1. print

In [1]:
```
print("Hello World")
```

```
Hello World
```

2. input

In [2]:
```
input("Enter your name: ")
```

```
Enter your name: Ahmed Ali
```

Out[2]:
```
'Ahmed Ali'
```

3. type ¶

In [3]:
```
a = 23
type(a)
```

Out[3]:
```
int
```

In [4]:
```
a = 1.2
type(a)
```

Out[4]:
```
float
```

In [5]:
```
a = True
type(a)
```

Out[5]:
```
bool
```

4. int etc.

In [2]:
```
# int('5')
# float("5")
# str(5)
# list((5,4,6))
# tuple((3,4))
```

Out[2]: 5

5. abs

In [6]:
```
abs(4)
```

Out[6]: 4

In [7]:
```
abs(-23.42)
```

Out[7]: 23.42

6. pow

In [13]:
```
pow(3, 3)
```

Out[13]: 27

In [16]:
```
pow(2, 3)
```

Out[16]: 8

7. min/max

In [8]:
```
min([4,3,2,1,0])
```

Out[8]:
```
0
```

In [9]:
```
max([2, 1, 3, 4, 0])
```

Out[9]:
```
4
```

In [13]:
```
min("kolkata")
```

Out[13]:
```
'a'
```

In [14]:
```
max("kolkata") # based on ascii value
```

Out[14]:
```
't'
```

8. round

In [11]:
```
c = 22/7
print(c)
round(c)
```

```
3.142857142857143
```

Out[11]:
```
3
```

In [14]:
```
round(c,6)
```

Out[14]:
```
3.142857
```

9. divmod

In [22]:
```
divmod(5, 2)
```

Out[22]:
```
(2, 1)
```

In [16]:
```
divmod(55,10)
```

Out[16]:
```
(5, 5)
```

10. bin/oct/hex

In [17]:
```
bin(12)
```

Out[17]:
```
'0b1100'
```

In [18]:
```
oct(12)
```

Out[18]:
```
'0o14'
```

In [19]:
```
hex(12)
```

Out[19]:
```
'0xc'
```

11. id

In [20]:
```
a = 3
id(a)
```

Out[20]:
```
2806840361264
```

12. ord

In [26]:
```
ord('A')
```

Out[26]: 65

In [25]:
```
ord('a')
```

Out[25]: 97

13. len

In [28]:
```
len('Mumbai')
```

Out[28]: 6

In [47]:
```
len([1, 2, 3])
```

Out[47]: 3

14. sum

In [56]:
```
sum([1, 2, 3, 4, 5])
```

Out[56]: 15

In [57]:
```
sum({1, 2, 3, 4, 5})
```

Out[57]: 15

In [58]:
```
sum([2,3,4,5,6])
```

Out[58]: 20

15. help

In [45]:
```
help('print')
```

```
Help on built-in function print in module builtins:

print(...)
    print(value, ..., sep=' ', end='\n', file=sys.stdout, flush=False)

    Prints the values to a stream, or to sys.stdout by default.
    Optional keyword arguments:
    file:  a file-like object (stream); defaults to the current sys.stdout.
    sep:   string inserted between values, default a space.
    end:   string appended after the last value, default a newline.
    flush: whether to forcibly flush the stream.
```

In [46]:
```
help('sum')
```

```
Help on built-in function sum in module builtins:

sum(iterable, /, start=0)
    Return the sum of a 'start' value (default: 0) plus an iterable of numbers

    When the iterable is empty, return the start value.
    This function is intended specifically for use with numeric values and may
    reject non-numeric types.
```

In [47]:
```
help('len')
```

```
Help on built-in function len in module builtins:

len(obj, /)
    Return the number of items in a container.
```

In [60]:
```
help('ord')
```

```
Help on built-in function ord in module builtins:

ord(c, /)
    Return the Unicode code point for a one-character string.
```

In [ ]:
```

```