Data types in Python

- Integers
- Float
- Strings
- Bool
- None

Check Data type

Integers

- You can use an integer represent numeric data, and more specifically, whole numbers from negative infinity to infinity, like 4, 5, or -1.

```
print(2345874519641982)
```

```
2345874519641982
```

```
# type tells the data types of object
```

```
print(type(123))
```

```
<class 'int'>
```

```
print(type(-12414234))
```

```
<class 'int'>
```

```
type(34243)
```

```
int
```

Type function

```
# tells the data type of an object
```

Floats

- "Float" stands for 'floating point number'. You can use it for rational numbers, usually ending with a decimal figure, such as 1.11 or 3.14.

```
type(234.0)
```

```
float
```

```
type(-0.0)
```

```
float
```

Strings

- Strings are collections of alphabets, words or other characters. In Python, you can create strings by enclosing a sequence of characters within a pair of single or double quotes. For example: 'cake', "cookie", etc.

```
type("Rahul")
```

```
str
```

```
type("2fwtfe432@$%")
```

```
str
```

Boolean

- This built-in data type that can take up the values: True and False, which often makes them interchangeable with the integers 1 and 0. Booleans are useful in conditional and comparison expressions

```
# True, False
```

```
# Quiz
```

```
type(True)
```

```
bool
```

```
type(False)
```

```
bool
```

```
## None
```

```
type(None)
```

```
NoneType
```

```
# quizzes
```

```
print(20+30)
```

```
50
```

```
print(8-5)
```

```
3
```

```
print("10+10")
```

```
10+10
```

```
print(type(true))
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/var/folders/zn/hkv6562d6_d30glfs8yc76900000gn/T/ipykernel_10602/61652065.py in <module>
----> 1 print(type(true))

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

```
print(type(True))
```

```
<class 'bool'>
```

Variables

It's a type of variable's Rules for naming variables

- Name must start from Alphabet(small or caps) or underscore(_)
- They are case sensitive. It can differentiate between small and CAPS.
- Name should not start with number
- No special characters

```
x = 5
y = 3
z = x + y
print(z)
```

```
8
```

```
print(x)
```

```
5
```

```
X = 45
print(x)
```

```
5
```

```
print(X)
```

```
45
```

```
1x = 45
```

```
  File
"/var/folders/zn/hkv6562d6_d30glfs8yc76900000gn/T/ipykernel_10602/1719
844030.py", line 1
    1x = 45
     ^
SyntaxError: invalid syntax
```

```
x1 = 6
print(x1)
```

```
6
```

```
# Python is case sensitive
```

```
name='Rahul'
print(Name)
```

```
---------------------------------------------------------------------
-----
NameError                                 Traceback (most recent call
last)
```

```
/var/folders/zn/hkv6562d6_d30glfs8yc76900000gn/T/ipykernel_10602/14667
83110.py in <module>
      1 name='Rahul'
----> 2 print(Name)

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

```
print(None)
```

```
None
```

```
# Variable holds latest value
```

```
# Quiz
```

```
x = 5
x = 25
x = 45
print(x)
```

```
45
```

```
x = 3
y = 'hello'
num = y
y = 5
print(num, y)
```

```
hello 5
```

```
heLLo = 3
```

```
rahul1 = 3
```

```
1rahul = 3
```

```
  File
"/var/folders/zn/hkv6562d6_d30glfs8yc76900000gn/T/ipykernel_10602/1569
275968.py", line 1
    1rahul = 3
     ^
SyntaxError: invalid syntax
```

```
rahul_janghu = 3
```

Input function

- It takes input from user i.e us
- Typecasting of input function

```
product = input()
```

```
 iphone
```

```
print(product)
```

```
iphone
```

```
type(product)
```

```
str
```

Challenge: Take two user input and add them

```
x = input()
y = input()
```

```
 4
 5
```

```
print(x, y)
```

```
4 5
```

```
type(x)
```

```
str
```

```
type(y)
```

```
str
```

```
# Default type of input function is string
```

```
print(x + y)
```

```
45
```

```
# String concatenation
```

```
print("1" + "1")
```

```
11
```

```
a = input()
```

```
 23.4
```

```
type(a)
```

```
str
```

```
# Typecasting

# The batman costume

x
```

```
'4'
```

```
x = int(x)

type(x)
```

```
int
```

```
# int("Rahul")


# final code

x = input()
y = input()

# convert into int
x = int(x)
y = int(y)

add = x + y
print(add)
```

```
 4
 3
```

```
7
```

```
x = int(input())
y = int(input())

add = x + y
print(add)
```

```
 23
 45

68
```

```
print("Hello", "world")
```

```
Hello world
```

```
print("Hello")
print("World")
```

```
Hello
World
```

```
print("Hello world!")
```

```
Hello world!
```

```
print("Hello")
print()
print("World")
```

```
Hello

World
```