is the name of the function. It is case sensitive and can’t be written as PRINT, Print, prInT

Error:

NameError: name 'prAint' is not defined. Did you mean: 'print'?

( ) Round brackets represent that it is a function. You can pass arguments to it by putting them between the braces.

”” There are quotes because it is a string (a series of characters)

Hello world is the content of the string.

Pay attention to syntax, it matters!

PRINT

Look for errors in the terminal

OR sometimes they will be underlined!

Hover over them to get more info

ERRORS

```
1  if 1 < 2:
2      PRINT("Yes")
3  else:
4      print("No")
5
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL

```
.py", line 3, in <module>
    PRINT("Yes")
    ^^^^^
NameError: name 'PRINT' is not defined
```

```
"PRINT" is not defined Pylance(reportUndefinedVariable)
(function) PRINT: Any
View Problem (⌥F8)    Quick Fix... (⌘.)
PRINT("Yes")
else:
print("No")
```

```
3  else:
4  print("No")
5
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS ···

```
    print("No")
    ^
IndentationError: expected an indented block after 'else' statement on l
ine 3
```

Python relies on indentation to understand your code

Right now, we aren't using indentation, but we will soon! However, even an extra space before your code can cause problems

Make sure to keep an eye on the number of spaces before your code

IndentationError: unexpected indent

Eventually, your programs will look more like this

```
if 1 < 2:
    print("Yes")
else:
    print("No")
```

```
1     print("hello world")
```

PROBLEMS 5 OUTPUT DEBUG CONSOLE TE

```
wson/SN1/SN1Week1.py
  File "/Users/lee/Documents/GitHub/java_expe
.py", line 1
    print("hello world")
IndentationError: unexpected indent
```

PYTHON INDENTATION

A statement is an instruction that the Python interpreter can execute.

assignment statement	pi = 3.14
Conditional statement	if age >= 18:
Repetitive statements	for x in range:
import statements.	import math
input/output	print, input

TYPES OF STATEMENTS

Variables are a way of keeping track of information, by putting a name to a specific location in memory. You can think of a variable like a labeled bucket that holds information.

Lets create a variable:

```
age  = 20
```

Break it down:

age is the name of the variable (it can be anything, within reason)

= is an assigner, it is putting content into the bucket of the variable. This does NOT mean equal (that is ==)

20 is the content, it can change!

VARIABLES

```python
my_va@r= "hello"
print(my_va@r)
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS ···

```
    my_va@r= "hello"
    ^^^^^^^
SyntaxError: cannot assign to expression here. Maybe you meant '==' in
ead of '='?
```

```python
True = "hello"
print(True)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
.py", line 1
    True = "hello"
    ^^^^
SyntaxError: cannot assign to True
```

- Name your variables clearly, using a single word, snake_case or camelCase
- Do not start with an upper case letter (ex Age)
- Do not include special characters (No !@#$%^&*(),.<>?:’
- _ is allowed
- Do not use “reserved” words, meaning words that Python already uses. For example, you can’t use True as a variable because its used elsewhere in Python

PYTHON VARIABLE

```
1   my_vars = "hello"
2   print(my_var)
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS ···

```
.py", line 2, in <module>
    print(my_var)
          ^^^^^^
NameError: name 'my_var' is not defined. Did you mean: 'my_vars'?
```

Always double check your spelling! You'll see an error and underline if the variable is not declared anywhere else

PYTHON VARIABLE

```
my_var = input("Prompt Message: ")
```

Display the message '*prompt-message*' on the screen and awaits the user to enter something

Once enter key is hit, the input data is assigned to **variable_name**

prompt-message is optional

We can then print that message

```
print(my_var)
```

myVar doesn't need quotes, because it is already a string.

INPUT

```
name = input("What is your name? ")
age = input("how old are you? ")
animal = input("what is your fav animal?")

print("Welcome to the survey! ")
print("your name is “,name, " and you like “, animal)
```

In this example we are asking for 3 pieces of information, and storing them in a variable

We’re then printing that information out again! We can add the input to a string to get more dynamic content. We use

PYTHON VARIABLE — CODING EXAMPLE

```
name = input("What is your name? ")
age = input("how old are you? ")
animal = input("what is your fav animal?")

age = 6

print("Welcome to the survey! \")
print("your name is “, name, " and you like “, animal)
print("You don't look a day over ",age)
```

Here, we changed the variable! You can re-assign it to anything you want.

CHANGE THE VARIABLE

```
print(" \n Welcome to the survey! “)
```

\ is an escape character, it means it is used to help add characters that are difficult to add to strings. Such as line breaks, quotations etc. \n is for a new line

https://www.geeksforgeeks.org/python-escape-characters/ find the rest here

```
print("your name is "+name+" and you like "+animal)
```

You can use + to connect two strings (concatenate strings). However, if the input was only a number, python will not interpret it as a string. For concatenating strings, , is useful if some will be numbered inputs to be printed

TypeError: can only concatenate str (not "int") to str

PRINTING STRINGS

Create a program for a the person sitting next to you. It can be:

- A joke, or a story
- An intake form
- A survey for a project
- Display and modify the information back to the user

Try to change the variables, or the bonus question if you want.

LAB

- You can use GitHub to upload your code if you know how! https://docs.github.com/en/get-started/start-your-journey/hello-world

- Email it to yourself or save it somewhere. You can copy the file either as text, or as a file. Right click “reveal in finder” if you can’t locate it

TAKE YOUR CODE HOME