In [1]:

```
1  name = "Arjun"
2  def printName():
3     name = "Sathish"
4     print("Name:", name)
5  printName()
6  print("Name: ", name)
```

Name: Sathish Name: Arjun

In [2]:

```
name = "Arjun"
def printName():
    name = "Sathish"
    print("Name:",name)
    print("Name:",name)
printName()
```

Name: Sathish Name: Sathish

Global Variable: Outside of all the functions Local Variable: Within a function block scope of global variable is both inside an outside the func. scope of local variable is only iside function local variable overwrites the global variable ig given inside a function

In [3]:

```
1
   #eg
 2
   def example():
        sub = "JAVA"
 3
4
        print("I dont know", sub)
 5
        print("My Name is", name)
   def display():
 6
7
        var = "songs"
8
        print("I Hate", var)
9
        print("My Name is", name)
10
   example()
   display()
```

I dont know JAVA My Name is Arjun I Hate songs My Name is Arjun

In [4]:

```
num1 = int(input("Enter value:")) #global variable
   num2 = int(input("Enter value:")) #global Varible
   def add():
4
        print("Addition:", num1+num2)
 5
   def sub():
 6
       print("Subtraction:", num1-num2)
 7
   def mul():
        print("Multiplication:", num1*num2)
8
9
   def div():
       print("Division:",num1/num2)
10
   inp = int(input("Enter\n'1'For Addition \n'2'For Subraction \n'3'For Multiplication \n'
11
   if (inp == 1):
12
13
        add()
14
   elif (inp == 2):
15
       sub()
16
   elif (inp == 3):
17
       mul()
   elif (inp == 4):
18
19
       div()
20
   else:
21
        print('Please Enter an Valid Input')
```

```
Enter value:5
Enter value:5
Enter
'1'For Addition
'2'For Subraction
'3'For Multiplication
'4'For Division4
Division: 1.0
```

'float' object cannot be interpreted as an integer

set.add() takes exactly one argument (2 given)

In [5]:

```
x = 0
 1
 2
   a = 5
 3 b = 5
 4
   if a > 0:
 5
        if b < 0:
            x = x + 5
 6
 7
        elif a > 5:
 8
            x = x + 4
9
        else:
10
            x = x + 3
11
   else:
12
        x = x + 2
13
   print(x)
```

3

In [6]:

-1 -2 -3 -4 -5 -6 -7 -8 -9 9 -1 -2 -3 -4 -5 -6 -7 -8 -9 18 -1 -2 -3 -4 -5 -6 -7 -8 -9 27 -1 -2 -3 -4 -5 -6 -7 -8 -9 36 -1 -2

-3 -4 -5 -6 -7 -8 -9 45 -1 -2

-3

-4

-5

-6

-7

-8

-9

54

-1

-2

-3

-4

-5

-6

-7 -8

-9

63

-1

-2

-3

-4 -5

-6

-7

-8 -9

72

-1

-2

-3 -4

-5

-6

-7

-8

-9

81

-1

-2 -3

-4

-5

-6

-7 -8

-9 90

```
In [7]:
```

```
def parking_charge():
 1
 2
        if (inp == 1):
            time = int(input("Enter the number of Hours:"))
 3
 4
            print("The total fare is ",time*10)
 5
        elif (inp == 2):
 6
            time = int(input("Enter the number of Hours:"))
 7
            print("The total fare is ",time*15)
 8
        elif (inp == 3):
 9
            time = int(input("Enter the number of Hours:"))
            print("The total fare is ",time*20)
10
   inp = int(input("Enter\n'1'For Bike \n'2'For Car \n'3'For Any heavy load vehicle"))
11
12
   parking_charge()
13
```

Enter

- '1'For Bike
- '2'For Car
- '3'For Any heavy load vehicle4

MODULE

```
In [8]:
```

```
# A module is a file containing python definitions and statements.
# can be function, class, variable
```

In [9]:

```
1 parking_charge()
```

In [10]:

```
1 import calendar as c
```

In [11]:

```
1 print(c.month(2022,9))
```

In [13]:

```
1  year = int(input("Enter the year: "))
2  mon = int(input("Enter the Month: "))
3  print(c.month(year,mon))
```

```
In [14]:
```

```
1 year = int(input("Enter the year: "))
2 print(c.calendar(year))
```

Enter the year: 2022

2022

January									Fel	orua	ary				March						
Мо	Tu	We		-	Sa	Su	Мо	Tu	We	Th	Fr	Sa	Su	Мо	Tu	We	Th	Fr	Sa	Su	
					1	2		1	2	3	4	5	6		1	2	3	4	5	6	
3	4	5	6	7	8	9	7	8	9	10	11	12	13	7	8	9	10	11	12	13	
10	11	12	13	14	15	16	14	15	16	17	18	19	20	14	15	16	17	18	19	20	
17	18	19	20	21	22	23	21	22	23	24	25	26	27	21	22	23	24	25	26	27	
24	25	26	27	28	29	30	28							28	29	30	31				
31																					
	April							May							June						
Мо	Tu	We	Th	Fr	Sa	Su	Мо	Tu	We	Th	Fr	Sa	Su	Мо	Tu	We	Th	Fr	Sa	Su	
				1	2	3							1			1	2	3	4	5	
4	5	6	7	8	9	10	2	3	4	5	6	7	8	6	7	8	9	10	11	12	
11	12	13	14	15	16	17	9	10	11	12	13	14	15	13	14	15	16	17	18	19	
18	19	20	21	22	23	24	16	17	18	19	20	21	22	20	21	22	23	24	25	26	
25	26	27	28	29	30		23	24	25	26	27	28	29	27	28	29	30				
							30	31													
		_																ı			
	July Mo Tu We Th Fr Sa Su							August Mo Tu We Th Fr Sa Su							September Mo Tu We Th Fr Sa Su						
MO	ΙU	we	ın											MO	Ιu	we					
	_	_	_	1	2	3	1	2	3	4	5	6	7				1	2	3	4	
4	_	6	7	8	9				40		4.0			_	_	_	_	_			
	11		4 4	4-	_	10	8	9				13		5		7	8	9	10	11	
ПX			14		16	17	15	16	17	18	19	13 20	21	12	13	14	15	16	17	18	
	19	20	21	22	16 23	17 24	15 22	16 23	17 24	18	19	13	21	12 19	13 20	14 21	15 22	16 23	17	18	
	19		21	22	16 23	17 24	15 22	16	17 24	18	19	13 20	21	12 19	13	14 21	15 22	16 23	17	18	
	19	20 27	21 28	22 29	16 23	17 24	15 22	16 23	17 24 31	18 25	19 26	13 20	21	12 19	13 20	14 21 28	15 22 29	16 23 30	17	18	
25	19 26	20 27 Oct	21 28 tobe	22 29 er	16 23 30	17 24 31	15 22 29	16 23 30	17 24 31 Nov	18 25 /eml	19 26 oer	13 20 27	21 28	12 19 26	13 20 27	14 21 28 Dec	15 22 29	16 23 30 ber	17 24	18 25	
25	19 26	20 27	21 28 tobe	22 29 er	16 23 30 Sa	17 24 31	15 22 29	16 23 30 Tu	17 24 31 Nov We	18 25 /eml	19 26 er Fr	13 20 27 Sa	21 28 Su	12 19 26	13 20	14 21 28 Dec	15 22 29 cemb	16 23 30 ber Fr	17 24 Sa	18 25 Su	
25	19 26 Tu	20 27 Oct	21 28 tobe	22 29 er	16 23 30	17 24 31	15 22 29	16 23 30	17 24 31 Nov	18 25 /emb Th 3	19 26 er Fr 4	13 20 27	21 28 Su 6	12 19 26	13 20 27	14 21 28 Dec	15 22 29	16 23 30 ber	17 24	18 25	
25 Mo	19 26 Tu 4	20 27 Oct We	21 28 tobe Th	22 29 er Fr	16 23 30 Sa 1 8	17 24 31 Su 2	15 22 29 Mo	16 23 30 Tu 1	17 24 31 Nov We 2 9	18 25 /emb Th 3	19 26 er Fr 4 11	13 20 27 Sa 5 12	21 28 Su 6	12 19 26 Mo	13 20 27 Tu	14 21 28 Dec We	15 22 29 cemb Th 1 8	16 23 30 ber Fr 2	17 24 Sa 3	18 25 Su 4	
25 Mo	19 26 Tu 4 11	20 27 Oct We 5 12	21 28 tobe Th	22 29 er Fr 7 14	16 23 30 Sa 1 8 15	17 24 31 Su 2 9	15 22 29 Mo 7 14	16 23 30 Tu 1 8 15	17 24 31 Nov We 2 9 16	18 25 /emt Th 3 10 17	19 26 Per Fr 4 11 18	13 20 27 Sa 5 12	21 28 Su 6 13 20	12 19 26 Mo	13 20 27 Tu	14 21 28 Dec We	15 22 29 cemb Th 1 8 15	16 23 30 ber Fr 2 9 16	17 24 Sa 3 10	18 25 Su 4 11 18	
25 Mo 3 10 17	19 26 Tu 4 11 18	20 27 Oct We 5 12	21 28 tobe Th 6 13 20	22 29 er Fr 7 14 21	16 23 30 Sa 1 8 15 22	17 24 31 Su 2 9 16	15 22 29 Mo 7 14 21	16 23 30 Tu 1 8 15	17 24 31 Nov We 2 9 16 23	18 25 /emt Th 3 10 17	19 26 Per Fr 4 11 18	13 20 27 Sa 5 12 19	21 28 Su 6 13 20	12 19 26 Mo 5 12	13 20 27 Tu 6 13 20	14 21 28 Dec We 7 14 21	15 22 29 Th 1 8 15 22	16 23 30 ber Fr 2 9 16	17 24 Sa 3 10 17 24	18 25 Su 4 11 18	

In [2]:

```
def Hanoi(n,one,two,three):
 2
        if n>0:
 3
            Hanoi(n-1,one,three,two)
 4
            if one:
 5
                three.append(one.pop())
 6
            Hanoi(n-1, two, one, three)
 7
   one = [1,2,3,4]
 8
   three = []
 9
   two = []
10 Hanoi(len(one), one, two, three)
11 print(one)
12 print(two)
13 print(three)
```

```
[]
[]
[1, 2, 3, 4]
```

In []:

In []:

1