The bool Data Type

- In Python , to represent **Boolean** values we have **bool data type.**

1 True 0 false

- The **bool data type** can be one of two values, either **True** or **False**.

- We use Booleans in programming to make comparisons and to control the flow of the program.

Some Examples

```
>>> a=False
>>> print(a)
False
```

```
>>> a=False
>>> type(a)
<class 'bool'>
```

Some Important Points About bool

True and **False** are **keywords** , so case sensitivity must be remembered while assigning them otherwise **Python** will give error

```
>>> a=false
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'false' is not defined
```

Some Important Points About bool

- All test conditions in **Python** return the result as **bool** which could be either **True** or **False**

```
>>> a=10
>>> b=5
>>> print(a>b)
True
```

```
>>> x=15
>>> y=15
>>> print(x<y)
False
```

Some Important Points About bool

To understand the next point , try to guess the output of the following:

```
a=True
b=True
c=a+b
print(c)
```

```
a=False
b=False
c=a+b
print(c)
```

```
a=True
b=False
c=a+b
print(c)
```

True → 1

False → 0

The above outputs make it clear that internally **Python** stores **True** and **False** as **integers** with the value **1** and **0** respectively