禪

[illegible] - 4

[illegible] (python) - [illegible]

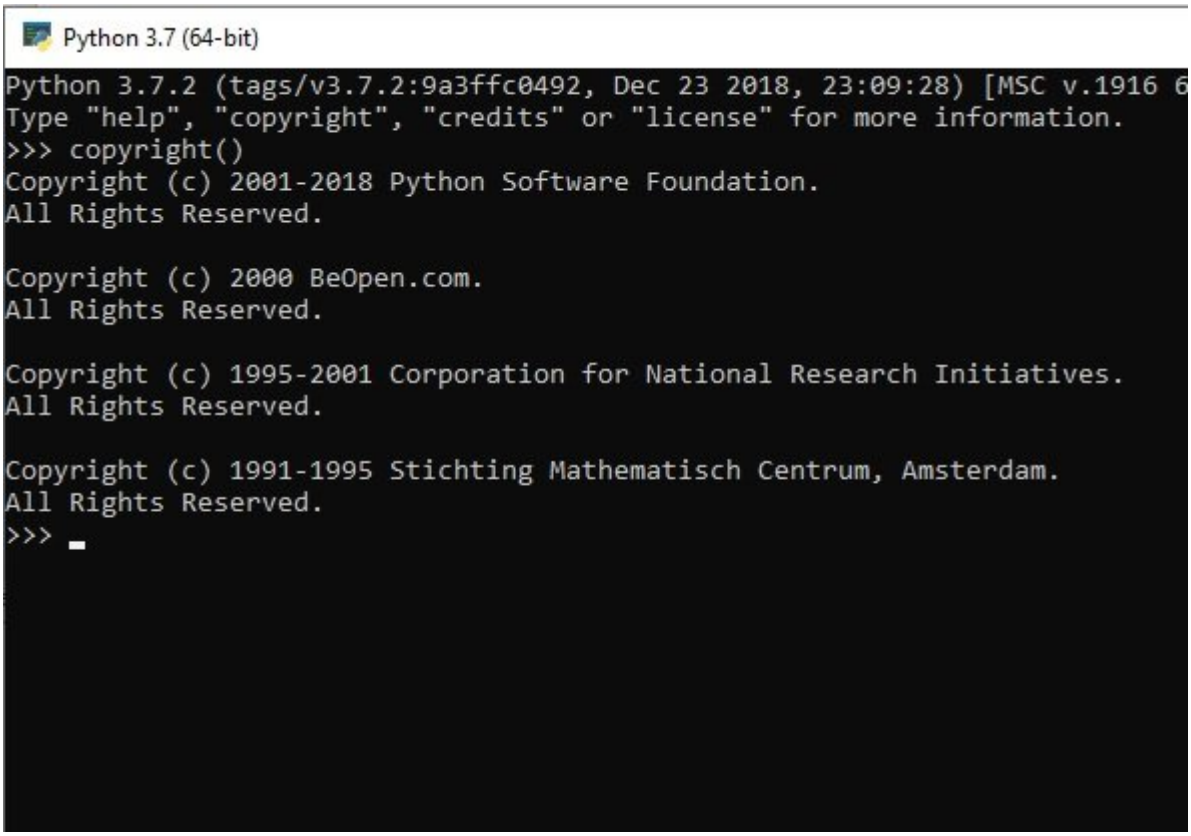
[illegible]

[illegible]

[illegible]

[illegible] 10 python [illegible]

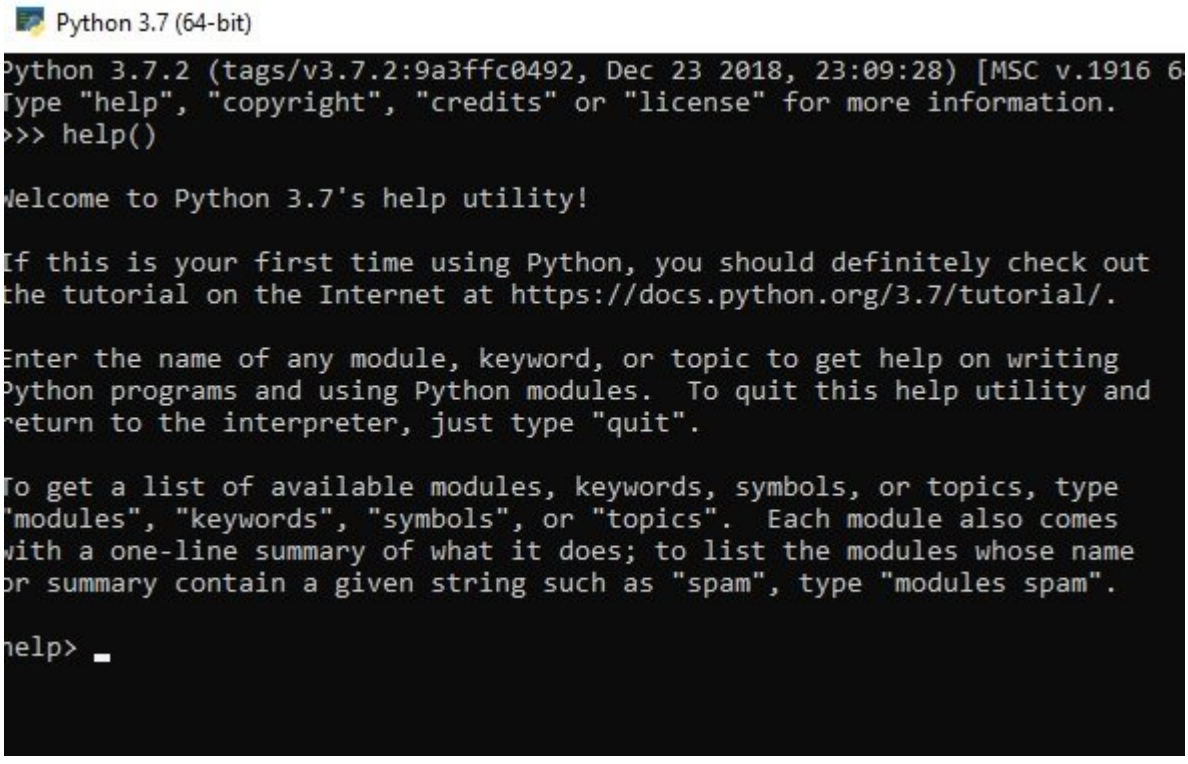
Copyright(): [illegible]



Credits(): [illegible].

License(): [illegible].

Help(): [illegible]. (You [illegible].)



□□□ □□ □□□□□ □□□□□□□ □□□□□ □□□□ □□□□□ □□□□□□□ □□□□ □□ □□□□□ □□□□ □□□□□□ □□□□ □□ □□ □□□□□□ □□ □□ □□□□ □□ □□□□ □□□□□□ □□□□□□ □□ □□□□□□ □□ □□□□□ □□□□□□□ □□□□□ □□□□ □□□ □□□□□ □□□□□. □□□□□ □□□□□ □ □□□□□□□ □□□□□□□□□□□□□ □□□□□□ □□ □□□□□□□□□ topics □ modules□ keywords □□□□□□□□□□□□□ □□ □□□ □□□□□□ □□□□□□ □□□□ □□□ □□ topics □□□□ □□□□ □□ □□□□ □□□□□ .□□□□□□□ □□□□ □□□□ □□□□□

□□□□□ □□□□□□ □□□ □□ □□ □□ □□□□□□ □□ □□□□□□□□ □□□□□ □□□ □□ □□ □□ □□□□□□ □□□□□□ □□ □□□□□□ □□ □□□□□ FUNCTIONS □□□□ □□ □□□□□□ □□□□□ □□□ □□ □□□□□□ □□□□□□□. □□□□ □□□□ □□ □□□□□ □□□□ □ □□□□ □□□□ □□ FUNCTIONS □□□□ □□□ □□□□ □□□□ □□□□□□ □□□□ □□ □□□□ □□□). (□□□ □□□□□□□□ □□ □□ □□□□□ □□□□□ □□□□ □□□□□□ □□□□ □□ □□□□□ □□□□ □□ □□□□□ □□□□ □□□□.)

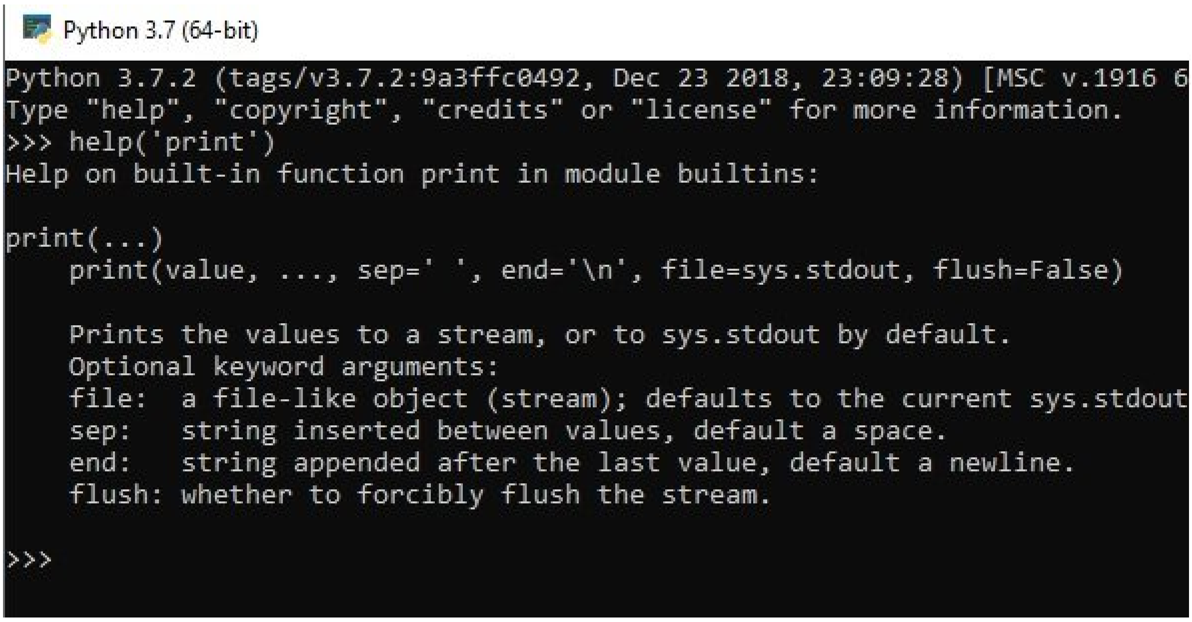
```
DEBUGGING           LITERALS            SEQUENCES

help> FUNCTIONS
Functions
*********

Function objects are created by function definitions.  The only
operation on a function object is to call it: "func(argument-list)".

There are really two flavors of function objects: built-in functions
and user-defined functions.  Both support the same operation (to call
the function), but the implementation is different, hence the
different object types.

See Function definitions for more information.

Related help topics: def, TYPES

help>
```

```
help> print
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file:  a file-like object (stream); defaults to the current sys.stdout
    sep:   string inserted between values, default a space.
    end:   string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.

help>
```

[illegible]

[illegible]

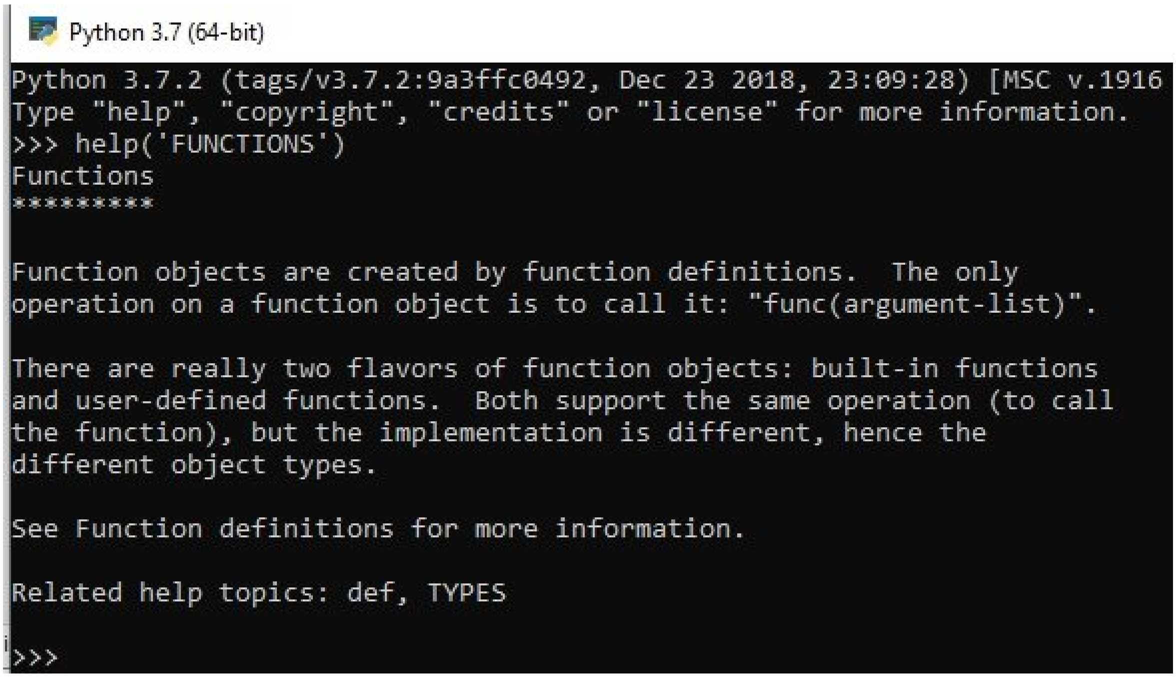
help('print')



```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916 6
Type "help", "copyright", "credits" or "license" for more information.
>>> help('print')
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file:  a file-like object (stream); defaults to the current sys.stdout
    sep:   string inserted between values, default a space.
    end:   string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.

>>>
```

[illegible] FUNCTIONS [illegible]

help('FUNCTIONS')



```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 23:09:28) [MSC v.1916
Type "help", "copyright", "credits" or "license" for more information.
>>> help('FUNCTIONS')
Functions
*********

Function objects are created by function definitions.  The only
operation on a function object is to call it: "func(argument-list)".

There are really two flavors of function objects: built-in functions
and user-defined functions.  Both support the same operation (to call
the function), but the implementation is different, hence the
different object types.

See Function definitions for more information.

Related help topics: def, TYPES

>>>
```

[illegible] IDLE [illegible]

[illegible]

[illegible] x [illegible]

[illegible] exit() [illegible] quit() [illegible] ERRORLEVEL [illegible] quit() [illegible] exit() [illegible].

[illegible] os.exit() [illegible] sys.exit() [illegible].

[illegible]

[illegible] IDLE [illegible] IDLE [illegible] IDLE [illegible] Komodo Edit [illegible].

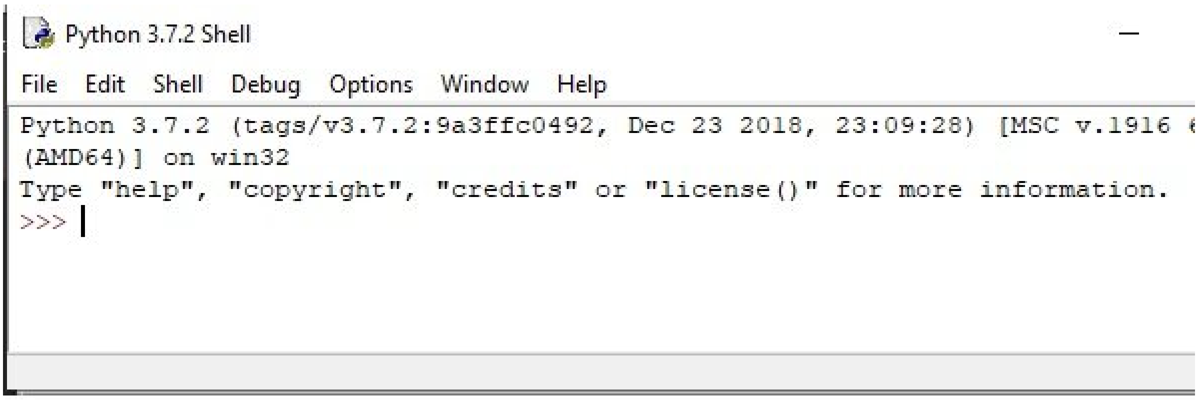
[illegible] IDLE

[illegible] IDLE [illegible]:

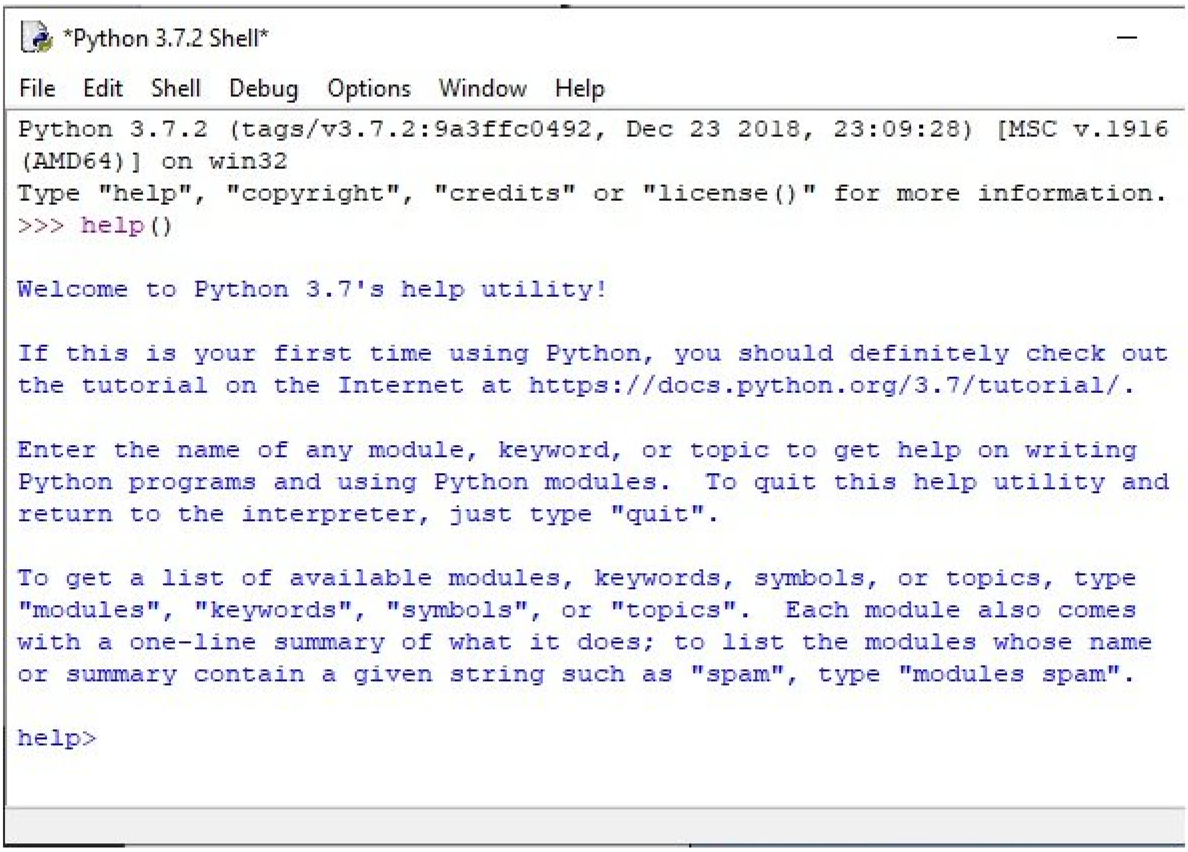
- [illegible]
- [illegible]
- [illegible] (…[illegible]) [illegible]
- [illegible]
- [illegible]
- [illegible] ([illegible])

IDLE [illegible]

IDLE [illegible] IDLE [illegible] IDLE [illegible] 10 [illegible] IDLE [illegible].

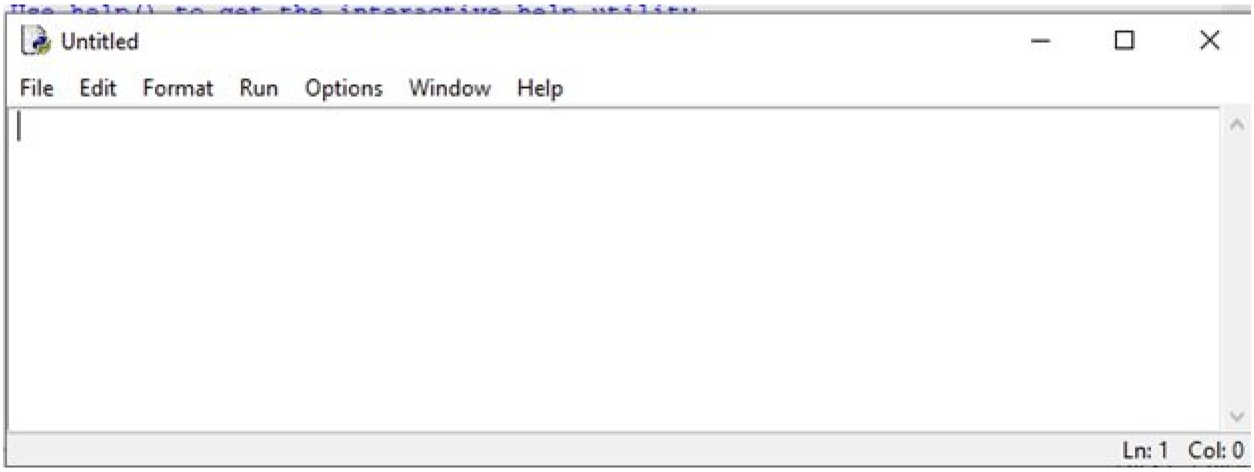


[illegible] help() [illegible]



[illegible]

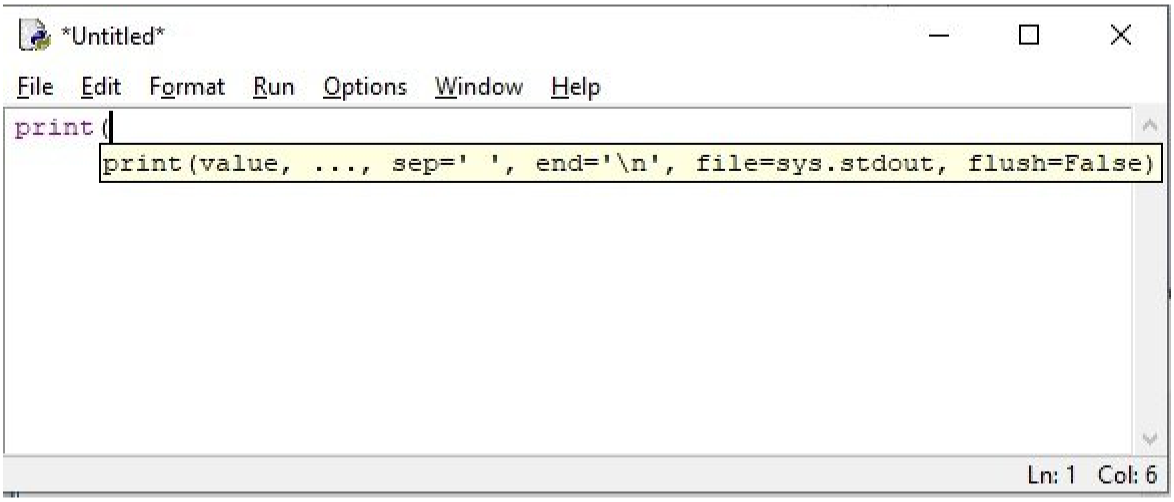
[illegible] IDLE [illegible] File→New File [illegible]



[illegible] Untitled [illegible] Shell 3.3.4 Python [illegible] [illegible]

[illegible]

[illegible] IDLE [illegible] ()print [illegible] print [illegible]



[illegible] print [illegible]

print(" [illegible] ")

[illegible]

ذخیره برنامه

در این مرحله برنامه خود را ذخیره می کنیم. در این مرحله می توانید برنامه خود را در هر جایی که می خواهید ذخیره کنید. برای ذخیره برنامه از منوی File گزینه Save را انتخاب کنید. پنجره زیر ظاهر می شود.

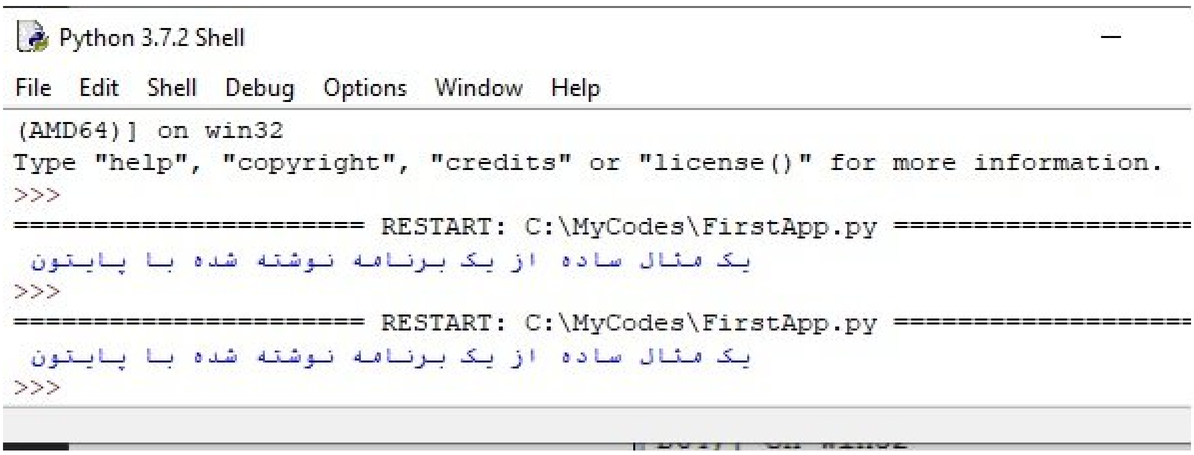
در این پنجره ابتدا باید مسیر ذخیره برنامه را مشخص کنید. بهتر است برای هر برنامه یک پوشه جداگانه ایجاد کنید. به عنوان مثال پوشه ای به نام MyCodes در درایو C ایجاد کنید و برنامه را در آن ذخیره کنید. در قسمت File name نام برنامه را وارد کنید. مثلا FirstApp.py

در قسمت Save as type نوع فایل را مشخص می کنید که پیش فرض آن Python files است.

[illegible]

[illegible] Run [illegible] Run Module [illegible] F5 [illegible] IDLE [illegible]

[illegible] ================================== RESTART ================================== [illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]:

[illegible]: @Python_0to100
([illegible])

[illegible]: @learnpy

[illegible]:
[illegible]
[illegible]:
[illegible]
[illegible]:
08:10 - 04/11/1397
[illegible]:
[illegible] - python [illegible] - python [illegible] - python [illegible] - [illegible]

[illegible]
[illegible]: https://www.shabakeh-mag.com/workshop/14481/%D8%A2%D9%85%D9%88%D8%B2%D8%B4-%D8%B1%D8%A7%DB%8C%DA%AF%D8%A7%D9%86-%D9%BE%D8%A7%DB%8C%D8%AA%D9%88%D9%86-python-%E2%80%93-%D9%86%D8%AD%D9%88%D9%87-%D8%A7%D8%B3%D8%AA%D9%81%D8%A7%D8%AF%D9%87-%D8%A7%D8%B2-%D8%B1%D8%A7%D9%87%D9%86%D9%85%D8%A7%DB%8C-%D9%BE%D8%A7%DB%8C%D8%AA%D9%88%D9%86