

Q)WAP to add two numbers

```
In [1]: number1=100
number2=200
add=number1+number2
print("the addition of {} and {} is {}".format(number1,number2,add))
```

the addition of 100 and 200 is 300

Q)Wap ask the user enter a number by using keyboard

input

```
In [4]: input() # input from the keyboard
```

100

Out[4]: '100'

```
In [6]: input("enter a number:")
```

enter a number:100

Out[6]: '100'

```
In [7]: input("enter your name:")
```

enter your name:python

Out[7]: 'python'

```
In [8]: '100'+'200'
```

Out[8]: '100200'

Note

key board will give string format only

```
In [9]: number1=input("enter a number1:") # step-1: it will ask the number1='100'
number2=input("enter a number2:") # step-2: it will ask the number2='200'
add=number1+number2 # '100'+'200'='100200'
print("the addition of {} and {} is {}".format(number1,number2,add))
```

enter a number1:100
enter a number2:200
the addition of 100 and 200 is 100200

```
In [ ]: a='100'
int(a)
```

```
In [10]: number1=input("enter a number1:") # step-1: it will ask the number1='100'
number2=input("enter a number2:") # step-2: it will ask the number2='200'
add=int(number1)+int(number2) # int('100')+int('200')=100+200
print("the addition of {} and {} is {}".format(number1,number2,add))
```

```
enter a number1:100
enter a number2:200
the addition of 100 and 200 is 300
```

```
In [ ]: # -----case-1:-----
number1=100 # manually we provided value 100
number2=200 # manually we provided value 200
add=number1+number2 # 100+200 =300
print("the addition of {} and {} is {}".format(number1,number2,add))

#----- case-2-----
number1=input("enter a number1:") # step-1: it will ask the number1='100'
number2=input("enter a number2:") # step-2: it will ask the number2='200'
add=number1+number2 # '100'+ '200'='100200'
print("the addition of {} and {} is {}".format(number1,number2,add))

#----- case-3(type casting)-----
number1=input("enter a number1:") # step-1: it will ask the number1='100'
number2=input("enter a number2:") # step-2: it will ask the number2='200'
add=int(number1)+int(number2) # int('100')+int('200')=100+200
print("the addition of {} and {} is {}".format(number1,number2,add))
```

```
In [11]: #----- case-4(type casting)-----
number1=int(input("enter a number1:")) # step-1: it will ask the number1=in
number2=int(input("enter a number2:")) # step-2: it will ask the number2=in
add=number1+number2 # 100+200=300
print("the addition of {} and {} is {}".format(number1,number2,add))
```

```
enter a number1:100
enter a number2:200
the addition of 100 and 200 is 300
```

```
In [1]: number1=100.5 # type = float
number2=200 # type= integer
add=number1+number2 # 100.5+200 =300.5
print("the addition of {} and {} is {}".format(number1,number2,add))
```

```
the addition of 100.5 and 200 is 300.5
```

```
In [4]: number1=float(input("enter a number1:"))
number1
```

```
enter a number1:100.5
```

```
Out[4]: 100.5
```

```
In [5]: number2=int(input("enter a number1:"))
        number2
```

enter a number1:100.5

```
-----
-
ValueError                                Traceback (most recent call las
t)
Cell In[5], line 1
----> 1 number2=int(input("enter a number1:"))
      2 number2

ValueError: invalid literal for int() with base 10: '100.5'
```

```
In [8]: float('100.5555')
```

Out[8]: 100.5555

```
In [22]: n1=int(input("enter n1:")) # this will fail if we do not provide integer va
        n2=float(input("enter n2:"))
        n1+n2
```

```
# we need to keep in mind
# always if i want to provide integer value type cast: int
#         if i want to provide float value type cast: float
#         if i provide by mistake int('100.5') ===== error
#         to avoid this we go for eval concept
```

enter n1:100.5

```
-----
-
ValueError                                Traceback (most recent call las
t)
Cell In[22], line 1
----> 1 n1=int(input("enter n1:"))
      2 n2=float(input("enter n2:"))
      3 n1+n2

ValueError: invalid literal for int() with base 10: '100.5'
```

eval

evaluate

```
In [23]: n1=eval(input("enter n1:"))
        n2=eval(input("enter n2:"))
        print("the addition of {} and {} is {}".format(n1,n2,n1+n2))
```

enter n1:100.5
enter n2:100
the addition of 100.5 and 100 is 200.5

```
In [24]: n1=input("enter n1:") # step-1: n1='100.5'
n2=input("enter n2:") # step-2: n2='200'
print("the addition of {} and {} is {}".format(n1,n2,eval(n1)+eval(n2)))
```

enter n1:100.5
enter n2:200
the addition of 100.5 and 200 is 300.5

```
In [ ]: n1=input("enter n1:") #n1='100.5'
n2=input("enter n2:") #n2='200'
eval(n1+n2) # ('100.5'+ '200'='100.5200')====>
eval(n1)+eval(n2)# eval('100.5')+eval('200')
# 100.5+200=300.5
```

```
In [25]: n1=input("enter n1:") #n1='100.5'
n2=input("enter n2:") #n2='200'
eval(n1+n2)
```

enter n1:100
enter n2:200

Out[25]: 100200

```
In [26]: n1=input("enter n1:") #n1='100.5'
n2=input("enter n2:") #n2='200'
eval(n1)+eval(n2)
```

enter n1:100
enter n2:200

Out[26]: 300

Q) wap ask the user enter two numbers

- n1= from user 100
- n2= from keyboard 200
- the addition of 100 and 200 is 300
- the subtraction of 100 and 200 is -100
- the multiplication of 100 and 200 is 20000

```
In [31]: num1=eval(input("enter n1:")) # step-1: int('100.5') ===== error
num2=eval(input("enter n2:"))
ans= num1+num2
sub = num1-num2
mul= num1*num2
print("the addition of {} and {} is : {}".format(num1,num2,ans))
print("the subtraction of {} and {} is : {}".format(num1,num2,sub))
print("the multiplication of {} and {} is : {}".format(num1,num2,mul))
```

enter n1:100
enter n2:200.5
the addition of 100 and 200.5 is : 300.5
the subtraction of 100 and 200.5 is : -100.5
the multiplication of 100 and 200.5 is : 20050.0

WAP ask the key board enter 3 numbers

- and find the average
- if a,b,c are three numbers then average is $(a+b+c)/3$

In [32]: `sum(1,2,3)`

```
-----
-
TypeError                                Traceback (most recent call last)
Cell In[32], line 1
----> 1 sum(1,2,3)

TypeError: sum() takes at most 2 arguments (3 given)
```

In [38]: `n1 = eval(input("enter first number:"))
n2 = eval(input("enter second number:"))
n3 = eval(input("enter third number:"))
average=(n1+n2+n3)/3
final_avg=round(average,2)
print("the original value before round off:",average)
print("the average of {},{} and {} is {}".format(n1,n2,n3,final_avg))`

```
enter first number:2
enter second number:3
enter third number:5
the original value before round off: 3.3333333333333335
the average of 2,3 and 5 is 3.33
```

In []: `12/2 # division
12%2 # modulus
12//2 # floor division`

division

- will provide normal division values

In [45]: `12/5 # why 2.4`

Out[45]: 2.4

modulus

- it will provide reminder

In [46]: `12%5 # why 2`

Out[46]: 2

floor division:

- quotient

In [47]: 12//5 # why 2

Out[47]: 2

Q)WAP ask the user to enter

- enter bill amount
- enter tip amount
- find total amount

```
In [48]: # WAP ask the user to enter
# bill amount
bill_amount=eval(input("enter the bill:"))
# tip amount
tip_amount=eval(input("enter the tip:"))
# total amount
total_amount=bill_amount+tip_amount
print("The total amount is:",total_amount)
```

```
enter the bill:1000
enter the tip:100
The total amount is: 1100
```

Q)WAP ask the user to enter

- enter bill amount
- enter tip percent
- find total amount

```
In [49]: # Wap ask the user enter
# bill amount    1000
# tip percentage on the bill (10% tip on the bill amount)
# how much percentage of tip want to give:10
# total amount=?

bill_amount=eval(input("enter bill amount:"))
tip_per=eval(input("enter tip percentage:"))
tip_amount=(bill_amount*tip_per)/100
total=bill_amount+tip_amount
print("total bill is:",total)
```

```
enter bill amount:1000
enter tip percentage:20
total bill is: 1200.0
```

Q)WAP ask the user enter Dollars

- input: how many dollars you want to pay : 100
- Manager will say: print('no dollars are not accepted')
- You: print("then what will accepted")
- Manager: print("only indian ruppes ")
- input: one dollar equal to how many ruppes : 80
- then total amount to pay: 100*80

```
In [52]: # WAP ask the user enter Dollars
# input: how many dollars you want to pay : 100
# print('no dollars are not accepted')
# print("then what will accepted")
# print("only indian ruppes ")
# input: one dollar equal to how many ruppes : 80
# then total amount to pay: 100*80

n1 = eval(input("How many dollars you want to pay"))
print('Manger: No dollars are not accepted')
print('Rashaad: Then what will accepted')
print('Manager: Only indian rupees')
n2 = eval(input("one dollar equal to how many indian rupees"))
amount=n1*n2
print("The total amount to pay {}".format(amount))
print("Manager: Thank you Rashaad")
```

```
How many dollars you want to pay100
Manger: No dollars are not accepted
Rashaad: Then what will accepted
Manager: Only indian rupees
one dollar equal to how many indian rupees80
The total amount to pay 8000
Manager: Thank you Rashaad
```

end

```
In [3]: # if we want to combine two print statements into one statement
print("hai",end='---')
print("hello")

# hai hello
```

```
hai---hello
```

sep

sepeartor

```
In [4]: print('hai','hello')
# how many words are there : 2
# you want to seperate hai and hello with
# and

# o/p: hai & hello
```

```
hai hello
```

```
In [5]: print('hai','hello',sep='&')
```

```
hai&hello
```

```
In [7]: print("hello",'python','excited',sep='--->')
```

```
hello--->python--->excited
```

```
In [8]: print('12','10','2023',sep='-')
```

12-10-2023

```
In [10]: print(3,5,sep='***')
```

3***5

Q)WAP ask the user enter a distance in km

- ask the user enter a fare in rs
- calculate total amount to pay

output:

- for 2kms the fare is 20rs.

```
In [ ]: # WAP ask the user enter a distance in km
#       ask the user enter a fare in rs
#       calculate total amount to pay

# output:
# for 2kms the fare is 20rs.
```

```
In [ ]: distance=eval(input("enter a distance:"))
fare=eval(input("enter fare in rs:"))
total=distance*fare
print("the total amout is:",total)
```

```
In [ ]: dist = eval(input("enter the Distance in Kilometeres"))
fare = eval(input("enter the fair in rupees"))
tot_amnt = dist*fare
print("For {} Kilometeres the fair is {}".format(dist,tot_amnt))
```

```
In [11]: dist = eval(input("enter the Distance in Kilometeres "))
fare = eval(input("enter the fair in rupees "))
tot_amnt = dist*fare
print("For {} Kilometeres the fair is {} rs.".format(dist,tot_amnt))
```

enter the Distance in Kilometeres 2
enter the fair in rupees 10
For 2 Kilometeres the fair is 20 rs.

Q) Implement above code using seperator operator

- For {} Kilometeres the fair is {} rs".format(dist,tot_amnt),','

```
In [ ]: st1:For 2 Kilometeres the fair is 20 rs,
st2:.
combine st1 and st2 using sep operators
```



```
In [3]: dist = eval(input("enter the Distance in Kilometeres "))
fare = eval(input("enter the fair in rupees "))
tot_amnt = dist*fare
print("For {} Kilometeres the fair is {} rs".format(dist,tot_amnt),
      '.',sep='')

# the dot is too far
```

```
enter the Distance in Kilometeres 2
enter the fair in rupees 10
For 2 Kilometeres the fair is 20 rs.
```

```
In [1]: print('a','b',sep='&')
```

```
a&b
```

```
In [5]: dist = eval(input("enter the Distance in Kilometeres "))
fare = eval(input("enter the fair in rupees "))
tot_amnt = dist*fare
print("For {} Kilometeres the fair is {} rs".format(dist,tot_amnt),
      '.',sep='')

# the dot is too far
```

```
enter the Distance in Kilometeres 2
enter the fair in rupees 10
For 2 Kilometeres the fair is 20 rs.
```

```
In [7]: print('Ramesh','35',sep=' & ') #
```

```
# Ramesh & 35
```

```
Ramesh & 35
```

```
In [8]: print('Ramesh','35',sep=' age is ')
```

```
Ramesh age is 35
```