

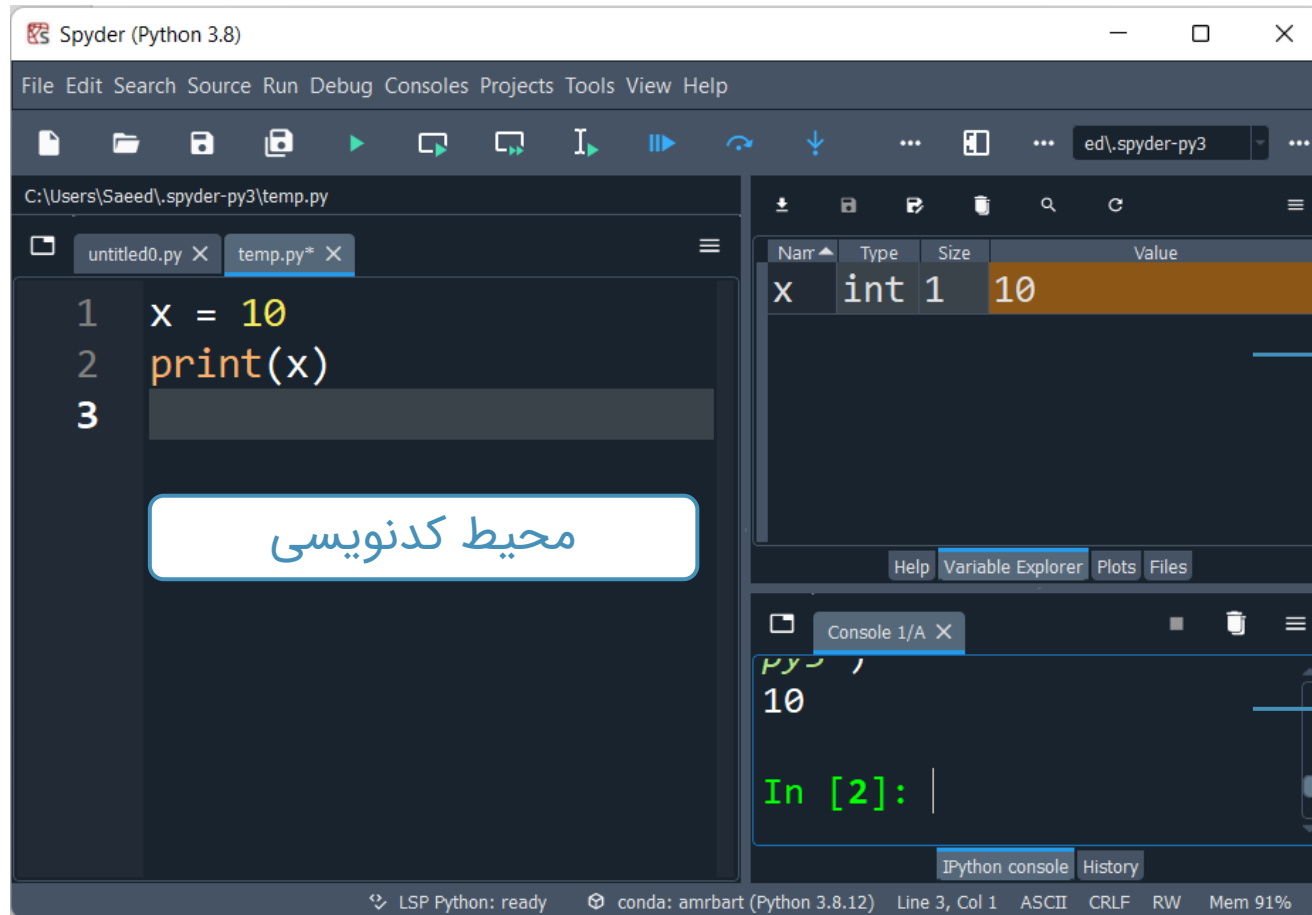


پایتون برای علم داده

سعید محققى

02: Getting Started

محیط Spyder



متغیرها

■ Spyder همراه Anaconda نصب می‌شود

خط فرمان (IPython)
و خروجی کد

<https://www.spyder-ide.org/>

نصب جداگانه Spyder (اگر Anaconda ندارید!)

محیط های کدنویسی آنلاین

- <https://www.w3schools.com/> 
- <https://www.tutorialspoint.com/codingground.htm> 
- <https://colab.research.google.com> 
- https://www.onlinegdb.com/online_python_compiler 

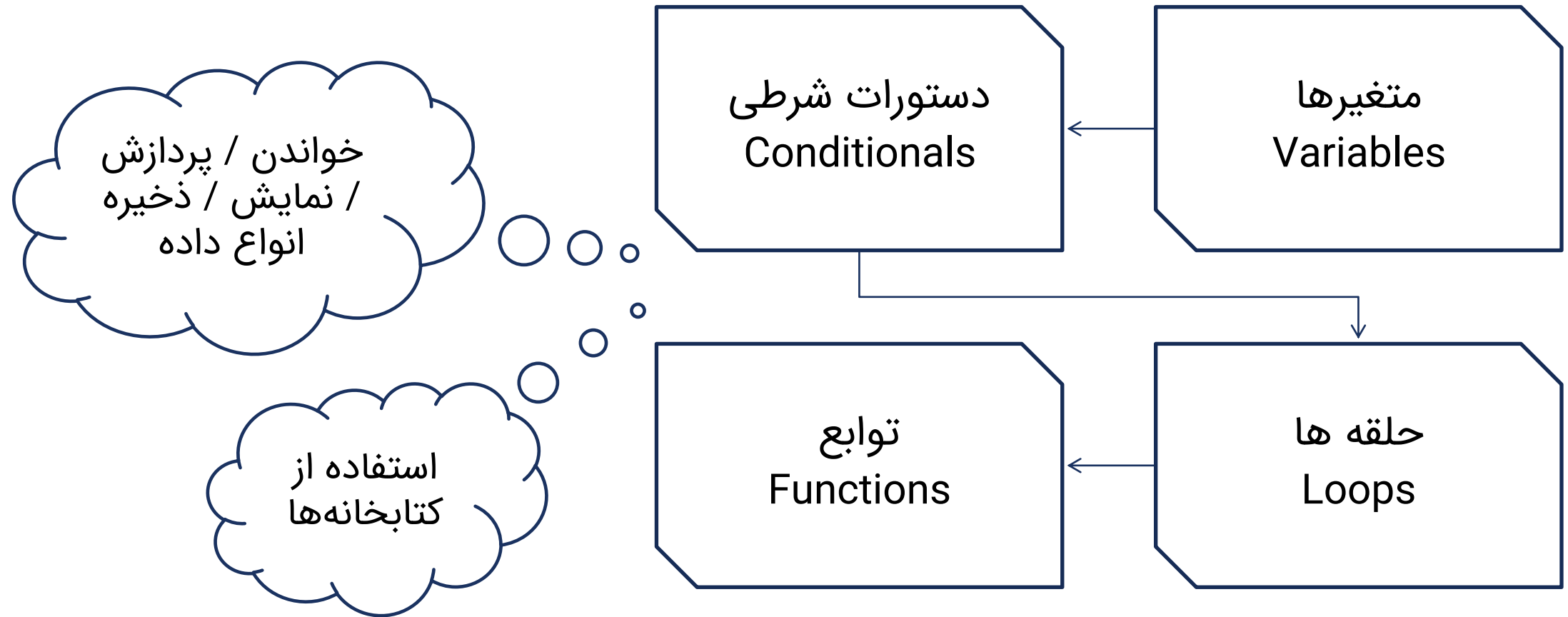
- <https://www.online-python.com/> 

- <https://onecompiler.com/python/> 

-> اجرای خط به خط

<http://www.pythontutor.com/live.html#mode=edit>

مراحل یادگیری پایتون



■ متغیرها

```
>>> x = 10

>>> a = b = 20

>>> name, age = 'Ali', 30
```

■ کامنت ها

```
# One line comment

'''
Multiple
Line
Comment
'''
```

■ کتابخانه ها

```
>>> import sys

>>> from sys import *

>>> from sys import version

>>> import pandas as pd
```

متغیرها (Variables)

■ هر متغیر، سه خصوصیت دارد: نام / مقدار / نوع

نام (ID)	مقدار (Value)	نوع (Type)
x	10	integer
y	1.25	float
firstname	"saeed"	string
is_ok	True	boolean

انواع متغیر پایه

`x = 2` → `integer (int)`
`x = 3.14` → `float / double`
`x = 'test'` → `string (str)`
`x = False` → `Boolean (bool)`

`x = [1, 2, 3]`
`x = [1, 2.5, 'a']` } → `list`
`x = (1, 2, 3)`
`x = (1, 2.5, 'a')` } → `tuple`

`x = {1, 2, 3}`
`x = {1, 2.5, 'a'}` } → `set`
`x = {'a':1, 'b':'c'}`
`x = {1:2, 3:'a'}` } → `dictionary`

✓ نوع متغیرها در حین اجرا با توجه به مقادیر آن ها تعیین می شود.

✓ تشخیص نوع متغیر با دستور `type(var)`

✓ انواع `list`، `tuple` و `set` قابل تبدیل به یکدیگر هستند.

✓ آیتم های `list` قابل تغییر ولی در `tuple` غیر قابل تغییر هستند.

✓ در `set` و `dict`، آیتم تکراری وجود نخواهد داشت.

قوانین ساختاری در کدنویسی پایتون

- دستورها و نام توابع و متغیرها به حروف بزرگ و کوچک حساس هستند. (Case Sensitive)
- تعداد فاصله و تب در ابتدای هر خط (Indentation) مهم است.
- نام توابع و متغیرها فقط می تواند شامل حروف، اعداد و _ باشد و نباید با عدد شروع شود.
(از اسامی رزرو شده هم نباشد).
- نامهای چند کلمه ای:

varName, _varName	→ camelCase	✓
VarName, _VarName	→ PascalCase	✓
var_name, _var_name	→ snake_case	✓
2varName, var\$name	→ invalid	X

متن (String):

■ دسترسی به کاراکترهای متن با شمارش: Indexing

```
>>> s = 'Hello World!'
>>> print(s[0])
>>> print(s[0:5])
>>> print(s[-6:-1])
>>> print(s[-6:])
```

■ ترکیب متن با متن / متن با انواع دیگر متغیر: Concatenation

```
>>> a = "Hello"
>>> b = "World!"
>>> print(a + " " + b)
```

```
>>> x = 10
>>> print("x = " + x)
>>> print("x =", x)
```

```
>>> print("=" * 50)
>>> i = 5
>>> print("-\_(ツ)_/ " * i)
```

کار با متغیرهای عددی در پایتون

■ پروژه 1: محاسبه شاخص توده بدنی (BMI calculator)

1. دریافت وزن کاربر

2. دریافت قد کاربر

3. محاسبه BMI

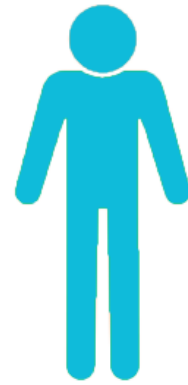
4. گزارش وضعیت کاربر

What will we learn? (Keywords)

- string / int / float Types
- Arithmetic Operations
- Conditional Operations
- String Formatting

Body Mass Index (BMI)

$$BMI = \frac{\text{Weight in } kg}{(\text{Height in } m)^2}$$



< 18.5
UNDERWEIGHT



18.5-24.9
NORMAL



25-29.9
OVERWEIGHT



30 <
OBESE

BMI	Status
< 18.5	Underweight
18.5 – 24.9	Normal
25 – 29.9	Overweight
30 – 34.9	Obesity Class I
35 – 39.9	Obesity Class II
> 40	Obesity Class III

Python Arithmetic Operators

Operator	Name	Example	Assignment	Example
+	Summation	x + 2	+=	x += 2
-	Subtraction	x - 3	-=	x -= 3
*	Multiplication	x * 3	*=	x *= 3
/	Division	x / 2	/=	x /= 2
**	Exponentiation	x ** 2	**=	x **= 2
%	Remainder	x % 2	%=	x %= 2

Result:

⇒ **int or float**

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

all operators

Python Comparison Operators

Operator	Name	Example
<code>==</code>	Equal	<code>x == y</code>
<code>!=</code>	Not equal	<code>x != y</code>
<code>></code>	Greater than	<code>x > y</code>
<code><</code>	Less than	<code>x < y</code>
<code>>=</code>	Greater than or equal to	<code>x >= y</code>
<code><=</code>	Less than or equal to	<code>x <= y</code>

logic	Example
<code>and</code>	<code>x and y</code>
<code>or</code>	<code>x or y</code>
<code>not</code>	<code>not (x and y)</code>

Result: Boolean

⇒ **True** or **False**

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

all operators

دستورهای شرطی

if *logical condition* :
→ *statements block*

```
if condition:  
    do this  
else:  
    do this
```

```
if condition1:  
    do this  
elif condition2:  
    do this  
else:  
    do this
```

- دستور if می تواند با چند دستور elif و تنها یک دستور else در انتها، همراه شود.
- دستورهای درون شرط if، elif و else باید به اندازه یک Tab یا 4 فاصله، جلوتر باشند.

عملیات بر روی رشته ها: String Formatting

■ ترکیب رشته ها و متغیرها

```
>>> x, y = 10, 2.5  
>>> s = 'values are %d, %f' %(x,y)  
>>> s = 'values are {}, {}'.format(x,y)  
>>> s = f'values are {x}, {y}'
```

Limit/round numbers:

- ~~%.2f~~ , ~~%6.2f~~
- {:.2f} , {:6.2f}
- {y:.2f} , {y:6.2f}

<https://realpython.com/python-f-strings/>

From % to f-String