# Basic code

```
In [4]: print(3+2)
```

```
5
```

```
In [6]: print(3-2)
```

```
1
```

```
In [8]: print(3*2)
```

```
6
```

```
In [10]: print(3/2)
```

```
1.5
```

```
In [12]: print(3**2)
```

```
9
```

```
In [14]: print(3%2)
```

```
1
```

```
In [16]: print(3//2)
```

```
1
```

```
In [20]: print(type(10))
```

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

```
In [22]: print(type(3.14))
```

```
<class 'float'>
```

```
In [24]: print(type(1+3j))
```

```
<class 'complex'>
```

```
In [26]: print(type('Anirudh'))
```

```
<class 'str'>
```

```
In [30]: print(type([1,2,3]))
```

```
<class 'list'>
```

```
In [28]: print(type({'name':'Anirudh'}))
```

```
<class 'dict'>
```

```
In [32]: print(type({9.8,3.14,2.7}))
```

```
<class 'set'>
```

```
In [36]: print(type((9.8,3.14,2.7)))
```

```
<class 'tuple'>
```

```
In [38]: print(type(3==3))
```

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

```
In [40]: print(type(3>=3))
```

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

# Operator

```
In [43]: print('Addition:',1+2)
```

```
Addition: 3
```

```
In [45]: print('Subtraction:',2-1)
```

```
Subtraction: 1
```

```
In [47]: print('Multiplication: ', 2 * 3)
```

```
Multiplication:  6
```

```
In [49]: print ('Division: ', 4 / 2)
```

```
Division:  2.0
```

```
In [51]: print('Division:', 6 / 2)
```

```
Division:  3.0
```

```
In [53]: print('Division:', 7 / 2)
```

```
Division:  3.5
```

```
In [55]: print('Division without the remainder: ', 7 // 2)
```

```
Division without the remainder:  3
```

```
In [57]: print('Modulus:',3%2)
```

```
Modulus: 1
```

```
In [59]: print('Exponential: ', 3 ** 2)
```

```
Exponential:  9
```

```
In [61]: print('Floating Number,PI', 3.14)
```

```
Floating Number,PI 3.14
```

```
In [63]: print('Floating Number, gravity', 9.81)
```

```
Floating Number, gravity 9.81
```

```
In [65]: print('Complex number: ', 1 + 1j)
```

```
Complex number:  (1+1j)
```

```
In [67]: print('Multiplying complex number: ',(1 + 1j) * (1-1j))
```

```
Multiplying complex number:  (2+0j)
```

```
In [69]:  a = 3
          b = 2
```

```
In [73]:  total = a + b
          total
```

```
Out[73]:  5
```

```
In [75]:  diff = a - b
```

```
In [77]:  product = a * b
          division = a / b
          remainder = a % b
          floor_division = a // b
          exponential = a ** b
```

```
In [79]:  print('a + b = ', total)
```

```
a + b =  5
```

```
In [81]:  print('a - b = ', diff)
```

```
a - b =  1
```

```
In [83]:  print('a * b = ', product)
```

```
a * b =  6
```

```
In [85]:  print('a / b = ', division)
```

```
a / b =  1.5
```

```
In [87]:  print('a % b = ', remainder)
```

```
a % b =  1
```

```
In [89]:  print('a // b = ', floor_division)
```

```
a // b =  1
```

```
In [91]:  print('a ** b = ', exponential)
```

```
a ** b =  9
```

```
In [99]:  num_one = 3
          num_two = 4
```

```
In [101…  total = num_one + num_two
```

```
In [103…  diff = num_two - num_one
          product = num_one * num_two
```

```
In [105…  div = num_two / num_two
```

```
In [107…  remainder = num_two % num_one
```

```
In [109…  print('total: ', total)
```

```
total:  7
```

In [111…

```
print('difference: ', diff)
```

```
difference:  1
```

In [113…

```
print('product: ', product)
```

```
product:  12
```

In [115…

```
print('division: ', div)
```

```
division:  1.0
```

In [117…

```
print('remainder: ', remainder)
```

```
remainder:  1
```

In [121…

```
radius = 10      # radius of a circle
area_of_circle = 3.14 * radius ** 2         # two * sign means exponent or power
```

In [123…

```
print('Area of a circle:', area_of_circle)
```

```
Area of a circle: 314.0
```

In [125…

```
length = 10
width = 20
area_of_rectangle = length * width
print('Area of rectangle:', area_of_rectangle)
```

```
Area of rectangle: 200
```

In [127…

```
mass = 75
gravity = 9.81
weight = mass * gravity
print(weight, 'N')
```

```
735.75 N
```

In [129…

```
print(3 > 2)     # True, because 3 is greater than 2
print(3 >= 2)    # True, because 3 is greater than 2
print(3 < 2)     # False,  because 3 is greater than 2
print(2 < 3)     # True, because 2 is less than 3
print(2 <= 3)    # True, because 2 is less than 3
print(3 == 2)    # False, because 3 is not equal to 2
print(3 != 2)    # True, because 3 is not equal to 2
```

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

In [131…

```
print(len('mango') == len('avocado'))  # False
print(len('mango') != len('avocado'))  # True
print(len('mango') < len('avocado'))   # True
print(len('milk') != len('meat'))      # False
print(len('milk') == len('meat'))      # True
print(len('tomato') == len('potato'))  # True
print(len('python') > len('dragon'))   # False
```

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

In [133…

```
print('True == True: ', True == True)
print('True == False: ', True == False)
print('False == False:', False == False)
print('True and True: ', True and True)
print('True or False:', True or False)
```

```
True == True:  True
True == False:  False
False == False: True
True and True:  True
True or False: True
```

In [135…

```
print('1 is 1', 1 is 1)                     # True - because the data values are t
print('1 is not 2', 1 is not 2)             # True - because 1 is not 2
print('A in Anirudh', 'A' in 'Anirudh') # True - A found in the string
print('D in Anirudh', 'D' in 'Anirudh') # False -there is no uppercase B
print('coding' in 'coding for all') # True - because coding for all has the wora
print('a in an:', 'a' in 'an')      # True
print('4 is 2 ** 2:', 4 is 2 ** 2)   # True
```

```
1 is 1 True
1 is not 2 True
A in Anirudh True
D in Anirudh False
True
a in an: True
4 is 2 ** 2: True
```

```
<>:1: SyntaxWarning: "is" with a literal. Did you mean "=="?
<>:2: SyntaxWarning: "is not" with a literal. Did you mean "!="?
<>:7: SyntaxWarning: "is" with a literal. Did you mean "=="?
<>:1: SyntaxWarning: "is" with a literal. Did you mean "=="?
<>:2: SyntaxWarning: "is not" with a literal. Did you mean "!="?
<>:7: SyntaxWarning: "is" with a literal. Did you mean "=="?
C:\Users\aniru\AppData\Local\Temp\ipykernel_21448\2320828669.py:1: SyntaxWarning:
"is" with a literal. Did you mean "=="?
  print('1 is 1', 1 is 1)                     # True - because the data values are
the same
C:\Users\aniru\AppData\Local\Temp\ipykernel_21448\2320828669.py:2: SyntaxWarning:
"is not" with a literal. Did you mean "!="?
  print('1 is not 2', 1 is not 2)             # True - because 1 is not 2
C:\Users\aniru\AppData\Local\Temp\ipykernel_21448\2320828669.py:7: SyntaxWarning:
"is" with a literal. Did you mean "=="?
  print('4 is 2 ** 2:', 4 is 2 ** 2)   # True
```

In [137…

```
print(3 > 2 and 4 > 3) # True - because both statements are true
print(3 > 2 and 4 < 3) # False - because the second statement is false
print(3 < 2 and 4 < 3) # False - because both statements are false
print(3 > 2 or 4 > 3)  # True - because both statements are true
print(3 > 2 or 4 < 3)  # True - because one of the statement is true
print(3 < 2 or 4 < 3)  # False - because both statements are false
```

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

In [139…

```
print(not 3 > 2)      # False - because 3 > 2 is true, then not True gives False
print(not True)       # False - Negation, the not operator turns true to false
print(not False)      # True
print(not not True)   # True
print(not not False) # False
```

```
False
False
True
True
False
```

# String

In [142…

```
letter='P'
```

In [144…

```
print(letter)
```

```
P
```

In [146…

```
print(len(letter))
```

```
1
```

In [148…

```
greeting='Hello World'
```

In [150…

```
print(greeting)
```

```
Hello World
```

In [152…

```
sentence="I Love Python"
```

In [154…

```
print(sentence)
```

```
I Love Python
```

In [156…

```
multiline_string = '''I am a student and enjoy learning.
I didn't find anything as rewarding as learning.
That is why I joined 30 days of python.'''
```

In [158…

```
print(multiline_string)
```

```
I am a student and enjoy learning.
I didn't find anything as rewarding as learning.
That is why I joined 30 days of python.
```

In [ ]:

```
multiline_string = """I am a student and enjoy learning.
I didn't find anything as rewarding as learning.
That is why I joined 30 days of python."""
```

In [160…

```
print(multiline_string)
```

```
I am a student and enjoy learning.
I didn't find anything as rewarding as learning.
That is why I joined 30 days of python.
```

In [162…

```python
first_name = 'Anirudh'
last_name = 'Bharadwaj'
space = ' '
full_name = first_name  +  space + last_name
print(full_name)
```

```
Anirudh Bharadwaj
```

In [164…

```python
print(len(first_name))
print(len(last_name))
print(len(first_name) > len(last_name))
print(len(full_name))
```

```
7
9
False
17
```

In [166…

```python
language = 'Python'
a,b,c,d,e,f = language
print(a)
print(b)
print(c)
print(d)
print(e)
print(f)
```

```
P
y
t
h
o
n
```

In [168…

```python
language = 'Python'
first_letter = language[0]
print(first_letter)
second_letter = language[1]
print(second_letter)
last_index = len(language) - 1
last_letter = language[last_index]
print(last_letter)
```

```
P
y
n
```

In [170…

```python
language = 'Python'
last_letter = language[-1]
print(last_letter)
second_last = language[-2]
print(second_last)
```

```
n
o
```

In [172…

```python
language = 'Python'
first_three = language[0:3]
```

```
last_three = language[3:6]
print(last_three)
```

```
hon
```

In [174…

```
last_three = language[-3:]
print(last_three)
last_three = language[3:]
print(last_three)
```

```
hon
hon
```

In [176…

```
language = 'Python'
pto = language[0:6:2]
print(pto)
```

```
Pto
```

In [178…

```
print('I hope every one enjoying the python challenge.\nDo you ?')
print('Days\tTopics\tExercises')
print('Day 1\t3\t5')
print('Day 2\t3\t5')
print('Day 3\t3\t5')
print('Day 4\t3\t5')
print('This is a back slash  symbol (\\)')
print('In every programming language it starts with \"Hello, World!\"')
```

```
I hope every one enjoying the python challenge.
Do you ?
Days    Topics  Exercises
Day 1   3       5
Day 2   3       5
Day 3   3       5
Day 4   3       5
This is a back slash  symbol (\)
In every programming language it starts with "Hello, World!"
```

In [180…

```
challenge = 'thirty days of python'
print(challenge.capitalize())
```

```
Thirty days of python
```

In [182…

```
challenge = 'thirty days of python'
print(challenge.count('y'))
print(challenge.count('y', 7, 14))
print(challenge.count('th'))
```

```
3
1
2
```

In [184…

```
challenge = 'thirty days of python'
print(challenge.endswith('on'))
print(challenge.endswith('tion'))
```

```
True
False
```

In [186…

```
challenge = 'thirty\tdays\tof\tpython'
print(challenge.expandtabs())
print(challenge.expandtabs(10))
```

```
thirty  days    of      python
thirty    days      of        python
```

In [188…

```
challenge = 'thirty days of python'
print(challenge.find('y'))
print(challenge.find('th'))
```

```
5
0
```

In [190…

```
first_name = 'Anirudh'
last_name = 'Bharadwaj'
job = 'Student'
country = 'India'
sentence = 'I am {} {}. I am a {}. I live in {}.'.format(first_name, last_name,
print(sentence)
```

```
I am Anirudh Bharadwaj. I am a Student. I live in India.
```

In [192…

```
radius = 10
pi = 3.14
area = pi
result = 'The area of circle with {} is {}'.format(str(radius), str(area))
print(result)
```

```
The area of circle with 10 is 3.14
```

In [194…

```
challenge = 'thirty days of python'
print(challenge.find('y'))
print(challenge.find('th'))
```

```
5
0
```

In [196…

```
challenge = 'ThirtyDaysPython'
print(challenge.isalnum())
```

```
True
```

In [ ]:

```
challenge = '30DaysPython'
print(challenge.isalnum())
```

In [198…

```
challenge = 'thirty days of python'
print(challenge.isalnum())
```

```
False
```

In [200…

```
challenge = 'thirty days of python 2019'
print(challenge.isalnum())
```

```
False
```

In [208…

```
challenge = 'thirtydaysofpython'
print(challenge.isalpha())
```

```
True
```

In [204…

```
num = '123'
print(num.isalpha())
```

```
False
```

In [210…

```
challenge = 'thirty days of python'
print(challenge.isalpha())
```

```
False
```

In [216…

```
challenge = 'Thirty'
print(challenge.isdigit())
challenge = '30'
print(challenge.isdigit())
```

```
False
True
```

In [224…

```
num = '10'
print(num.isdecimal())
num = '10.5'
print(num.isdecimal())
```

```
True
False
```

In [226…

```
challenge = '30DaysOfPython'
print(challenge.isidentifier())
challenge = 'thirty_days_of_python'
print(challenge.isidentifier())
```

```
False
True
```

In [228…

```
challenge = 'thirty days of python'
print(challenge.islower())
challenge = 'Thirty days of python'
print(challenge.islower())
```

```
True
False
```

In [230…

```
challenge = 'thirty days of python'
print(challenge.isupper())
challenge = 'THIRTY DAYS OF PYTHON'
print(challenge.isupper())
```

```
False
True
```

In [232…

```
num = '10'
print(num.isnumeric())
print('ten'.isnumeric())
```

```
True
False
```

In [234…

```
web_tech = ['HTML', 'CSS', 'JavaScript', 'React']
result = '#, '.join(web_tech)
print(result)
```

```
HTML#, CSS#, JavaScript#, React
```

In [244…

```
challenge = ' thirty days of python '
print(challenge.strip('y'))
```

```
 thirty days of python
```

In [238…

```
challenge = 'thirty days of python'
print(challenge.replace('python', 'coding'))
```

```
thirty days of coding
```

In [246…

```
challenge = 'thirty days of python'
print(challenge.split())
```

```
['thirty', 'days', 'of', 'python']
```

In [248…

```
challenge = 'thirty days of python'
print(challenge.title())
```

```
Thirty Days Of Python
```

In [250…

```
challenge = 'thirty days of python'
print(challenge.swapcase())
challenge = 'Thirty Days Of Python'
print(challenge.swapcase())
```

```
THIRTY DAYS OF PYTHON
tHIRTY dAYS oF pYTHON
```

In [252…

```
challenge = 'thirty days of python'
print(challenge.startswith('thirty')) # True
challenge = '30 days of python'
print(challenge.startswith('thirty'))
```

```
True
False
```

# Variables

In [254…

```
first_name = 'ANIRUDH'
last_name = 'BHARADWAJ'
country = 'HYD'
city = 'TELENGANA'
age = 500048
is_married = False
skills = ['HTML', 'CSS', 'JS', 'React', 'Python']
person_info = {
    'firstname':'Anirudh',
    'lastname':'Bharadwaj',
    'country':'India',
    'city':'Hyderabad'
    }
```

In [256…

```
print('First name:', first_name)
print('First name length:', len(first_name))
print('Last name: ', last_name)
print('Last name length: ', len(last_name))
print('Country: ', country)
print('City: ', city)
print('Age: ', age)
print('Married: ', is_married)
print('Skills: ', skills)
print('Person information: ', person_info)
```

```
First name: ANIRUDH
First name length: 7
Last name:  BHARADWAJ
Last name length:  9
Country:  HYD
City:  TELENGANA
Age:  500048
Married:  False
Skills:  ['HTML', 'CSS', 'JS', 'React', 'Python']
Person information:  {'firstname': 'Anirudh', 'lastname': 'Bharadwaj', 'country':
'India', 'city': 'Hyderabad'}
```

In [262…

```
first_name, last_name, country, age, is_married = 'Anirudh', 'Bharadwaj', 'India
```

In [264…

```
print(first_name, last_name, country, age, is_married)
print('First name:', first_name)
print('Last name: ', last_name)
print('Country: ', country)
print('Age: ', age)
print('Married: ', is_married)
```

```
Anirudh Bharadwaj India 23 False
First name: Anirudh
Last name:  Bharadwaj
Country:  India
Age:  23
Married:  False
```