



Section (1) **Data Mining**

Nagwa El-Araby – Mohammed El-Babeer – Aml Magdi
Information System Department
2019 – 2020



General Rules

- The attendance is only in your section.
- You should sign with your name in the attendance form.
- Every week, attendance is delivered to the professor of the course.
- The section has 10 marks from the total degree.
- There will be an evaluation project in groups (Maximum 3 members).
- Finally, keep smiling as there is no impossible ...



Python for Data Science

- Python is an open source language.
- It is widely used as a high-level programming language for general-purpose programming.
- It has gained high popularity in data science world.
- Python has awesome robust libraries for machine learning, natural language processing, deep learning, big data and artificial Intelligence.
- It can be easily integrated with big data frameworks such as Spark and Hadoop.
- Python has a great online community support.

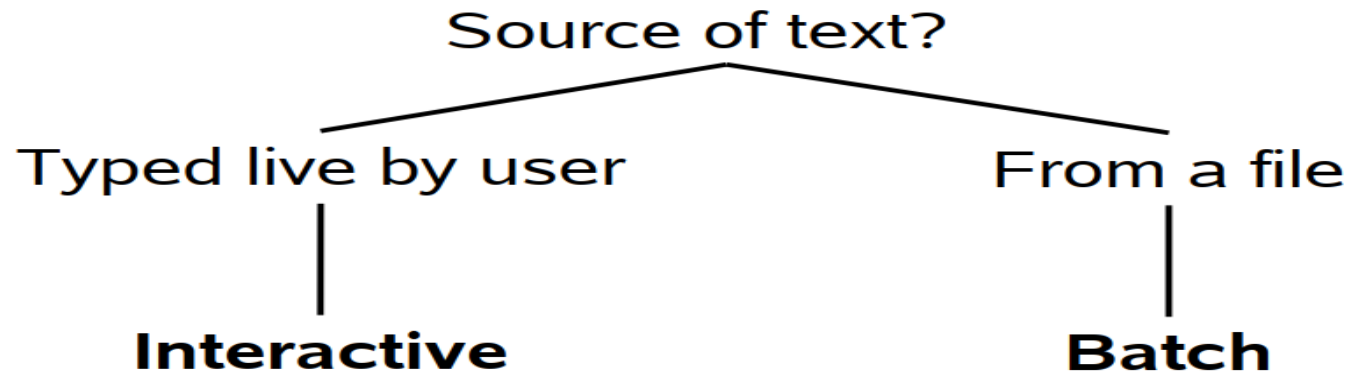


How to Install Python

- There are two ways to download and install Python :
 - Download Python from its official website. You have to manually install libraries.
 - <https://www.python.org/downloads/>
 - Download Anaconda. It comes with Python software along with preinstalled popular libraries.
 - <https://www.anaconda.com/distribution/>
 - Anaconda comes with two popular IDEs :
 - Jupyter (Ipython) Notebook
 - Spyder

Coding Environments

- Python command line:



> python

Unix command

Unix prompt

**Bold face means
you type it.**

Coding Environments (CONT.)

- Python command line (typed live by user):

> **python**

Python command

Python 2.5.1 (r251:54863, Aug 1 2008 00:32:16)

[GCC 4.2.1 (SUSE Linux)]

Command "argument"

Type "help", "copyright", "credits" or "license"...

>>> **print 'Hello, world!'**

Hello, world!

Result of command

>>>

Python prompt

Coding Environments (CONT.)

- Python command line (from a file) :

Unix command to show file content

Name of file whose content we want

```
> more hello.py
```

```
print 'Hello, world!'
```

File content

```
> python hello.py
```

```
Hello, world!
```

Name of file passed to Python

```
>
```

Unix command to run Python

Result of Python commands

Coding Environments (CONT.)

- Anaconda:
 - **Spyder**
 - It is like RStudio for Python. It gives an environment wherein writing python code is user-friendly.
 - It comes with a syntax editor where you can write programs. It has a console to check each and every line of code.
 - Under the 'Variable explorer', you can access your created data files and function.



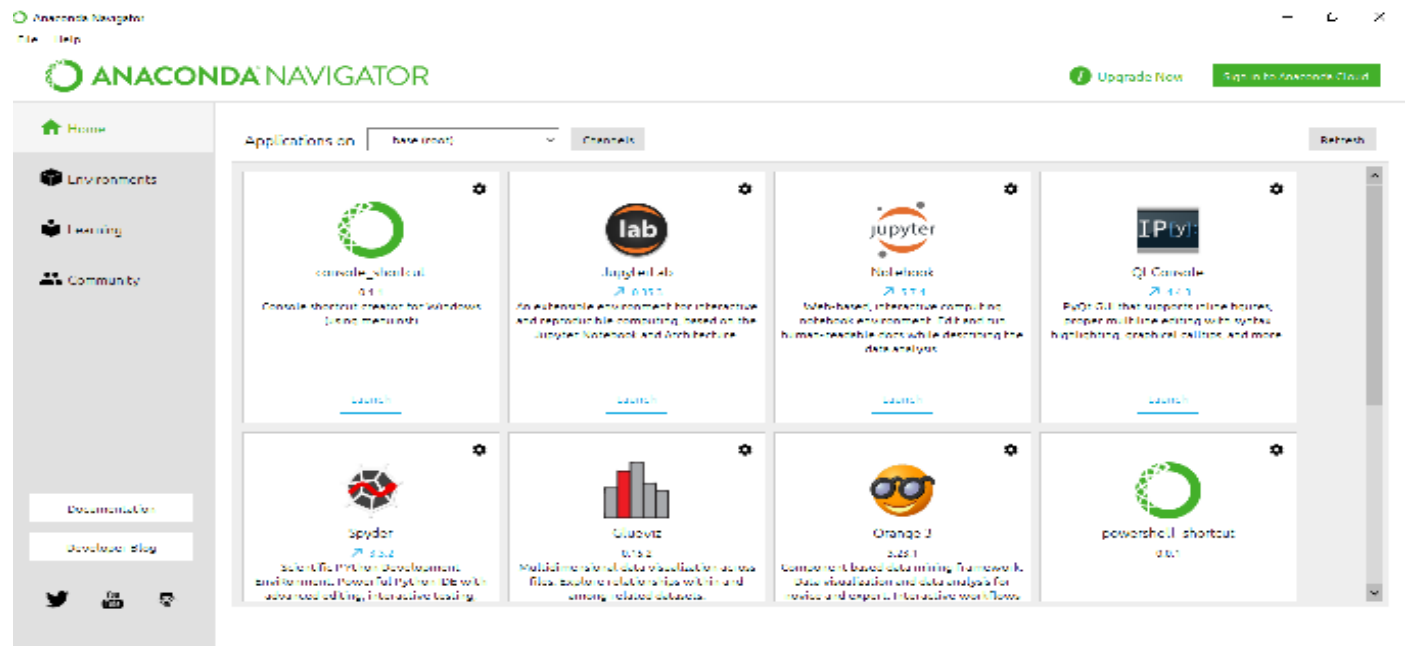
Coding Environments (CONT.)

- Anaconda:
 - **Jupyter (Ipython) Notebook**
 - The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.
 - Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.



Getting started with Jupyter

- Jupyter Environment:
 - You can open Jupyter with two ways:
 - Open new tab in your favorite browser then write: <http://localhost:8888/>
 - Open Anaconda Navigator then choose Jupyter.



Exercises – Python Fundamentals

- Exercise (1) :
 - This exercise explores :
 - Variables
 - Functions
 - Conditions
 - Loops
 - In this exercise, we will find:
 - To make a comment , use `#` . What are the benefits of comments ?!
 - `Print()` : to display something like texts , variables' values, etc.
 - `Print()` function can take a parameter called 'sep' , to separate between texts.
 - An escape character is a special character that is preceded with a backslash (`\`)
 - `\n` new line, `\t` tab, `\"`, `\'`, and `\\`
 - `%s`
 - `print("Hello , %s "% "Data Mining")`





Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - To define a variable, give it just a name and value.
 - Assignment operator is used to assign a value to the declared variable.
 - Like : `age = 20`
 - List of arithmetic operators with examples:

Arithmetic Operators	Operation	Example
+	Addition	$10 + 2 = 12$
-	Subtraction	$10 - 2 = 8$
*	Multiplication	$10 * 2 = 20$
/	Division	$10 / 2 = 5.0$
%	Modulus (Remainder)	$10 \% 3 = 1$
**	Power	$10 ** 2 = 100$
//	Floor	$17 // 3 = 5$
$(x + (d-1)) // d$	Ceiling	$(17 +(3-1)) // 3 = 6$



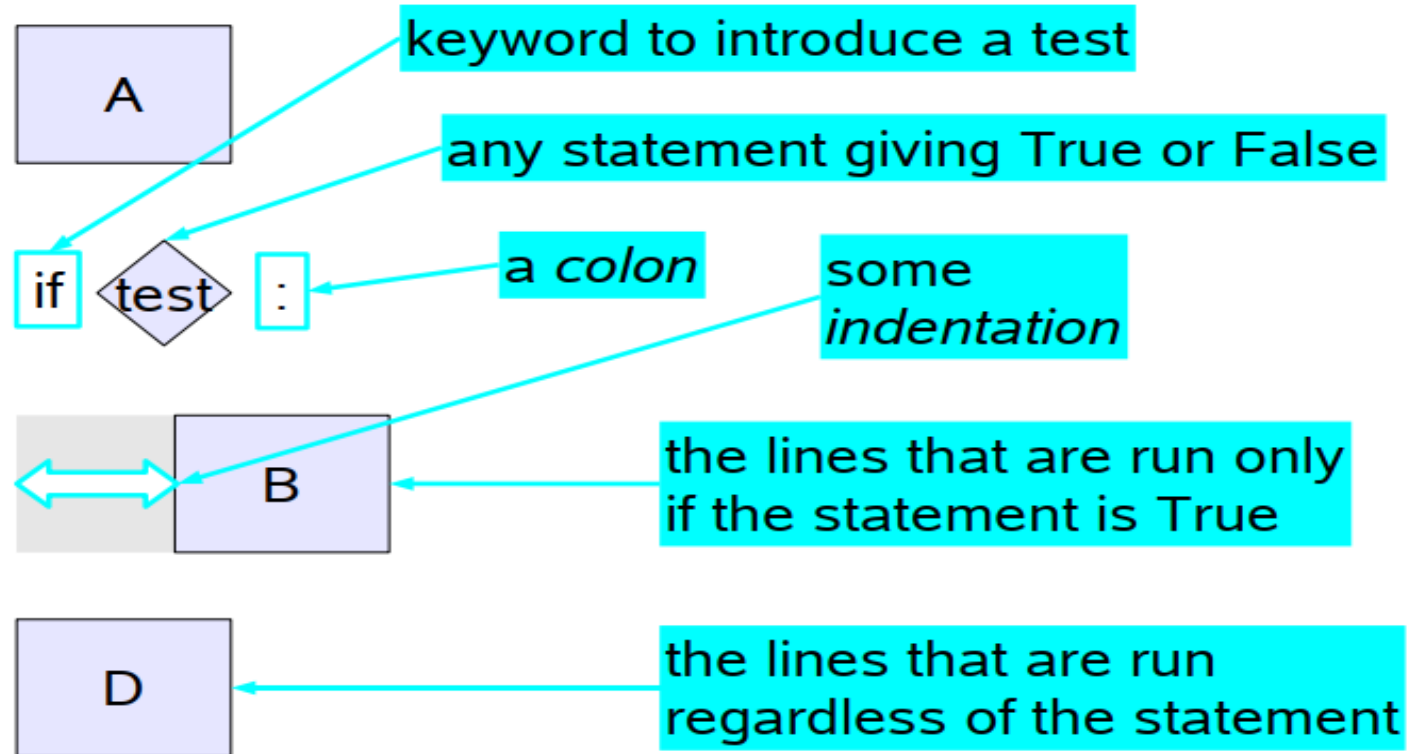
Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - List of Comparison & Logical Operators

Comparison & Logical Operators	Description	Example
>	Greater than	5 > 3 returns True
<	Less than	5 < 3 returns False
>=	Greater than or equal to	5 >= 3 returns True
<=	Less than or equal to	5 <= 3 return False
==	Equal to	5 == 3 returns False
!=	Not equal to	5 != 3 returns True
and	Check both the conditions	x > 18 and x <=35
or	If atleast one condition hold True	x > 35 or x < 60
not	Opposite of Condition	not(x>7)

Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - If, else, elif conditions



Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - If, else, elif conditions

A

if  test :

 B

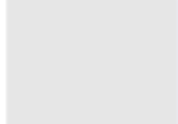
else :

 C

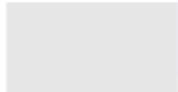
D

answer = 6*7

if answer == 42 :

 print 'Life, the universe,'
print 'and everything!'

else :

 print 'Wrong answer!'

print answer

Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - 1. Function starts with **def** keyword followed by **function name** and **()**.
 - 2. Function body starts with a colon **(:)** and is **indented**
 - 3. The keyword **return** ends a function and give value of previous expression.

```
def sum_fun(a, b):  
    result = a + b  
    return result
```

```
z = sum_fun(10, 15)
```

Result : z = 25

Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - Examples of built in functions :
 - **print()**, explained previously.
 - **range()** function returns a sequence of numbers, starting from 0 by default, and increments by 1 (by default), and ends at a specified number.
 - **input()** function allows user input.
 - The returned value is a string
 - **int()** function converts the specified value into an integer number.
 - **type()** function is used to get the type of a given variable.



Exercises – Python Fundamentals (CONT.)

- Exercise (1) :
 - In this exercise, we will find:
 - Loop
 - While loop , for loop
 - Syntax :
 - while expression:
statement(s)
 - for iterator_var in sequence:
statements(s)

A

```
epsilon = 1.0
```

while  test :

```
while 1.0 + epsilon > 1.0 :
```

B

```
epsilon = epsilon / 2.0
```

D

```
epsilon = epsilon * 2.0  
print epsilon
```



Assignments – Python Fundamentals

- **Assignment (1):**

- We want to know your grade (Pass / Good / Very Good / Excellent) from your degree. So , you should receive the following values from the user : Name , Department , Degree . Then show the following (Demo) : Hello Mohammed , “Your department is IS and your degree is <<GRADE >>!” .



Assignments – Python Fundamentals

● Assignment (2):

- BMI (Body Mass Index) is a measurement of body fat based on height and weight that applies to both men and women between the ages of 18 and 65 years. BMI can be used to indicate if you are overweight, obese, underweight or normal.
- You should receive the following fields : Name , Weight (KG), Height (CM).
- Use the following formula to calculate BMI :
$$\text{BMI} = (\text{Weight in Kilograms} / (\text{Height in Meters} \times \text{Height in Meters}))$$

- Notes :

- `bmi <= 18.5` then UNDERWEIGHT
- `bmi > 18.5 AND bmi <= 24.9` then NORMAL WEIGHT
- `bmi > 24.9 AND num <= 29.9` then OVERWEIGHT
- `bmi > 30.0` then OBESE

- Demo for the final result:

- "Your BMI value is <<BMI_VALUE>> and you are : <<STATUS>>

