Basic data types in python

- int
- str
- bool (True, False)
- float
- None

Type casting

convering object from one data type into another

- int()
- float()
- str()
- bool()

Survival functions

- type()
- help()
- dir()

In [31]:

```python
foo = 1000
type(foo) # `foo` is a variable referring to `int` value
```

Out[31]:

```
int
```

In [3]:

```python
foo = "prashant"
type(foo)  # `foo` is variable referring to `str` value
```

Out[3]:

```
str
```

In [5]:

```python
# how to convert integer into string?

foo = 600
result = str(foo)
type(result)
```

Out[5]:

```
str
```

In [13]:

```python
# how do you convert float value into int?

bar = 60.50
result = int(bar)
result
```

Out[13]:

```
60
```

In [12]:

```python
# how to convert a `string` representing numerical float into `float`?
foo = "40.50"
float(foo)
```

Out[12]:

```
40.5
```

In [35]:

```python
# can we convert a `string` representing numerical float into `int`?

foo = "40.9"      # step 1
print(foo)

foo = float(foo)   # step 2
print(foo)

result = int(foo)    # step 3
print(result)
```

```
prashant40.9

---------------------------------------------------------------
-----
ValueError                                Traceback (most recent call
 last)
Cell In [35], line 6
      3 foo = "prashant40.9"      # step 1
      4 print(foo)
----> 6 foo = float(foo)   # step 2
      7 print(foo)
      9 result = int(foo)    # step 3

ValueError: could not convert string to float: 'prashant40.9'
```

In [18]:

```python
help("keywords")      # False, True, None
```

Here is a list of the Python keywords. Enter any keyword to get more
help.

```
False               class               from                or
None                continue            global              pass
True                def                 if                  raise
and                 del                 import              return
as                  elif                in                  try
assert              else                is                  while
async               except              lambda              with
await               finally             nonlocal            yield
break               for                 not
```

Boolean value - type casting

Rules

- Any non-empty string is considered as `True`
- Any non-zero integer is considered as `True`

In [39]:

```python
# examples of empty strings

foo = ''
foo = bool(foo)
print(foo)
foo = ""
foo = bool(foo)
print(foo)
foo = ''''''
foo = bool(foo)
print(foo)
foo = """"""
foo = bool(foo)
print(foo)
```

```
False
False
False
False
```

In [23]:

```python
foo = True
type(foo)
```

Out[23]:

```
bool
```

In [24]:

```python
bar = False
type(bar)
```

Out[24]:

```
bool
```

In [25]:

```python
# how to convert string into boolean?

country = "India"
result = bool(country)
result
```

Out[25]:

```
True
```

In [26]:

```python
state = ""
bool(state)   # empty strings are False
```

Out[26]:

```
False
```

In [27]:

```python
# how to convert `int` into `bool`?
foo = 10
bool(foo)
```

Out[27]:

```
True
```

In [28]:

```python
bar = 0
bool(bar)
```

Out[28]:

```
False
```

In [32]:

```python
# converting float into bool
random = 40.50
bool(random)
```

Out[32]:

```
True
```

In [33]:

```python
random = 0.0
bool(random)
```

Out[33]:

False

In [33]:

```python
random = 0.0
bool(random)
```

Out[33]:

False