

# لغة البايثون

## الهدف من الجلسة

- التعريف بلغة البايثون

## لغة البايثون

من اللغات عالية المستوى التي يتم تفسيرها.

يتم استخدامها في العديد من المجالات نظراً لسهولة استخدامها ومرونتها ووجود مجتمع ضخم يستخدم هذه اللغة.

## أساسيات لغة البايثون:

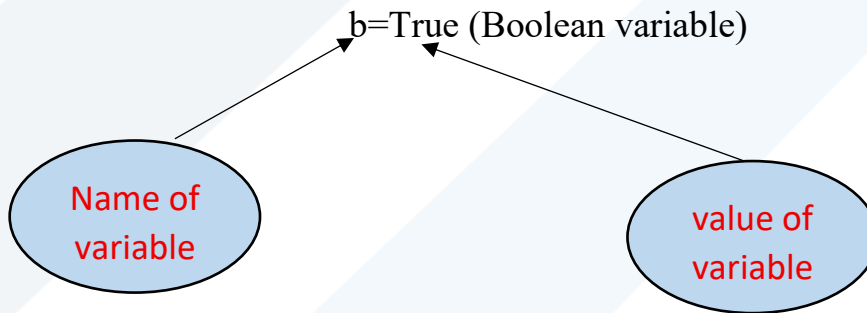
تعريف المتغيرات:

يتم تعريف المتغير عن طريق كتابة اسمه واسناد قيمة له، مثال:

x=10 (int variable)

s=" hello world" (string variable)

b=True (Boolean variable)



عمليات الإدخال والطباعة الأساسية:

الإدخال من لوحة المفاتيح:

عن طريق التابع input:

age=input ("add age number")



الطباعة على الشاشة:

ع طريق التعليمة print:

age=10

print("age of student is:", age)

Message  
to print

Result is: age of student is: 10

## بنى الاختبار:

بنية if-else:

1) If condition:

Statement

مثال:

```
age=10
if age<18:
    print ("under age")
```

2) if-elif-else condition:

statement

elif condition:

statement

else:

statement

مثال:

```
age=15
if age>=20:
    print("adult")
elif age>=14 and age<=19:
    print("teenager")
else:
    print("child")
```

result is: teenager

تمرين 1:

اكتب برنامج بلغة بايثون يسمح للمستخدم بإدخال علامته بالشكل التالي:

1. إذا كانت العلامة بين 90 و 100 يطبع A
2. إذا كانت العلامة بين 80 و 89 يطبع B
3. إذا كانت العلامة بين 70 و 79 يطبع C
4. إذا كانت العلامة بين 60 و 69 يطبع D
5. أقل من 60 يطبع F

## بنى التكرار:

حلقة for:

1- for iteration\_variable in range (start, end, step)

Example 1:

```
for i in range(1,6,2):
    print(i)
```

result:

1

3

5

Example 2:

```
for i in range(3):
    print(i)
```

result:

0  
1  
2

2- for iteration\_variable in data\_structure  
example:

```
fruits=['apple', 'banana', 'orange']  
for i in fruits:  
    print(i)
```

result:

apple

banana

orange

while loop:

While condition:

statement

Example:

```
cnt = 0  
while (cnt < 3):  
    cnt = cnt + 1  
    print("Hello world")
```

result:

Hello world

Hello world

## تمرين 2:

اكتب برنامج بلغة بايثون لطباعة الأعداد الزوجية بين 10 و 20، مرة باستخدام حلقة for ومرة باستخدام حلقة while.

## بنى المعطيات:

### :Lists

تخزن عناصر مختلفة وكل عنصر يملك index، وهي قابلة للتعديل:

```
List = [1, 2, 3, "string", 2.3] -1
```

```
[[‘apple’,’red’], ['banana',’yellow’],1] -2
```

### :Tuples

تخزن عناصر مختلفة وكل عنصر يملك index، لكنها غير قابلة للتغيير:

```
my_tuple = (1, "world", 2.71)
```

### :Sets

مجموعات غير مرتبة من العناصر الفريدة:

```
my_set = {1, 2, 3, 2, 4}
```

### :dictionary

مجموعات غير مرتبة من أزواج القيمة الرئيسية:

```
my_dict = {"name": "Alice", "age": 30, "city": "New York"}
```

### تمرين 3:

عدل الكود في الطلب الثاني ليطبوع مجموعة القيم الموجودة في اللائحة التالية:

```
[[‘apple’,’red’], ['banana',’yellow’],1]
```

و ال dictionary التالي:

```
{"name": "Alice", "age": 30, "city": "New York"}
```

## التوابع

تعريف التابع:

```
def name_of_method (params)
```

مثال 1:

```
def my_function():  
    print("Hello from a function")
```

مثال 2:

```
def fahrenheit_to_celsius(fahrenheit):  
    return (fahrenheit - 32) * 5 / 9
```

#### تمرين 4:

اكتب تابع يعيد باقي قسمة عدد  $x$  على عدد آخر  $y$ .