
rcii

Recap

```
print("an example input")
```
موجود

```
type("an example input")
```
موجود

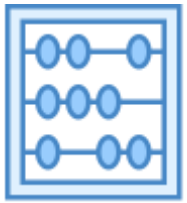
```
print("an example input")
```
اسم تابع

ورودی

Recap

Variable Type	Description	Example Assignment
`int` (Integer)	Whole numbers, positive or negative, without decimals	`x = 10`
`float` (Floating Point)	Numbers with decimal points	`y = 3.14`
`str` (String)	A sequence of characters	`name = "Ali"`
`bool` (Boolean)	Logical value: `True` or `False`	`is_active = True`

Recap



عملگرهای حسابی

Arithmetic Operators

Operators	Meaning	Example	Result
+	Addition	4 + 2	6
–	Subtraction	4 – 2	2
*	Multiplication	4 * 2	8
/	Division	4 / 2	2
%	Modulus operator to get remainder in integer division	5 % 2	1
**	Exponent	5**2 = 5^2	25
//	Integer Division/ Floor Division	5//2 -5//2	2 -3

```
print("salam")
print(3 + 0.14)
```

output

```
salam
3.14
```

What about input ?

print("an example input")

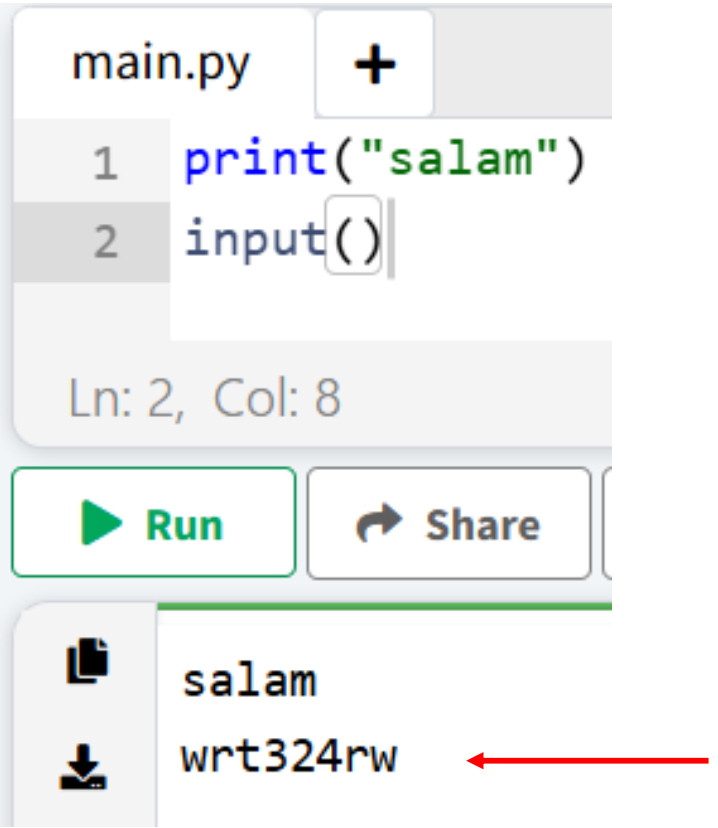
اسم تابع

ورودی

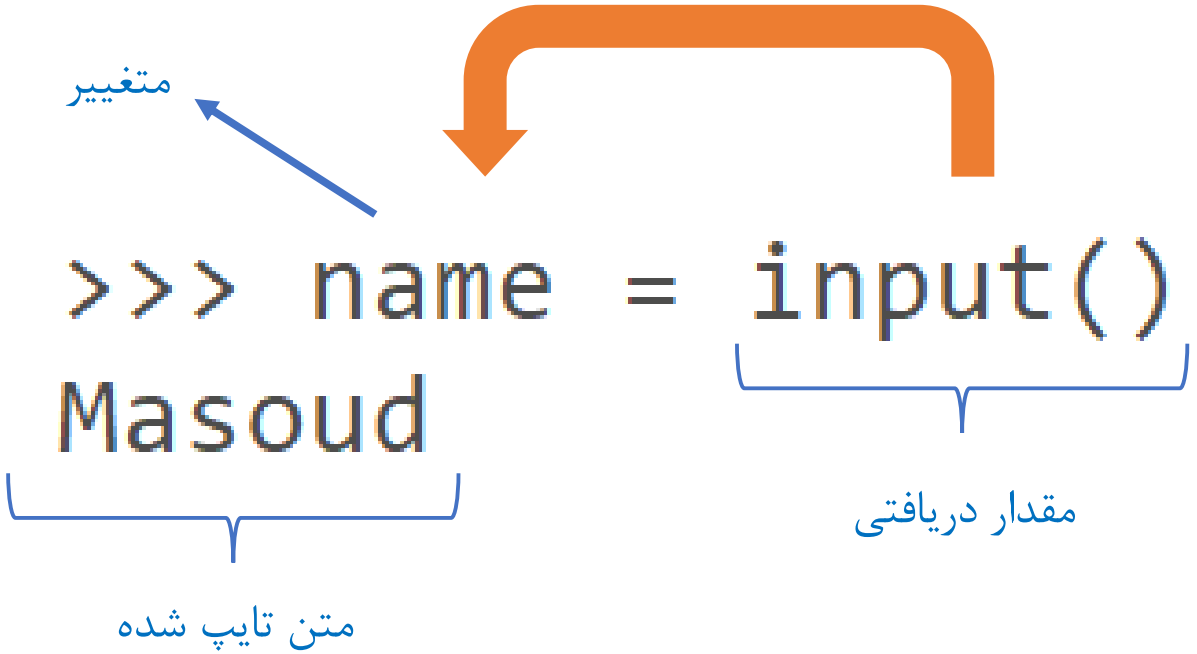
input()

بدون ورودی

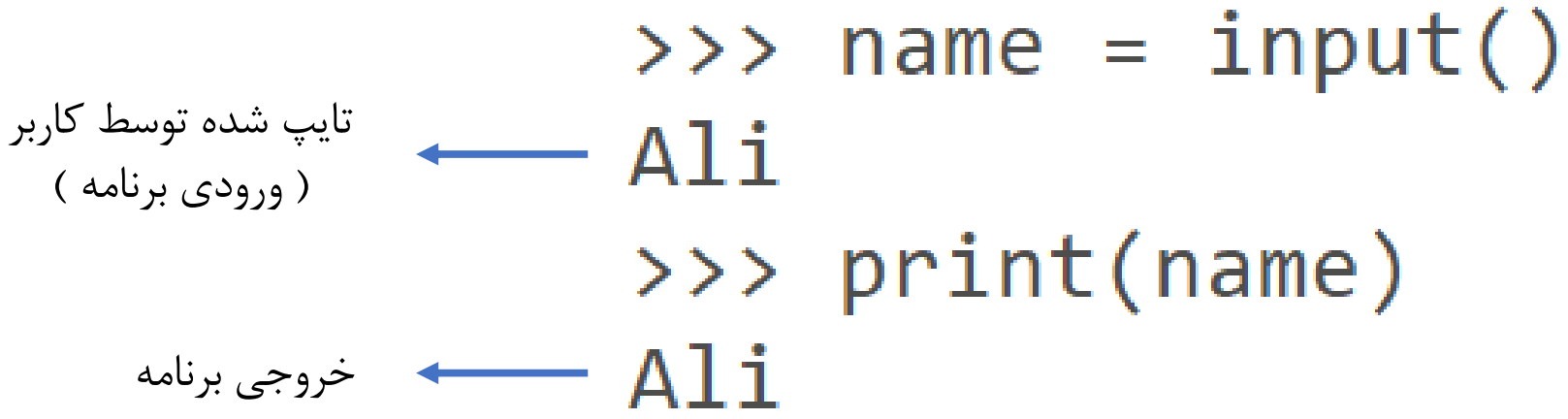
```
>>> input()
my text :)
'my text :)'
```



نشان دادن سر کلاس



print(name)


>>> name = input()
Ali
>>> print(name)
Ali
تایپ شده توسط کاربر
(ورودی برنامه)
خروجی برنامه

print("an example input")

اسم تابع

ورودی

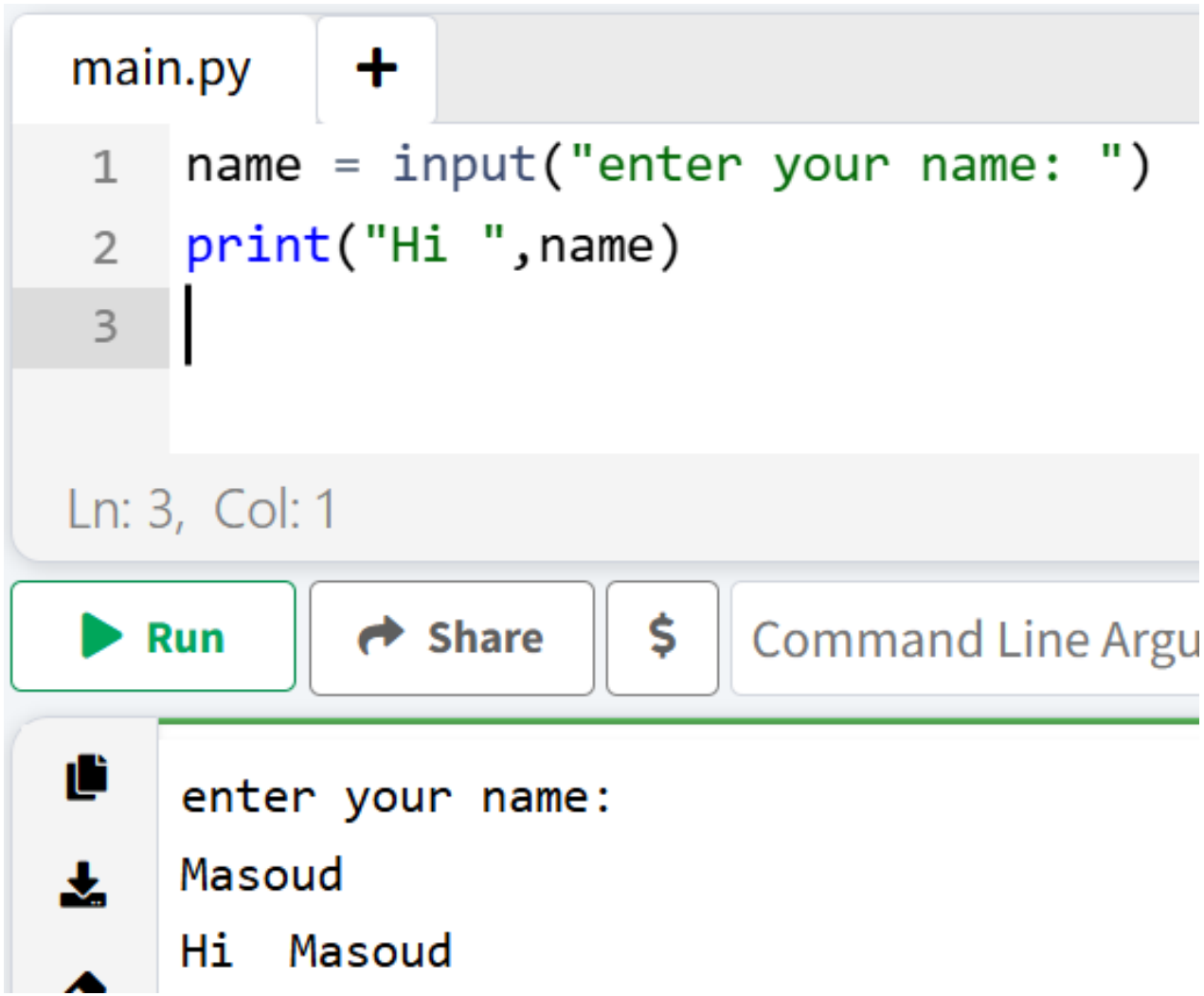
input(“enter your name: ”)

input(“enter your name: ”)

```
>>> name = input("enter your name: ")
enter your name: Masoud
>>> print(name)
Masoud
```

نشان دادن سر کلاس

```
name = input("enter your name: ")
print("Hi ",name)
```


main.py
1 name = input("enter your name: ")
2 print("Hi ",name)
3
Ln: 3, Col: 1
Run
Share
$
Command Line Argu
enter your name:
Masoud
Hi Masoud

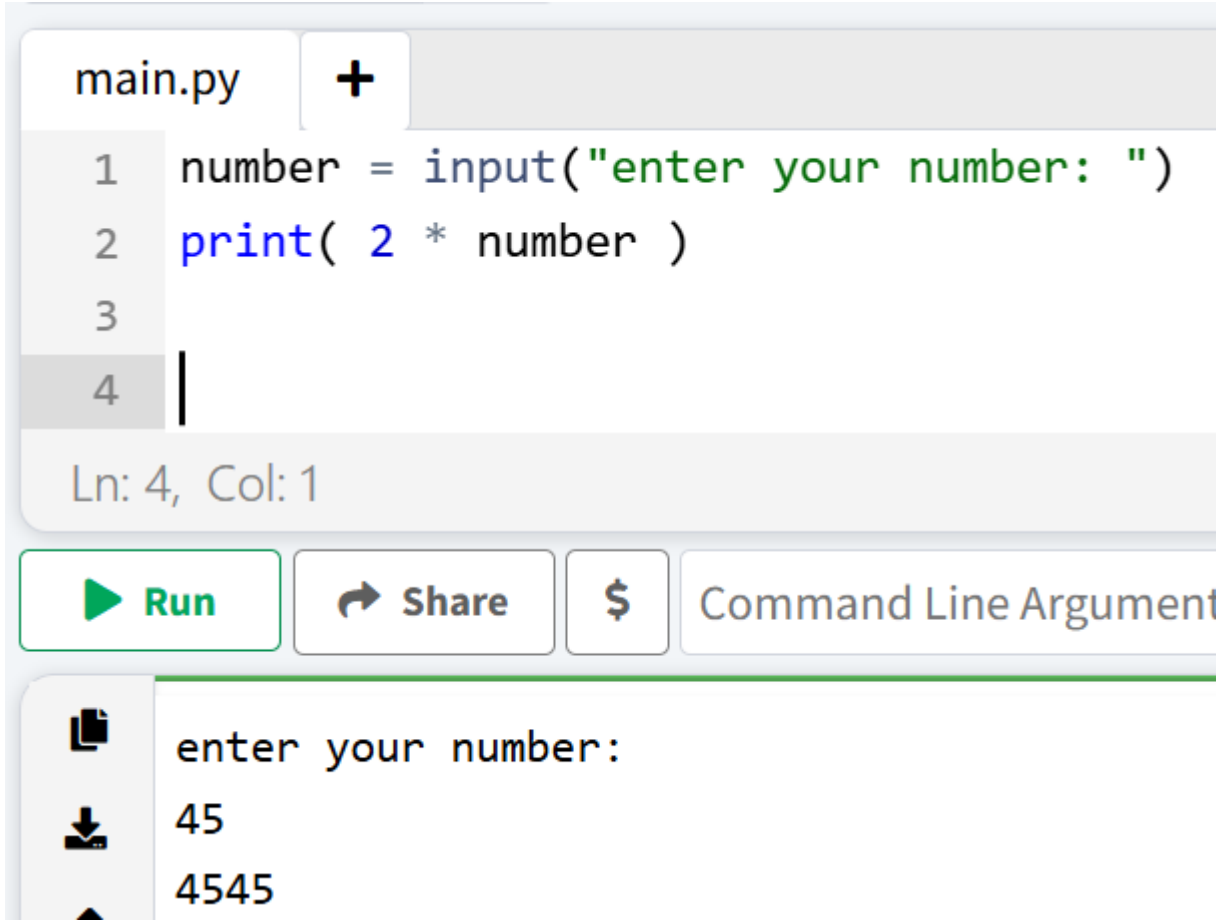
```
>>> name = input("enter your name: ")
enter your name: Masoud
>>> type(name)
```

```
>>> name = input("enter your name: ")
enter your name: Masoud
>>> type(name)
<class 'str'>
```

برنامه ای بنویسید که در آن یک عدد از کاربر گرفته شود و دو برابر آن عدد به کاربر نمایش داده شود

```
number = input("enter your number: ")
print( 2 * number )
```

suggestion


main.py
1 number = input("enter your number: ")
2 print(2 * number)
3
4
Ln: 4, Col: 1
Run
Share
$
Command Line Argument
enter your number:
45
4545

```
>>> 2 * "He"
'HeHe'
>>> 2 * "He "
'He He '
```

main.py +

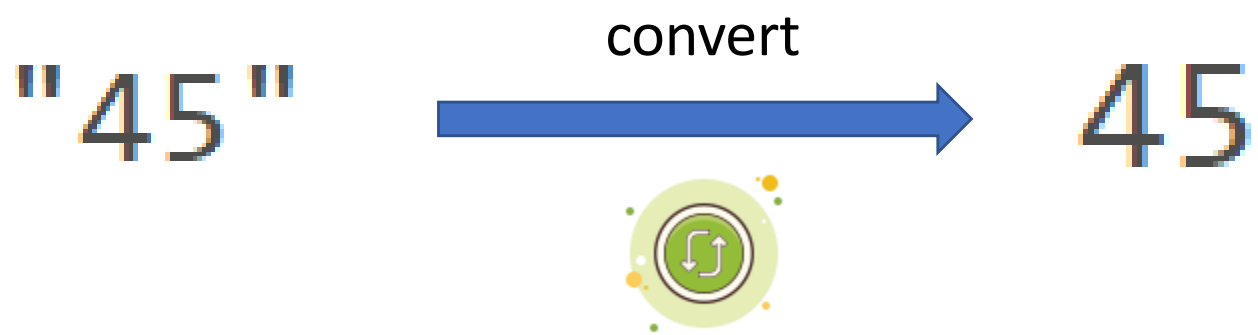
```
1  number = input("enter your number: ")
2  print( 2 * number )
3
4
```

Ln: 4, Col: 1

Run | Share | $ | Command Line Argument

```
enter your number:
45
4545
```

```
>>> 2 * 45
90
>>> 2 * "45"
'4545'
```


"45"
convert
45

int(“45”)

int(“45”)

```
>>> "45"
'45'
>>> int("45")
45
```

str →

int →

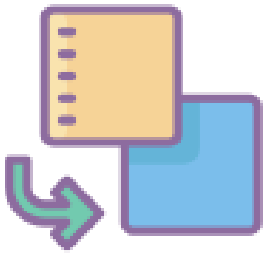
Type Conversion

```
>>> number = "45"
>>> type(number)
<class 'str'>
>>> number = int(number)
>>> type(number)
<class 'int'>
```

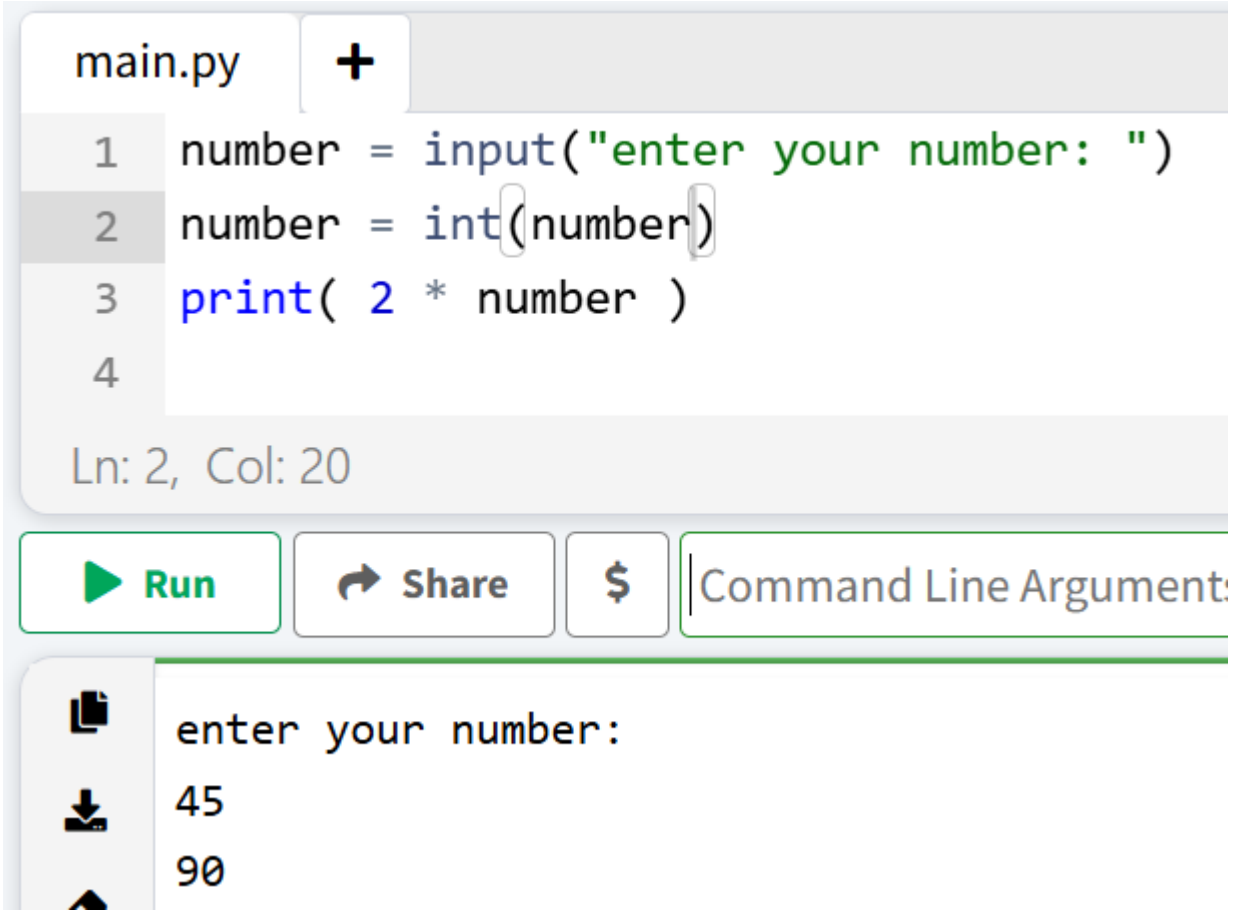
برنامه ای بنویسید که در آن یک عدد از کاربر گرفته شود و دو برابر آن عدد به کاربر نمایش داده شود

```
number = input("enter your number: ")
print( 2 * number )
```

What change ?



برنامه ای بنویسید که در آن یک عدد از کاربر گرفته شود و دو برابر آن عدد به کاربر نمایش داده شود

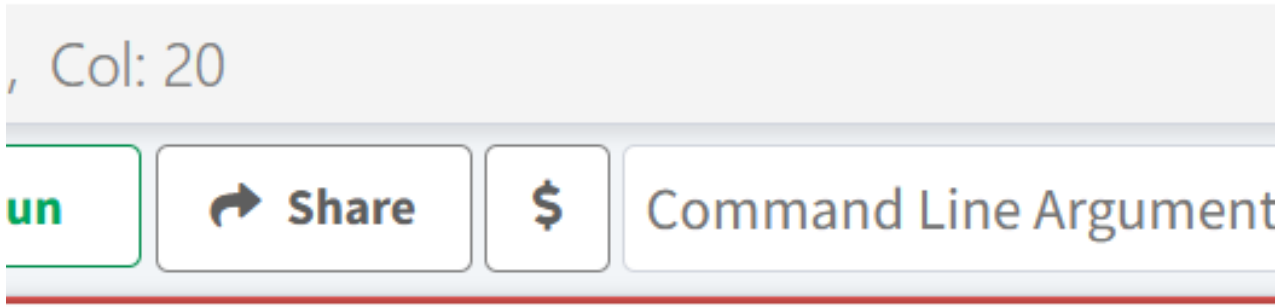


```
>>> number = "65"
>>> int(number)
65
>>> number = "salam"
>>> int(number)
```

```
>>> number = "salam"
>>> int(number)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: invalid literal for int() with base
10: 'salam'
```

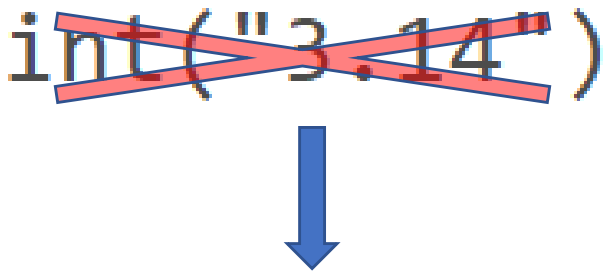
نشان دادن سر کلاس

```
number = input("enter your number: ")
number = int(number)
print( 2 * number )
```



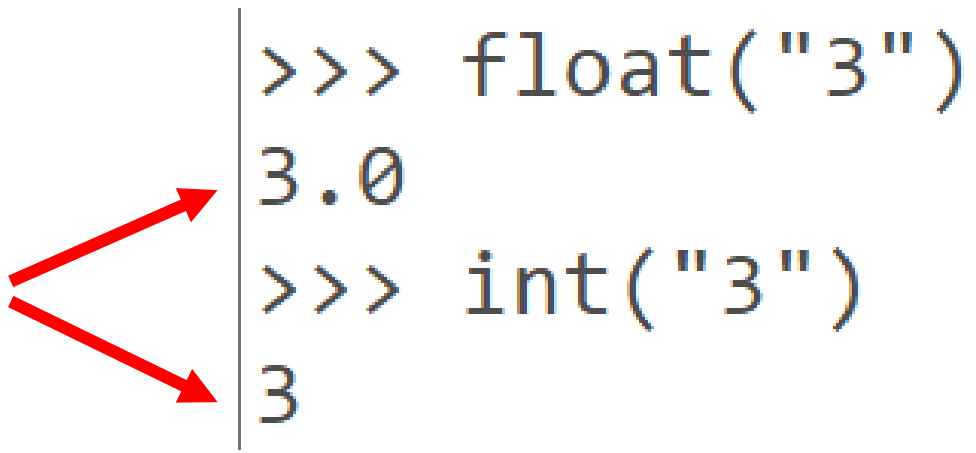

```
enter your number:
3.14
```

```
>>> int("3.14")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ValueError: invalid literal for int() with base
10: '3.14'
```

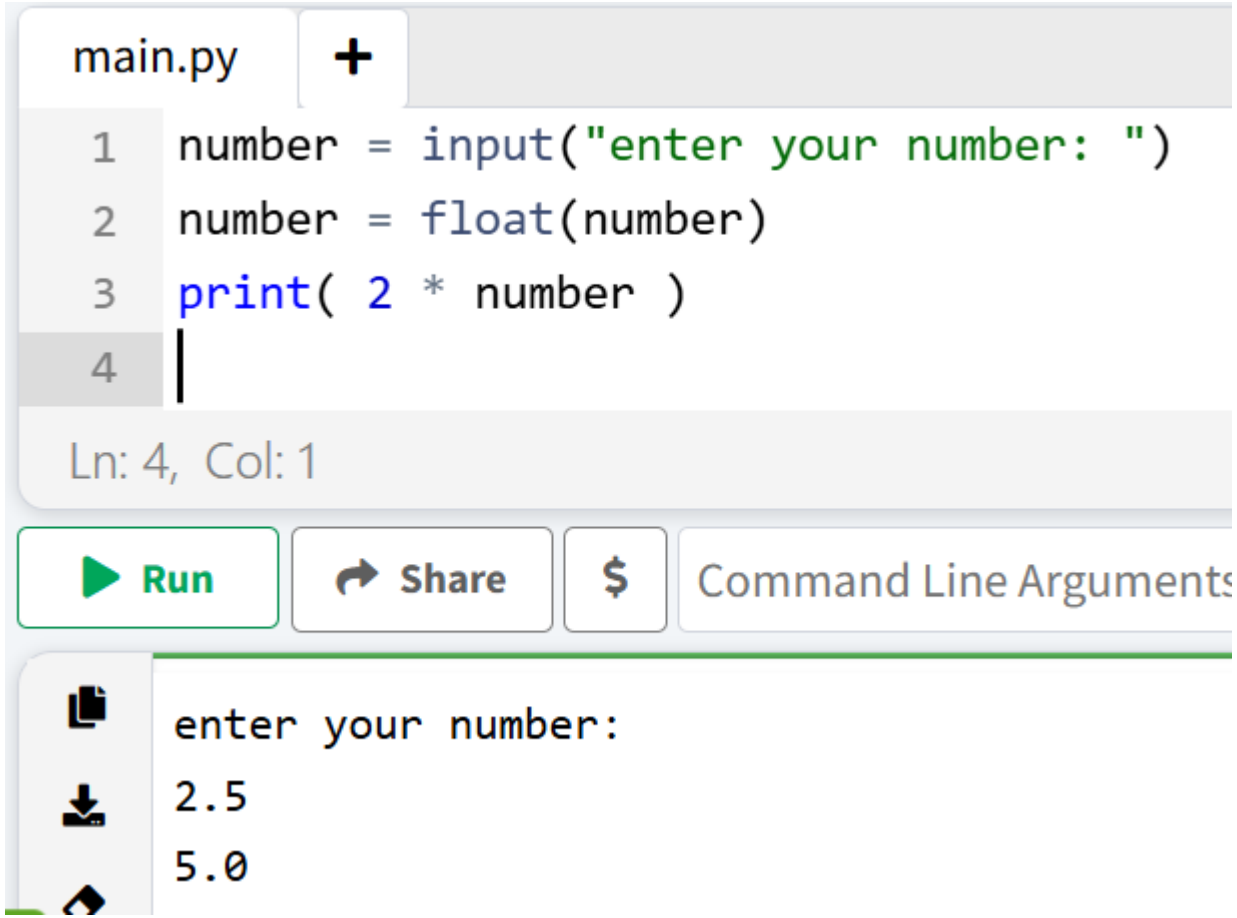



```
>>> float("3.14")
3.14
```

```
float("3")
```


>>> float("3")
3.0
>>> int("3")
3

برنامه ای بنویسید که در آن یک عدد از کاربر گرفته شود و دو برابر آن عدد به کاربر نمایش داده شود



```
>>> int(3.14)
```

```
>>> int(3.14)
3
```

```
>>> int(3.81)
```

```
>>> int(3.81)
3
```

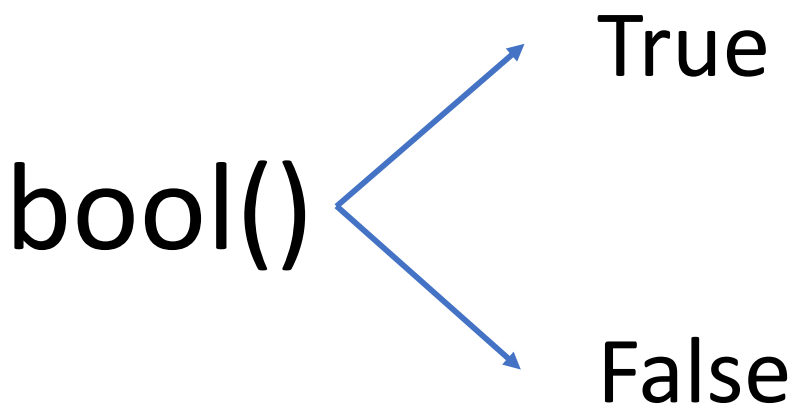
```
>>> "Masoud" + "Khodaverdian"
'MasoudKhodaverdian'
>>> "Masoud" + " " + "Khodaverdian"
'Masoud Khodaverdian'
```

```
>>> age = 25
>>> "Masoud age is " + age
```

```
>>> age = 25
>>> "Masoud age is " + age
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int")
to str
```

```
>>> str(19)
'19'
>>> str(3.14)
'3.14'
```

- int()
- float()
- str()
- ?


bool()
True
False

```
>>> bool(1)
```

```
>>> bool(1)
True
```

```
>>> bool(0)
False
```

```python
print(bool(0))     # False
print(bool(1))     # True
print(bool(-5))    # True
print(bool(3.14)) # True
print(bool(0.0))  # False
```

```
>>> bool("dhgerth")
True
>>> bool("")
False
```

```
bool("0")
```

```
>>> bool("0")
True
```

```
>>> bool(" ")
```

```
>>> bool(" ")
True
```