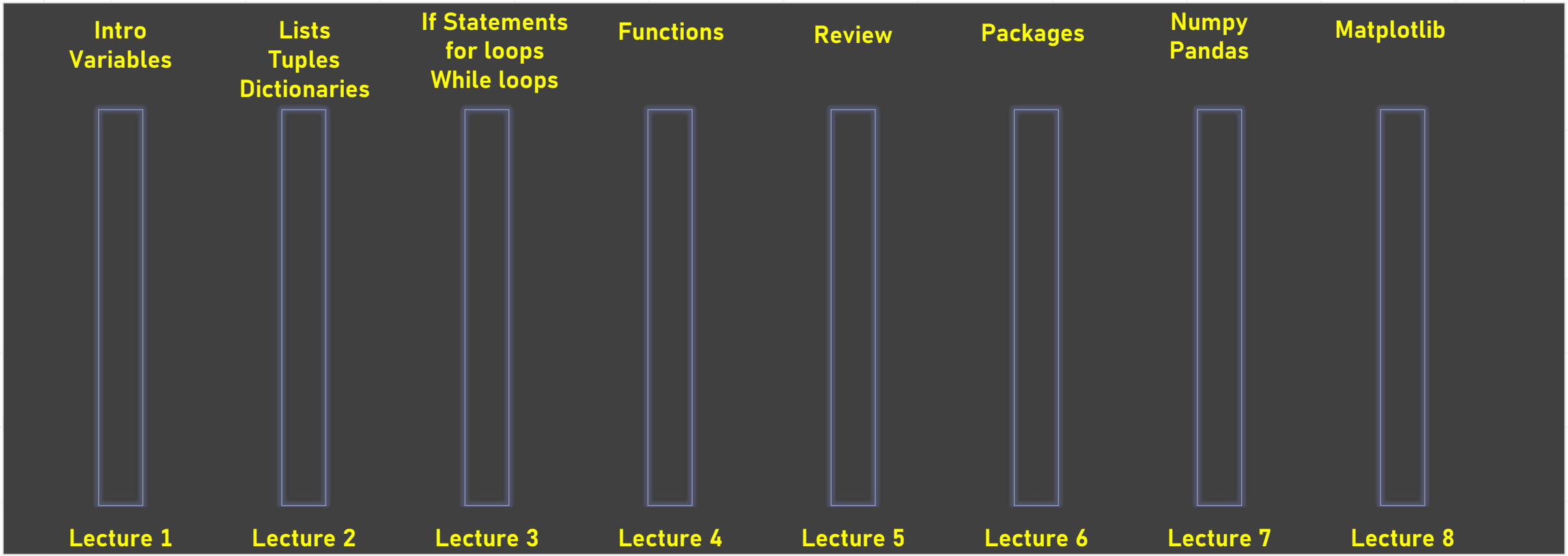
Programming in Python
Lecture 1- Intro, Variables

Lecturer

- **Dor Shani**
- **Business Analyst**
- **Contact** :
  - **mdors1990@gamil.com**

Syllabus


Intro
Variables
Lists
Tuples
Dictionaries
If Statements
for loops
While loops
Functions
Review
Packages
Numpy
Pandas
Matplotlib
Lecture 1
Lecture 2
Lecture 3
Lecture 4
Lecture 5
Lecture 6
Lecture 7
Lecture 8

Plan for today

- **Brief background**
- **Installing Python**
- **Python basics**:
  - Variables
    - numbers
    - Strings
- **Computational Operators**
- **Logical Operators**

Welcome to the python Programming model!

- We will learn to program in **Python**.
- **Goal**: enable you to use **programming as a tool** to solve day-to-day problems.
- **Hard work is required!!!**

Preface

- We assume no prior knowledge.
- However, we advance fast.
- The only way to keep on track is to **practice and ask questions**!

Why Python?

Python (since 1991):

- Quick development
- Can handle large quantity of data
- Easy to learn(interpreter)
- Huge community(w3schools, datacamp, github, stackoverflow, youtube, facebook etc.)
- Short development-execution rounds
- Fast enough for most applications
- Python is widely industrial used (Google, Yahoo! YouTube, BitTorrent, IDF, NASA)
- Cross-platform
- **Packages/ libraries**

Guido van Rossum

Installing and Running Python

- Regular python installation available here:
  - http://python.org/download/
- Install Pycharm distribution for Python from here:
  - https://www.jetbrains.com/pycharm/download/#section=windows
- Available for window, Mac OS and Linux
- Pycharm comes with many useful Python packages
- Tutorials:
  - https://www.guru99.com/how-to-install-python.html
  - https://www.youtube.com/watch?v=Kn1HF3oD19c
  - https://www.youtube.com/watch?v=lM5Y7BnP56k
  - Python Online Compiler- https://www.programiz.com/python-programming/online-compiler/

Questions?

My First Python Program: Hello World!

Hands On

Hello World!

```
1  #First code
2
3  print("Hello World")
4
```

Python Console

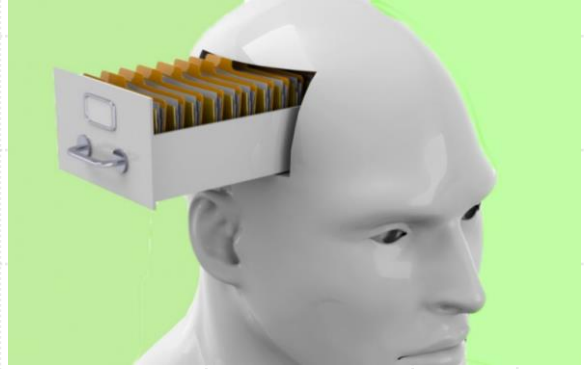
```
C:\Users\USER\PycharmProjects\pythonProject\venv\Scripts\python.exe "C:\Program Files\JetBrains\PyCh

import sys; print('Python %s on %s' % (sys.version, sys.platform))
sys.path.extend(['C:\\Users\\USER\\PycharmProjects\\pythonProject', 'C:/Users/USER/PycharmProjects/p

Python Console
Hello World
```

Questions?

Memory



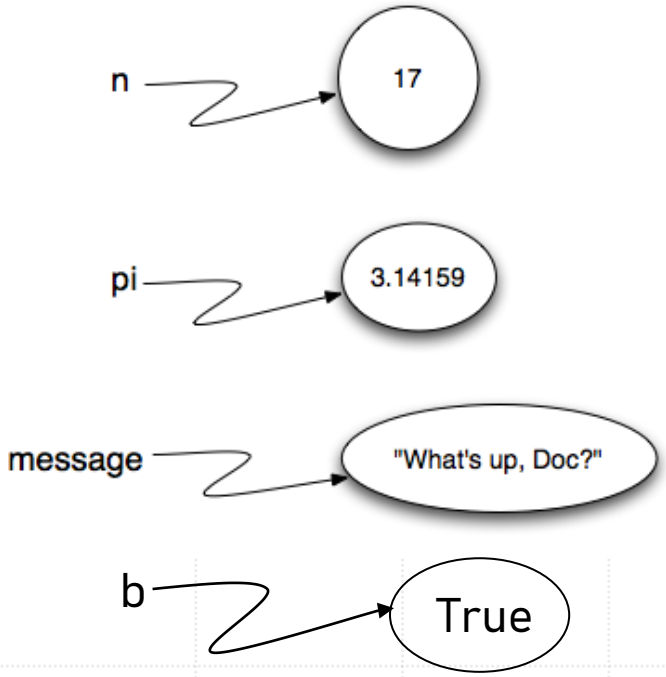
- The computer memory is composed of a long list of bits (0 and 1)
- Bits are grouped into bytes (8 bits) and words (4 bytes, 8 on 64-bit systems)
- Every byte is numbered sequentially
- This number is called an address

What are Variables ?

- A location in the computer’s memory.
- A **variable**:
  - has a name
  - holds a value
  - has type – according to its value
  - This is how data is handled

Why Do We Need Different Types?

- Saving memory
- Execution speed
- Variables types:
  - Int(integer)
  - Float(numbers with decimal point)
  - Strings(text sequences)
  - Booleans(True or False)



Variables and Assignments

```
>>> n = 34
>>> m = (34+ 4) * 5
```

The left-hand side is a variable.

The right-hand side is an expression.

The interpreter:

1. evaluates the expression
2. assigns its value to the variable.

The variable's name is a sequence of letters and digits, starting with a letter.

Why do We Need Variables?

- Computer programs manipulate data
- Data is given as input or calculated throughout the program
- To access them later, variables must be remembered
- Thus, variables are stored in the memory
- Variable name → memory address
- You can not concatenate different types of variable!!

Arithmetic Operators

Operator	Use	Description
+	x + y	Adds x to y
-	x - y	Subtracts x from y
*	x * y	Multiplies x by y
**	x ** y	X to the power y
/	x / y	Divides x by y
%	x % y	Computes the remainder of dividing x by y

Questions?

Hands On

Strings

- Strings are text sequences.
- An ordered list of characters

Strings Slicing

```
str="51689"
print( str[1])
>>>'1'
print( str[0:3])
>>>'516'
print( str[1:])
>>> '1689"
print( str[-3:-1])
>>> '16'
print( str[:-3])
>>> '51"
print([::-1])
>>> '98615'
```

5	1	6	8	9
0	1	2	3	4
-5	-4	-3	-2	-1

Strings concatenation

```
word1 = "Hello"
word2 = “World"
print(word1 + word2)
>>> 'HelloWorld’
print(word1 + ‘ ‘+word2)
>>> 'Hello World’
```

Concatenation different types

```
word1 = "Hello“
first_int=2
print(word1+first_int)
>>>  File "<input>", line 1, in <module>TypeError: can only concatenate str (not "int") to str
first_int_str=first_int.__str__()
print(word1+first_int_str)
>>> 'Hello2’
```

Strings Built In Methods

https://www.w3schools.com/python/python_ref_string.asp

- Len(len(str))- the function returns the number of items (length) in an object.
- Upper(upper.str)- Converts a string into upper case.
- Lower(lower.str)- Converts a string into lower case.
- index(index.str)- return the letter index.
- Replace(replace.str)- Returns a string where a specified value is replaced with a specified value.
- Count(count.str)- Returns the number of times a specified value occurs in a string.
- Split (str. Split())- split a string according to an argument(default-spices)

Input

```
input("Enter a String:")
>>>  Enter a String
>>> ‘It’s a lovely day’

Int(input("Enter a Number:"))
>>>  Enter a Number
>>> 3
```

Questions?

Hands On

Comparison Operators

- Compares two variables and returns a Boolean type result/variable

Operator	Name	Description
x < y	Less than	true if x is less than y, otherwise false.
x > y	Greater than	true if x is greater than y, otherwise false.
x <= y	Less than or equal to	true if x is less than or equal to y, otherwise false.
x >= y	Greater than or equal to	true if x is greater than or equal to y, otherwise false.
x == y	Equal	true if x equals y, otherwise false.
x != y	Not Equal	true if x is not equal to y, otherwise false.

Logical Operators

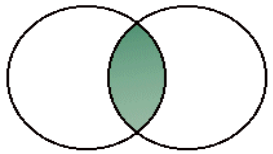
Operates on two Booleans and returns a Boolean

Operator	Description
x **and** y	Both True: **True**, otherwise: **False**.
x **or** y	At least one is True: **True**, Otherwise: **False**.
not x	x is False → **True**, x is True → **False**

And, or, not

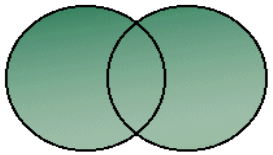
and

- The guy is tall **and** nice



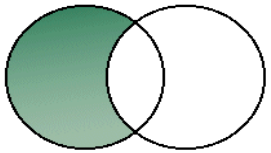
or

- The guy is either tall **or** nice



not

- The guy is **not** tall



Questions?

Hands On

Summary

- Variables
  - Numbers
  - Strings
- Computational Operators
- Logical Operators

Homework

1. create a Python script which accepts the user's first and last name and print them in reverse order with a space between them(use index())

input:

>>> 'Dor Shani'

Output:

>>> 'Hello Shani Dor'

2. create a Python script which accepts an integer (n) and computes the value of n+nn+nnn.

input:

>>>n =5

Output:

>>> 615

Homework

3. create a Python script which get a string from a given string where all occurrences of its first char have been changed to '$',except the first char itself.

Input:

>>> ‘restart’

Output:

>>> ‘resta$t’

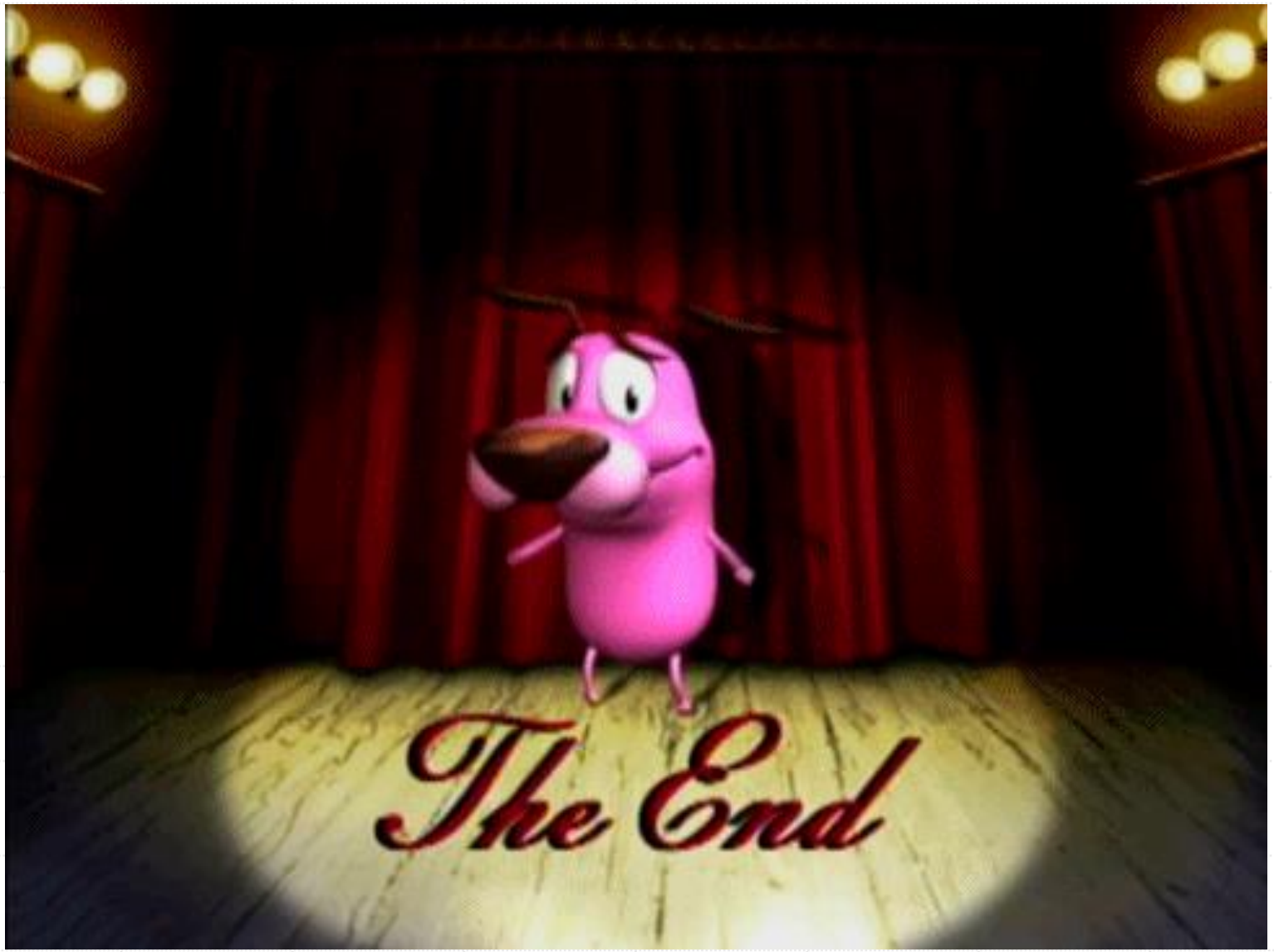
4. create a Python script which single string from two given strings, separated by a space and swap the first two characters of each string.

Input:

>>> ‘abc’, ‘xyz’

Output:

>>> ‘xyc’, ‘abz’


The End