

دورة تعلم البرمجة بلغة البايثون

اعداد : م.محمد عبد المطلب

eng.mohammed7795@gmail.com

الدرس الثاني - محتويات

■ علامات التنصيص – Quotes

- أنواعها واستخداماتها

■ الدوال المبنية – Build-in function

- الدالة المبنية Type

- الدالة Str

- الدالة print

■ السلاسل – Strings

- ماهي؟

- العمليات عليها - operations

- هياؤها - strings format

- الطرق - methods

■ الارقام – Numbers

- الاعداد الصحيحة – integer

- الاعداد العشرية – float

- المركبة – complex

- العمليات عليها (+ ، - ، ÷ ، ×)

السلاسل - STRINGS

■ ماهي؟

هي كل شي بين علامتي تنقيص في لغة البايثون، ولا يخلو كود بايثون منها ابدا

>>> **hello** ليست سلسلة

>>> **everybody fine** ليست سلسلة

>>> "hello" هذه سلسلة

>>> "everybody fine" سلسلة

■ العمليات على السلاسل

هي عملية تستخدم لربط سلسلة مع سلسلة اخري أو تكرار السلسلة اكثر من مرة

١- عملية الربط

تقوم هذه العملية بربط او جمع سلسلتين او اكثر وتتم هذه باستخدام رمز الجمع + العادية

"hello " + "everyone"

➔ "hello everyone"

"hello " + "everyone " + "here"

➔ "hello everyone here"

السلاسل - STRINGS

■ العمليات على السلاسل

هي عملية تستخدم لربط سلسلة مع سلسلة أخرى أو **تكرار** السلسلة أكثر من مرة

٢- عملية التكرار

تقوم هذه العملية بتكرار سلسلة واحدة أكثر من مرة وتتم هذه باستخدام رمز الضرب (النجمة) *

“hello ” * 2



“hello hello”

“hello ” * 4



“hello hello hello hello”

“hello everyone ” * 2



“hello everyone hello everyone”

السلاسل - STRINGS

■ هيئات السلاسل – Strings format

هي استخدام بعض الرموز الخاصة مثل **%f**, **%o**, **%d**, **%s** وغيرها في الاشارة الي سلسلة او رقم او غيرها

➤ **%s Format**

تقوم هذه بعملية تعويض او استبدال الرمز **%s** ب سلسلة مكانها

<code>“hello %s” % “mohammed”</code>	➔	<code>“hello mohammed”</code>
<code>“hello %s” % “everyone”</code>	➔	<code>“hello everyone”</code>
<code>“hello %s here” % “everyone”</code>	➔	<code>“hello everyone here”</code>
<code>“hello %s %s” % (“everyone”, “here”)</code>	➔	<code>“hello everyone here”</code>

السلاسل - STRINGS

■ هيئات السلاسل ٢ - Strings format 2

هي تحتوي على ارقام بين علامة ال % و الحروف **f, o, d, s** مثل **%4f, %3o, %5d, %10s** وغيرها في الاشارة الي سلسلة او رقم او غيرها. وعمل مسافة يمين (سالب) او يسار (موجب) بمقدار الرقم ناقص عدد حروف السلسلة

➤ **%(+Number)s Format**

تقوم هذه بعملية تعويض او استبدال الرمز **%s** ب سلسلة مكانها وعمل مسافة يسار (موجب) بمقدار الرقم ناقص عدد حروف السلسلة

“hello %3s” %”mohammed”



“hello mohammed”

“hello %9s” %”mohammed”



“hello mohammed”

“hello %10s” %”mohammed”



“hello mohammed”

“hello %12s” %”mohammed”



“hello mohammed”

السلاسل - STRINGS

■ هيئات السلاسل ٢ – Strings format 2

%(-Number)s Format

تقوم هذه بعملية تعويض او استبدال الرمز %s ب سلسلة مكانها وعمل مسافة يمين (سالب) بمقدار الرقم ناقص عدد حروف السلسلة

“hello %-3s” %”mohammed”

→ “hello mohammed”

“hello %-9s” %”mohammed”

→ “hello mohammed ”

“hello %-10s” %”mohammed”

→ “hello mohammed ”

“hello %-12s” %”mohammed”

→ “hello mohammed ”

السلاسل - STRINGS

■ الطرق – Methods

هي عبارة عن مميزات اضافة لل السلسلة تقوم مثلاً ب تحويل الحروف الي حروف كبيرة upper case او صغير lower case او جعل الحروف الاولي فقط كبيرة وغيرها من المميزات.

➤ string.method()

استعراض بعض انواع ال methods في السلاسل Strings

“hello”.upper()

➔ “HELLO”

“HELLO”.lower()

➔ “hello”

“hello everyone here”.title()

➔ “Hello Everyone Here”

“HeLLo eVerYoNe hErE”.swapcase()

➔ “hElIo EvERyOnE HeRe”

الارقام - NUMBERS

■ ماهي

هي عبارة عن كل رقم يكتب **من غير** علامة تنصيص

رقم 5 >>>

ليس رقم (سلسلة) >>> "5"

رقم 54.5 >>>

ليس رقم (سلسلة) >>> "545"

■ أنواعها

تقسم الارقام في لغة البايثون إلى ثلاثة أنواع رئيسية

١- الارقام الصحيحة – Integer (Int)

وهو اي رقم (**خارج علامة تنصيص**) **لا** يحتوي على فواصل عشرية او كسر او جزء تخيلي

عدد صحيح Int 22

ليس عدد صحيح 56.3

ليس عدد صحيح 65j

الارقام - NUMBERS

٢- الارقام العشرية - Floats

وهو اي رقم (خارج علامة تنصيص) يحتوي على فواصل عشرية او كسر ولا يحتوي على جزء تخيلي

5.6 → float 56÷3 → float 77 → not float (int)
65j → not float 55.00 → float

٣ - الارقام المركبة - Complex

وهو اي رقم يحتوي على جزء تخيلي (جزر سالب واحد) ويرمز له بالرمز j

5j → complex 5.2j → complex
5j+2 → complex 4 → not complex (int)
8.2 → not complex (float)

الارقام - NUMBERS

■ العمليات على الارقام

هي نفسها العمليات الرياضية العادية عملية الضرب و القسمة والجمع والطرح

١- عملية الجمع (+)

وتكون عملية جمع عديدين بطريقة العادية ويمكن جمع انواع الارقام المختلفة التي تطرقنا اليها سابقا

$$5 + 2 \rightarrow 7$$

$$\text{int} + \text{int} \rightarrow \text{int}$$

$$3 + 2j \rightarrow 3+2j$$

$$\text{int} + \text{complex} \rightarrow \text{complex}$$

$$3.2 + 5 \rightarrow 8.20$$

$$\text{float} + \text{int} \rightarrow \text{float}$$

$$5j + 9.5 \rightarrow 9.5+5j$$

$$\text{complex} + \text{float} \rightarrow \text{complex}$$

الارقام - NUMBERS

٢- عملية الضرب (*)

وتكون عملية ضرب عديدين بطريقة العادية ويمكن ضرب انواع الارقام المختلفة التي تطرقنا اليها سابقا

$$5 * 2 \rightarrow 10$$

$$\text{int} * \text{int} \rightarrow \text{int}$$

$$3 * 2j \rightarrow 6j$$

$$\text{int} * \text{complex} \rightarrow \text{complex}$$

$$3.2 * 5 \rightarrow 16.00$$

$$\text{float} * \text{int} \rightarrow \text{float}$$

$$5j * 9.5 \rightarrow 47.5j$$

$$\text{complex} * \text{float} \rightarrow \text{complex}$$

الارقام - NUMBERS

٣- عملية الطرح (-)

وتكون عملية **طرح** عديدين بالطريقة العادية ويمكن **طرح** انواع الارقام المختلفة التي تطرقنا اليها سابقا

$$5 - 2 \rightarrow 3$$

$$\text{int} - \text{int} \rightarrow \text{int}$$

$$3 - 2j \rightarrow 3-2j$$

$$\text{int} - \text{complex} \rightarrow \text{complex}$$

$$3.2 - 5 \rightarrow -1.8$$

$$\text{float} - \text{int} \rightarrow \text{float}$$

$$5j - 9.5 \rightarrow 9.5-5j$$

$$\text{complex} - \text{float} \rightarrow \text{complex}$$

الارقام - NUMBERS

٤- عملية القسمة (/)

وتكون عملية قسمة عددين بطريقة العادية ويمكن قسمة انواع الارقام المختلفة التي تطرقنا اليها سابقا

$$5 / 2 \rightarrow 2.5$$

$$\text{int} / \text{int} \rightarrow \text{float}$$

$$10 / 2j \rightarrow 5j$$

$$\text{int} / \text{complex} \rightarrow \text{complex}$$

$$20 / 5 \rightarrow 4.0$$

$$\text{float} / \text{int} \rightarrow \text{float}$$

$$5j / 15 \rightarrow 3.33j$$

$$\text{complex} / \text{float} \rightarrow \text{complex}$$

السلاسل - STRINGS

■ هيئات السلاسل – Strings format

هي استخدام بعض الرموز الخاصة مثل **%f**, **%o**, **%d**, **%s** وغيرها في الاشارة الي سلسلة او رقم او غيرها

➤ **%d** Format

تقوم هذه بعملية تعويض او استبدال الرمز **%d** ب **عدد صحيح integer** مكانها

"hello %d" %5	➔	"hello 5"
"hello %d" %33	➔	"hello 33"
"hello %d here " %72	➔	"hello 72 here"
"hello %d %d" %(100 , 45)	➔	"hello 100 45"

السلاسل - STRINGS

■ هيئات السلاسل – Strings format

هي استخدام بعض الرموز الخاصة مثل **%f**, **%o**, **%d**, **%s** وغيرها في الاشارة الي سلسلة او رقم او غيرها

➤ **%f** Format

تقوم هذه بعملية تعويض او استبدال الرمز **%f** ب **كسر عشري float** مكانها

“hello %f” %7.21	➔	“hello 7.21”
“hello %f” %5/2	➔	“hello 2.5”
“hello %f here ” %5+3.5	➔	“hello 8.5 here”
“hello %f %f” %(20/5 , 2*3)	➔	“hello 4.00 6.00”

علامات التنصيص - QUOTES

■ ماهي

هي علامات تستخدم لكتابة ال سلاسل داخلها

■ أنواعها واستخداماتها

هناك ٣ أنواع منها وهي :

- ١- علامة التنصيص المفردة - Single quotes (')
- ٢- علامة التنصيص المزدوجة - Double quotes ("")
- ٣- علامة التنصيص الثلاثية - Triple quotes ('''')

وتستخدم جميعا في عمل السلاسل

علامات التنصيص - QUOTES

■ التعرف عليها

١- علامة التنصيص المفردة - Single quotes (')

مشكلتها انها لا تعمل مع الاختصارات الانجليزية مثلا

'my name is mohammed' → name is == name's

'my name's mohammed' → **Error with Single Quotes**

٢- علامة التنصيص المزدوجة - Double quotes (")

مشكلتها انها تكتب في سطر واحد فقط ولا تدعم الكتابة على عدة اسطر

"my name's mohammed" → work

"my name is __

__mohammed" → **Error Multi lines**

علامات التنصيص - QUOTES

٣- علامة التنصيص الثلاثية – Tribe quotes (“”)

حلة مشكلة الكتابة في عدة اسطر (تدعم الكتابة في اكثر من سطر دون تقطع)

“”my name is __

__mohammed“” → works

“” my __

__ name is __

__mohammed“” → works

الدوال المبنية – BUILD-IN FUNCTIONS

■ تعريف

هي عبارة عن مجموعة اسطر برمجية لعمل غرض معين من طباعة او تحويلات او غيرها مختصرة تحت كلمة واحدة وتنتهي الكلمة ب قوسين () - **parentheses**

■ الدالة **type()**

تستخدم هذه الدالة للتعرف على نوع البيانات في البايتون سوا كانت **strings** سلسلة او عدد صحيح **integer** او كسر عشري **float** او عدد مركب **complex** او غيرها

>>> **type**("hello") → **strings**

>>> **type**(5) → **integer**

>>> **type**(77.5) → **float**

>>> **type**(9j) → **complex**

>>> **type**(6+2j) → **complex**

type(7+2) → **integer**

type(5/2) → **float**

الدوال المبنية – BUILD-IN FUNCTIONS

■ الدالة `str()`

هي اختصار ل كلمة `strings` وتعني سلاسل وتستخدم هذه الدالة لتحويل مايكتب داخلها الي سلاسل

<code>>>> str("hello")</code>	→	<code>"hello"</code>	→ <code>strings</code>
<code>>>> str(5)</code>	→	<code>"5"</code>	→ <code>strings</code>
<code>>>> str(77.5)</code>	→	<code>"77.5"</code>	→ <code>strings</code>
<code>>>> str(9j)</code>	→	<code>"9j"</code>	→ <code>strings</code>
<code>>>> str(6+2j)</code>	→	<code>"6+2j"</code>	→ <code>strings</code>

الدوال المبنية – BUILD-IN FUNCTIONS

■ الدالة `print()`

تعني بالعربية طباعة وهي تقوم بطباعة كل ما يكتب داخلها كما هو

<code>>>> print("hello")</code>	→ hello
<code>>>> print("my name is mohammed")</code>	→ my name is mohammed
<code>>>> print("hello" + "everyone")</code>	→ hello everyone
<code>>>> print("hello"*2)</code>	→ hello hello
<code>>>> print(5+2)</code>	→ 7
<code>>>> print(10/2)</code>	→ 5
<code>>>> print(30-10)</code>	→ 20
<code>>>> print(5*2)</code>	→ 10



THE END

eng.mohammed7795@gmail.com

اعداد وتقدم : م.محمد عبدالمطلب