In [ ]:

conversion to int

In [3]:
```python
int (2.3, 4.5)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[3], line 1
----> 1 int (2.3, 4.5)

TypeError: 'float' object cannot be interpreted as an integer
```

In [5]:
```python
int(True)
```

Out[5]: 1

In [6]:
```python
int(False)
```

Out[6]: 0

In [7]:
```python
int (True, False)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[7], line 1
----> 1 int (True, False)

TypeError: int() can't convert non-string with explicit base
```

In [8]:
```python
int (true)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[8], line 1
----> 1 int (true)

NameError: name 'true' is not defined
```

In [ ]:
```python
#Data types convert into int
```

In [10]:
```python
print (int(2.3))
print (int(True))
print (int('10'))
print (int(1+2j))
```

```
2
1
10
```

```
---------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[10], line 4
      2 print (int(True))
      3 print (int('10'))
----> 4 print (int(1+2j))

TypeError: int() argument must be a string, a bytes-like object or a real number, no
t 'complex'
```

In [12]: `int('1')`

Out[12]: 1

In [13]: `#convertion to float`

In [14]:
```
print (float(2))
print (float(False))
print (float(2+3j))
```

```
2.0
0.0
---------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[14], line 3
      1 print (float(2))
      2 print (float(False))
----> 3 print (float(2+3j))

TypeError: float() argument must be a string or a real number, not 'complex'
```

In [ ]:

In [ ]: `#convertion to complex`

In [17]: `complex(10)`

Out[17]: (10+0j)

In [18]: `complex(10,20) #max 2 arguments can be written in complex conversion`

Out[18]: (10+20j)

In [19]: `complex(10,20,30)`

```
---------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[19], line 1
----> 1 complex(10,20,30)

TypeError: complex() takes at most 2 arguments (3 given)
```

In [20]: `complex (2.3 , 5.6)`

Out[20]: (2.3+5.6j)

In [21]:
```python
complex(2.3, 5)
```

Out[21]: (2.3+5j)

In [22]:
```python
complex (5 + 2.3)
```

Out[22]: (7.3+0j)

In [23]:
```python
complex (False, False)
```

Out[23]: 0j

In [24]:
```python
complex ('1',23) #complex cannot take one number if another is string
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[24], line 1
----> 1 complex ('1',23)

TypeError: complex() can't take second arg if first is a string
```

In [25]:
```python
complex ('1', '23') #complex cannot take two strings
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[25], line 1
----> 1 complex ('1', '23')

TypeError: complex() can't take second arg if first is a string
```

In [26]:
```python
complex ('1') #complex can take one string
```

Out[26]: (1+0j)

In [ ]:

In [ ]:
```python
#convertion to boolean
```

In [28]:
```python
print (bool (1))
print (bool (1.1))
print (bool (1+2j))
print (bool ('nit'))
```

```
True
True
True
True
```

In [31]:
```python
bool ()
```

Out[31]: False

In [39]:
```python
print (str(2))
print (str(2.2))
print (str(2+3j))
print (str(True))
```

```
2
2.2
(2+3j)
True
```

In [ ]: