

Lesson 7:

TYPE CHECKING

```
>>> name = "humna"
>>> print(name)
humna
>>> name=b"humna"
>>> print(name)
b'humna'
>>> name = "humna"
>>> bin_name = bin(name)
Traceback (most recent call last):
  File "<pyshell#5>", line 1, in <module>
    bin_name = bin(name)
TypeError: 'str' object cannot be interpreted as an integer
>>> bin_name = byte(name)
Traceback (most recent call last):
  File "<pyshell#6>", line 1, in <module>
    bin_name = byte(name)
NameError: name 'byte' is not defined. Did you mean: 'bytes'?
>>> bin_name = bytes(name)
Traceback (most recent call last):
  File "<pyshell#7>", line 1, in <module>
    bin_name = bytes(name)
TypeError: string argument without an encoding
>>> bin_name = name.encode('utf-8')
>>> print(bin_name)
b'humna'
>>> print(type(bin_name))
<class 'bytes'>
>>> a = 7.98
>>> print(type(a))
<class 'float'>
>>> c = 6 + 7j
>>> print(type(c))
<class 'complex'>
>>> comp_sbhdchj= 6 +7j
>>> print(type(comp_sbhdchj))
<class 'complex'>
>>> |
```

DATATYPE CONVERSION:

```
>>> int_data = int(st_data)
Traceback (most recent call last):
  File "<pyshell#22>", line 1, in <module>
    int_data = int(st_data)
ValueError: invalid literal for int() with base 10: 'hello'
>>> st_data = "123"
>>> int_data = int(st_data)
>>> print(int_data)
123
>>> print(st_data)
123
>>> print(type(int_data))
<class 'int'>
>>> print(type(st_data))
<class 'str'>
```

OPERATORS

1. Arithmetic Operator

```
#Practice Arithmetic operators
```

```
a = 10
b = 6
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(a%b)
```

```
""" output
16
4
60
1.6666666666666667
4
"""
```

2. Assignment Operator

```
#practice Assignment operators

a = 10

"""
print(a += 5)
print(a -= 5)
print(a *= 5)
print(a /= 5)
print(a %= 5)

these print will give invalid syntax error
because these are statements written in print

"""
```

```
#correct
a = 10
a+=5    # a = a+5
print(a) # 15

a= 10
a -= 5    # a = a-5    ---> 5
print(a)

a/=5      # a= 5 --> a = a /5 --->1
print(a)

a%=5      # a=1 ---> a= a%5 ----> a=1
print(a)

a =10
a*=5      # a = a *5 ---> a =50
print(a)
```