Typecasting

(int)-int/float/bool/str/complex

In [1]:
```
x=int(10)
print(int(10))
```

```
10
```

In [2]:
```
x=int(10)
print(float(x))
```

```
10.0
```

In [3]:
```
x=int(10)
print(bool(x))
```

```
True
```

In [7]:
```
x=int(10)
print(str(x))
```

```
10
```

In [5]:
```
x=int(10)
print(complex(x))
```

```
(10+0j)
```

(float)=int/float/bool/str/complex

In [8]:
```
y=float(2.3)
print(int(y))
```

```
2
```

In [13]:
```
y=float(2.3)
print(float(y))
```

```
2.3
```

In [14]:
```
y=float(3.6)
print(bool(y))
```

```
True
```

In [20]:
```
y=float(3.6)
print(str(y))
```

```
3.6
```

In [16]:
```
y=float(3.6)
print(complex(y))
```

```
(3.6+0j)
```

(bool)-int/float/bool/str/complex

In [19]:
```
z=bool(True)
print(int(z))
```

```
1
```

In [22]:
```
z=bool(False)
print(float(z))
```

```
0.0
```

In [23]:
```
z=bool(False)
print(bool(z))
```

```
False
```

In [24]:
```
z=bool(False)
print(str(z))
```

```
False
```

In [25]:
```
z=bool(False)
print(complex(z))
```

```
0j
```

(str)-int/float/bool/str/complex

In [29]:
```
x=str('john')
print(int(x))
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[29], line 2
      1 x=str('john')
----> 2 print(int(x))

ValueError: invalid literal for int() with base 10: 'john'
```

In [30]:
```
x=str('john')
print(float(x))
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[30], line 2
      1 x=str('john')
----> 2 print(float(x))

ValueError: could not convert string to float: 'john'
```

In [32]:
```
x=str('john')
print(bool(x))
```

```
True
```

In [33]:
```
x=str('john')
print(complex(x))
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Cell In[33], line 2
      1 x=str('john')
----> 2 print(complex(x))

ValueError: complex() arg is a malformed string
```

(complex)-int/float/bool/str/complex

In [35]:
```
y=complex(1+2j)
print(int(y)) #TypeError: int() argument must be a string, a bytes-like object or a
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[35], line 2
      1 y=complex(1+2j)
----> 2 print(int(y))

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

In [45]:
```
y=complex(1+2j)  #TypeError: float() argument must be a string or a real number, no
print(float(y))
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[45], line 2
      1 y=complex(1+2j) #
----> 2 print(float(y))

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

In [39]:
```
y=complex(1+2j)
print(bool(y))
```

```
True
```

In [41]:
```
y=complex(1+2j)
print(str(y))
```

```
(1+2j)
```

In [42]:
```
y=complex(1+2j)
print(complex(y))
```

```
(1+2j)
```

In [43]:
```
y.real
```

Out[43]:
```
1.0
```

In [44]:
```
type(y)
```

Out[44]:
```
complex
```

In [ ]: