Operators in Python

- Arithmetic Operators
- Relational Operators
- Logical Operators
- Bitwise Operators
- Assignment Operators
- Membership Operators

```
# Arithmetic Operators
print(5+6)

print(5-6)

print(5*6)

print(5/2)

print(5//2)

print(5%2)

print(5**2)
```

```
11
-1
30
2.5
2
1
25
```

```
# Relational Operators
print(4>5)

print(4<5)

print(4>=4)

print(4<=4)

print(4==4)

print(4!=4)
```

```
False
True
True
True
True
False
```

```
# Logical Operators
print(1 and 0)

print(1 or 0)

print(not 1)
```

```
0
1
False
```

```
# Bitwise Operators

# bitwise and
print(2 & 3)

# bitwise or
print(2 | 3)

# bitwise xor
print(2 ^ 3)

print(~3)

print(4 >> 2)

print(5 << 2)
```

```
2
3
1
-4
1
20
```

```
# Assignment Operators

# =
# a = 2

a = 2

# a = a % 2
a %= 2

# a++ ++a

print(a)
```

```
4
```

```
# Membership Operators

# in/not in

print('D' not in 'Delhi')

print(1 in [2,3,4,5,6])
```

```
False
False
```

```
# Program - Find the sum of a 3 digit number entered by the user

number = int(input('Enter a 3 digit number'))

# 345%10 -> 5
a = number%10

number = number//10

# 34%10 -> 4
b = number % 10

number = number//10
# 3 % 10 -> 3
c = number % 10

print(a + b + c)
```

```
Enter a 3 digit number666
18
```

If-else in Python

```
# login program and indentation
# email -> nitish.campusx@gmail.com
# password -> 1234

email = input('enter email')
password = input('enter password')

if email == 'nitish.campusx@gmail.com' and password == '1234':
  print('Welcome')
elif email == 'nitish.campusx@gmail.com' and password != '1234':
  # tell the user
  print('Incorrect password')
  password = input('enter password again')
  if password == '1234':
    print('Welcome,finally!')
  else:
    print('beta tumse na ho paayega!')
else:
  print('Not correct')
```

```
    enter emailsrhreh
    enter passworderhetjh
    Not correct
```

```
# if-else examples
# 1. Find the min of 3 given numbers
# 2. Menu Driven Program
```

```
# min of 3 number

a = int(input('first num'))
b = int(input('second num'))
c = int(input('third num'))

if a<b and a<c:
  print('smallest is',a)
elif b<c:
  print('smallest is',b)
else:
  print('smallest is',c)
```

```
    first num4
    second num1
    third num10
    smallest is 1
```

```
# menu driven calculator
menu = input("""
Hi! how can I help you.
1. Enter 1 for pin change
2. Enter 2 for balance check
3. Enter 3 for withdrawl
4. Enter 4 for exit
""")

if menu == '1':
  print('pin change')
elif menu == '2':
  print('balance')
else:
  print('exit')
```

```

    Hi! how can I help you.
    1. Enter 1 for pin change
    2. Enter 2 for balance check
    3. Enter 3 for withdrawl
    4. Enter 4 for exit
    2
    balance
```

Modules in Python

- math
- keywords
- random
- datetime

```
# math
import math

math.sqrt(196)
```

```
14.0
```

```
# keyword
import keyword
print(keyword.kwlist)
```

```
['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except',
```

```
# random
import random
print(random.randint(1,100))
```

```
88
```

```
# datetime
import datetime
print(datetime.datetime.now())
```

```
2022-11-08 15:50:21.228643
```

```
help('modules')
```

```
Please wait a moment while I gather a list of all available modules...

/usr/local/lib/python3.7/dist-packages/caffe2/proto/__init__.py:17: UserWarning: Caffe2 support is not enabled in this PyTorch build.
/usr/local/lib/python3.7/dist-packages/caffe2/proto/__init__.py:17: UserWarning: Caffe2 support is not enabled in this PyTorch build.
/usr/local/lib/python3.7/dist-packages/caffe2/python/__init__.py:9: UserWarning: Caffe2 support is not enabled in this PyTorch build.
Cython              collections         kaggle              requests_oauthlib
IPython             colorcet            kanren              resampy
OpenGL              colorlover          kapre               resource
PIL                 colorsys            keras               rlcompleter
ScreenResolution    community           keras_preprocessing rmagic
__future__          compileall          keyword             rpy2
_abc                concurrent          kiwisolver          rsa
_ast                confection          korean_lunar_calendar runpy
_asyncio            configparser        langcodes           samples
_bisect             cons                lib2to3             sched
_blake2             contextlib          libfuturize         scipy
_bootlocale         contextlib2         libpasteurize       scs
_bz2                contextvars         librosa             seaborn
_cffi_backend       convertdate         lightgbm            secrets
_codecs             copy                linecache           select
_codecs_cn          copyreg             llvmlite            selectors
_codecs_hk          crashtest           lmdb                send2trash
_codecs_iso2022     crcmod              locale              setuptools
_codecs_jp          crypt               locket              setuptools_git
_codecs_kr          csimdjson           logging             shapely
_codecs_tw          csv                 lsb_release         shelve
_collections        ctypes              lunarcalendar       shlex
_collections_abc    cufflinks           lxml                shutil
_compat_pickle      curses              lzma                signal
_compression        cv2                 macpath             simdjson
_contextvars        cvxopt              mailbox             site
_crypt              cvxpy               mailcap             sitecustomize
_csv                cycler              markdown            six
_ctypes             cymem               markupsafe          skimage
_ctypes_test        cython              marshal             sklearn
_curses             cythonmagic         marshmallow         sklearn_pandas
_curses_panel       daft                math                slugify
_cvxcore            dask                matplotlib          smart_open
_datetime           dataclasses         matplotlib_venn     smtpd
_dbm                datascience         mimetypes           smtplib
_decimal            datetime            missingno           sndhdr
_distutils_hack     dateutil            mistune             snowballstemmer
_dlib_pybind11      dbm                 mizani              socket
```

```
    _dummy_thread       dbus                mlxtend             socketserver
    _ecos               debugpy             mmap                socks
    _elementtree        decimal             modulefinder        sockshandler
    _functools          decorator           more_itertools      softwareproperties
    _hashlib            defusedxml          moviepy             sortedcontainers
    _heapq              descartes           mpmath              soundfile
    _imp                difflib             msgpack             spacy
    _io                 dill                multidict           spacy_legacy
    _json               dis                 multipledispatch    spacy_loggers
    _locale             distributed         multiprocessing     sphinx
    _lsprof             distutils           multitasking        spwd
    _lzma               dlib                murmurhash          sql
    _markupbase         dns                 music21             sqlalchemy
```

Loops in Python

- Need for loops
- While Loop
- For Loop

```
# While loop example -> program to print the table
# Program -> Sum of all digits of a given number
# Program -> keep accepting numbers from users till he/she enters a 0 and then find the avg
```

```
number = int(input('enter the number'))

i = 1

while i<11:
  print(number,'*',i,'=',number * i)
  i += 1
```

```
    enter the number12
    12 * 1 = 12
    12 * 2 = 24
    12 * 3 = 36
    12 * 4 = 48
    12 * 5 = 60
    12 * 6 = 72
    12 * 7 = 84
    12 * 8 = 96
    12 * 9 = 108
    12 * 10 = 120
```

```
# while loop with else

x = 1

while x < 3:
  print(x)
  x += 1

else:
  print('limit crossed')
```

```
    1
    2
    limit crossed
```

```
# Guessing game

# generate a random integer between 1 and 100
import random
jackpot = random.randint(1,100)

guess = int(input('guess karo'))
counter = 1
while guess != jackpot:
  if guess < jackpot:
    print('galat!guess higher')
  else:
    print('galat!guess lower')

  guess = int(input('guess karo'))
  counter += 1

else:
  print('correct guess')
  print('attempts',counter)
```

```
guess karo7
galat!guess higher
guess karo50
galat!guess lower
guess karo30
galat!guess higher
guess karo40
galat!guess lower
guess karo35
galat!guess lower
guess karo32
galat!guess higher
guess karo33
correct guess
attempts 7
```

```
# For loop demo

for i in {1,2,3,4,5}:
  print(i)
```

```
1
2
3
4
5
```

```
# For loop examples
```

Program - The current population of a town is 10000. The population of the town is increasing at the rate of 10% per year. You have to write a program to find out the population at the end of each of the last 10 years.

```
curr_pop = 10000

for i in range(10,0,-1):
  print(i,curr_pop)
  curr_pop = curr_pop - 0.1*curr_pop
```

```
10 10000
9 9000.0
8 8100.0
7 7290.0
6 6561.0
5 5904.9
4 5314.41
3 4782.969
2 4304.6721
1 3874.20489
```

Sequence sum

1/1! + 2/2! + 3/3! + ...

```
# code here
```

```
# For loop vs While loops (When to use what?)
```

Nested Loops

```
# Examples
```

```
# Program - Unique combination of 1,2,3,4
# Program - Pattern 1 and 2
```

Pattern 1

*

**

***

```
Start coding or generate with AI.
```

Pattern 2

1

121

12321

1234321

```
Start coding or generate with AI.
```

Loop Control Statement

- Break
- Continue
- Pass

```
# Break demo
```

```
# Break example (Linear Search) -> Prime number in a given range
```

```
# Continue demo
```

```
# Continue Example (Ecommerce)
```