In [2]:

```python
#Task No:01

class DataType:
    def __init__(self, name, value) -> None:
        self.name = name
        self.value = value

#Qustion Part:

data_type1 = DataType('Integer', 1234)
print(data_type1.name)
print(data_type1.value)
print('=====================')
data_type2 = DataType('String', 'Hello')
print(data_type2.name)
print(data_type2.value)
print('====================')
data_type3 = DataType('Float', 4.0)
print(data_type3.name)
print(data_type3.value)

#Output:

#Integer
#1234
#=====================
#String
#Hello
#=====================
#Float
#4.0
```

```
Integer
1234
=====================
String
Hello
====================
Float
4.0
```

In [2]:

```
#Task No:02

class Flower:
    def __init__(self):
        self.name=None
        self.color=""
        self.num_of_petal=0


#Qustion Part:

flower1 = Flower()
flower1.name="Rose"
flower1.color="Red"
flower1.num_of_petal=6
print("Name of this flower:", flower1.name)
print("Color of this flower:",flower1.color)
print("Number of petal:",flower1.num_of_petal)
print("=====================")
flower2 = Flower()
flower2.name="Orchid"
flower2.color="Purple"
flower2.num_of_petal=4
print("Name of this flower:",flower2.name)
print("Color of this flower:",flower2.color)
print ("Number of petal:",flower2. num_of_petal)

#Output:

#Name of this flower: Rose
#Color of this flower: Red
#Number of petal: 6
#=====================
#Name of this flower: Orchid
#Color of this flower: Purple
#Number of petal: 4
```

```
Name of this flower: Rose
Color of this flower: Red
```

```
Number of petal: 6
=====================
Name of this flower: Orchid
Color of this flower: Purple
Number of petal: 4
```

In [1]:

```
#Task No:03

#Qustion Part:

class Wadiya():
    def __init__(self):
        self.name='Aladeen'
        self.designation = 'President Prime Minister Admiral General'
        self.num_of_wife=100
        self.dictator = True

#Output:

#Name of President: Aladeen
#Designation: President Prime Minister Admiral General
#Number of wife: 100
#Is he/she a dictator: True
#Part 2:
#Name of President: Donald Trump
#Designation: President
#Number of wife: 1
#Is he/she a dictator: False



w1=Wadiya()
print("part1:")
print(f"Name of the president: {w1.name}")
print(f"Designation: {w1.designation}")
print(f"Number of wife: {w1.num_of_wife}")
print(f"Is he/she a dictator: {w1.dictator}")




w2=Wadiya()
w2.Name : "Donald Trump"
w2.Designation: "President"

```

```
w2.num_of_wife : 1
w2. dictator: False



print("part2:")
print(f"Name of the president: {w2.name}")
print(f"Designation: {w2.designation}")
print(f"Number of wife: {w2.num_of_wife}")
print(f"Is he/she a dictator: {w2.dictator}")
```

```
part1:
Name of the president: Aladeen
Designation: President Prime Minister Admiral General
Number of wife: 100
Is he/she a dictator: True
part2:
Name of the president: Aladeen
Designation: President Prime Minister Admiral General
Number of wife: 100
Is he/she a dictator: True
```

In [2]:

```python
#Task No:04

class Joker:
    def __init__(self,a,b,c):
        self.name=a
        self.power=b
        self.is_he_psycho=c




#Qustion Part:

j1 = Joker('Heath Ledger', 'Mind Game', False)
print(j1.name)
print(j1.power)
print(j1.is_he_psycho)
print("=====================")
j2 = Joker('Joaquin Phoenix', 'Laughing out Loud', True)
print(j2.name)
print(j2.power)
print(j2.is_he_psycho)
print("=====================")
if j1 == j2:
    print('same')
else:
    print('different')
j2.name = 'Heath Ledger'
if j1.name == j2.name:
    print('same')
else:
    print('different')


#Output:

#Heath Ledger
#Mind Game
```

```
#False
#=====================
#Joaquin Phoenix
#Laughing out Loud
#True
#=====================
#different
#same
```

```
Heath Ledger
Mind Game
False
=====================
Joaquin Phoenix
Laughing out Loud
True
=====================
different
same
```

In [1]:

```python
#Task No:05

class pokemon:
    def __init__(self,a,b,c,d,e):
        self.pokemon1_name=a
        self.pokemon1_power=c
        self.pokemon2_name=b
        self.pokemon2_power=d
        self.damage_rate=e




#Qustion Part:

team_pika = pokemon('pikachu', 'charmander', 90, 60, 10)
print('=======Team 1=======')
print('Pokemon 1:',team_pika.pokemon1_name,team_pika.pokemon1_power)
print('Pokemon 2:',team_pika.pokemon2_name,team_pika.pokemon2_power)
pika_combined_power = (team_pika.pokemon1_power +team_pika.pokemon2_power) *
print('Combined Power:', pika_combined_power)



team_bulbasaur = pokemon('bulbasaur','squirtle', 80, 70, 9)
print('=======Team 2=======')
print('Pokemon 1:',team_bulbasaur.pokemon1_name,
team_bulbasaur.pokemon1_power)
print('Pokemon 2:',team_bulbasaur.pokemon2_name,
team_bulbasaur.pokemon2_power)
b_combined_power = (team_bulbasaur.pokemon1_power +
team_bulbasaur.pokemon2_power) * team_bulbasaur.damage_rate
print('Combined Power:', b_combined_power)



#Output:

```

```

#=======Team 1=======
#Pokemon 1: pikachu 90
#Pokemon 2: charmander 60
#Combined Power: 1500
#=======Team 2=======
#Pokemon 1: bulbasaur 80
#Pokemon 2: squirtle 70
#Combined Power: 1350
```

```
=======Team 1=======
Pokemon 1: pikachu 90
Pokemon 2: charmander 60
Combined Power: 1500
=======Team 2=======
Pokemon 1: bulbasaur 80
Pokemon 2: squirtle 70
Combined Power: 1350
```

In [2]:

```
#Task No:06

class Player:
    def __init__(self):
        self.name=None
        self.jersy_number=None
        self.position=None





#Qustion Part:


player1 = Player()
player1.name = "Ronaldo"
player1.jersy_number = 9
player1.position = "Striker"
print("Name of the Player:", player1.name)
print("Jersey Number of player:", player1.jersy_number)
print("Position of player:", player1.position)
print("============================")
player2 = Player()
player2.name = "Neuer"
player2.jersy_number = 1
player2.position = "Goal Keeper"
print("Name of the player:", player2.name)
print("Jersey Number of player:", player2.jersy_number)
print("Position of player:", player2.position)



#Output:


#Name of the Player: Ronaldo
#Jersy Number of player: 9
```

```
#Position of player: Striker
#===========================
#Name of the player: Neuer
#jersy Number of player: 1
#Position of player: Goal Keeper
```

```
Name of the Player: Ronaldo
Jersey Number of player: 9
Position of player: Striker
===========================
Name of the player: Neuer
Jersey Number of player: 1
Position of player: Goal Keeper
```

In [3]:

```
#Task No:07

class country:
    def __init__(self):
        self.name="Bangladesh"
        self.continent="Asia"
        self.capital="Dhaka"
        self.fifa_ranking=187



#Qustion Part:


country = country()
print('Name:',country.name)
print('Continent:',country.continent)
print('Capital:',country.capital)
print('Fifa Ranking:',country.fifa_ranking)
print('===================')
country.name = "Belgium"
country.continent = "Europe"
country.capital = "Brussels"
country.fifa_ranking = 1
print('Name:',country.name)
print('Continent:',country.continent)
print('Capital:',country.capital)
print('Fifa Ranking:',country.fifa_ranking)






#Output:

#Name: Bangladesh
#Continent: Asia
```

```
#Capital: Dhaka
#Fifa Ranking: 187
#===================
#Name: Belgium
#Continent: Europe
#Capital: Brussels
#Fifa Ranking: 1
```

```
Name: Bangladesh
Continent: Asia
Capital: Dhaka
Fifa Ranking: 187
===================
Name: Belgium
Continent: Europe
Capital: Brussels
Fifa Ranking: 1
```

In [4]:

```
#Task No:08

class DemonSlayer:
    def __init__(self,a,b,c,d):
        self.name=a
        self.style=b
        self.number_of_technique=c
        self.kill=d


#Qustion Part:


tanjiro = DemonSlayer("Tanjiro", "Water Breathing", 10, 10)
print('Name:',tanjiro.name)
print('Fighting Style:',tanjiro.style)
print(f'Knows {tanjiro.number_of_technique} technique(s) and has killed {tan
print('===================')
zenitsu = DemonSlayer("Zenitsu", "Thunder Breathing", 1, 4)
print('Name:',zenitsu.name)
print('Fighting Style:',zenitsu.style)
print(f'Knows {zenitsu.number_of_technique} technique(s) and has killed {zen
print('===================')
inosuke = DemonSlayer("Inosuke", "Beast Breathing", 5, 7)
print('Name:',inosuke.name)
print('Fighting Style:',inosuke.style)
print(f'Knows {inosuke.number_of_technique} technique(s) and has killed {ino
print('===================')
print(f'{tanjiro.name}, {zenitsu.name}, {inosuke.name} knows total {tanjiro.
print(f'They have killed total {tanjiro.kill + zenitsu.kill + inosuke.kill}



#Output:


# Name: Tanjiro
#Fighting Style: Water Breathing
```

```
#Knows 10 technique(s) and has killed 10 demon(s)
#===================
#Name: Zenitsu
#Fighting Style: Thunder Breathing
#Knows 1 technique(s) and has killed 4 demon(s)
#===================

#Name: Inosuke
#Fighting Style: Beast Breathing
#Knows 5 technique(s) and has killed 7 demon(s)
#===================
#Tanjiro, Zenitsu, Inosuke knows total 16 techniques
#They have killed total 21 demons
```

```
Name: Tanjiro
Fighting Style: Water Breathing
Knows 10 technique(s) and has killed 10 demon(s)
===================
Name: Zenitsu
Fighting Style: Thunder Breathing
Knows 1 technique(s) and has killed 4 demon(s)
===================
Name: Inosuke
Fighting Style: Beast Breathing
Knows 5 technique(s) and has killed 7 demon(s)
===================
Tanjiro, Zenitsu, Inosuke knows total 16 techniques
They have killed total 21 demons
```

In [5]:

```
#Task No:09

class box:
    def __init__(self,lst):
        self.height=lst[0]
        self.width=lst[1]
        self.breadth=lst[2]
        print("Creating a Box!")
        print("Volume of the box is",self.height*self.width*self.breadth,"cu





#Qustion Part:


print("Box 1")
b1 = box([10,10,10])
print("========================")
print("Height:", b1.height)
print("Width:", b1.width)
print("Breadth:", b1.breadth)
print("-------------------------")
print("Box 2")
b2 = box((30,10,10))
print("========================")
print("Height:", b2.height)
print("Width:", b2.width)
print("Breadth:", b2.breadth)
b2.height = 300
print("Updating Box 2!")
print("Height:", b2.height)
print("Width:", b2.width)
print("Breadth:", b2.breadth)
print("-------------------------")
print("Box 3")
b3 = b2
```

```
print("Height:", b3.height)
print("Width:", b3.width)
print("Breadth:", b3.breadth)




#Output:

#Box 1
#Creating a Box!
#Volume of the box is 1000 cubic units.
#=========================
#Height: 10
#Width: 10
#Breadth: 10
#---------------------------------------------
#Box 2
#Creating a Box!
#Volume of the box is 3000 cubic units.
#=========================
#Height: 30
#Width: 10
#Breadth: 10
#Updating Box 2!
#Height: 300
#Width: 10
#Breadth: 10
#---------------------------------------------
#Box 3
#Height: 300
#Width: 10
#Breadth: 10
```

```
Box 1
Creating a Box!
Volume of the box is 1000 cubic units
=========================
Height: 10
Width: 10
Breadth: 10
```

```
-------------------------
Box 2
Creating a Box!
Volume of the box is 3000 cubic units
=========================
Height: 30
Width: 10
Breadth: 10
Updating Box 2!
Height: 300
Width: 10
Breadth: 10
-------------------------
Box 3
Height: 300
Width: 10
Breadth: 10
```

In [1]:

```
#Task No:10

class buttons:
    def __init__(self, *button_data) -> None:
        self.word, self.spaces, self.border = button_data
        self.pera = 1 + self.spaces + len(self.word.strip()) + self.spaces +
        print(f"{self.word} Button Specification:")
        print(f"Button name: {self.word}")
        print(f"Number of the border characters for the top and the bottom: 
        print(f"Number of spaces between the left side border and the first 
        print(f"Number of spaces between the right side border and the last 
        print(f"Characters representing the borders: {self.border}")
        print(f"{self.border*self.pera}\n{self.border}{self.spaces*' ' }{sel



#Qustion Part:



word = "CANCEL"
spaces = 10
border = 'x'
b1 = buttons(word, spaces, border)
print("=======================================================")
b2 = buttons("Notify",3, '!')
print("=======================================================")
b3 = buttons('SAVE PROGRESS', 5, '$')



#Output:

#CANCEL Button Specifications:
#Button name: CANCEL
#Number of the border characters for the top and the bottom: 28
#Number of spaces between the left side border and the first character of th
#name: 10
```

```
#Number of spaces between the right side border and the last character of th
#name: 10
#Characters representing the borders: x
#xxxxxxxxxxxxxxxxxxxxxxxxxxxx
#x CANCEL x
#xxxxxxxxxxxxxxxxxxxxxxxxxxxx
#=========================================================

#Notify Button Specifications:
#Button name: Notify
#Number of the border characters for the top and the bottom: 14
#Number of spaces between the left side border and the first character of th
#name: 3
#Number of spaces between the right side border and the last character of th
#name: 3
#Characters representing the borders: !
#!!!!!!!!!!!!!!
#! Notify !
#!!!!!!!!!!!!!!
#=========================================================
#SAVE PROGRESS Button Specifications:
#Button name: SAVE PROGRESS
#Number of the border characters for the top and the bottom: 25
#Number of spaces between the left side border and the first character of th
#name: 5
#Number of spaces between the right side border and the last character of th
#name: 5
#Characters representing the borders: $
#$$$$$$$$$$$$$$$$$$$$$$$$$
#$ SAVE PROGRESS $
#$$$$$$$$$$$$$$$$$$$$$$$$$
```

```
CANCEL Button Specification:
Button name: CANCEL
Number of the border characters for the top and the bottom: 28
Number of spaces between the left side border and the first character of the button  name: 10
Number of spaces between the right side border and the last character of the button  name: 10
Characters representing the borders: x
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
x          CANCEL          x
xxxxxxxxxxxxxxxxxxxxxxxxxxxx
=======================================================
Notify Button Specification:
Button name: Notify
Number of the border characters for the top and the bottom: 14
Number of spaces between the left side border and the first character of the button  name: 3
Number of spaces between the right side border and the last character of the button  name: 3
Characters representing the borders: !
!!!!!!!!!!!!!!
!   Notify   !
!!!!!!!!!!!!!!
=======================================================
SAVE PROGRESS Button Specification:
Button name: SAVE PROGRESS
Number of the border characters for the top and the bottom: 25
Number of spaces between the left side border and the first character of the button  name: 5
Number of spaces between the right side border and the last character of the button  name: 5
Characters representing the borders: $
$$$$$$$$$$$$$$$$$$$$$$$$$
$     SAVE PROGRESS     $
$$$$$$$$$$$$$$$$$$$$$$$$$
```