```
print("hello world")
```

```
hello world
```

```
print(2+3)
```

```
5
```

```
print(2-3)
```

```
-1
```

```
print(2*3)
```

```
6
```

```
print(2/3)
```

```
0.6666666666666666
```

```
a=20
b=30
print(a-b)
```

```
-10
```

```
a=20
b=30
c=10
print(a+b+c)
```

```
60
```

```
a=20
b=30
print(a+b+z)
c=50
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/ipython-input-1156438634.py in <cell line: 0>()
      1 a=20
      2 b=30
----> 3 print(a+b+z)
      4 c=50

NameError: name 'z' is not defined
```

```
a=10
b=20
print(a+b)
```

```
30
```

```
a=20
b=30
print(a*b)
```

```
600
```

```
a=10
a=20
print(a+a)
```

```
40
```

```
5*2+6*2-1
```

```
21
```

```
a="hello"
b="world"
print(a+b)
```

```
helloworld
```

```
a="hello"
b=30
print(a*b)
```

```
hellohellohellohellohellohellohellohellohellohellohellohellohellohellohelloh
```

```
name="python"
print("*"*3+name+"*"*3)
```

```
***python***
```

```
a="shahin"
b=len(a)
print(b)
```

```
6
```

```
a="prajwala"
b=len(a)
print(b)
```

```
8
```

```
a=input()
print(a)
```

```
prajwala
prajwala
```

```
a=int(input())
print(a)
```

```
65
65
```

```
name=input()
print("*"*3+name+"*"*3)
```

```
python
***python***
```

```
number =int(input())
print(number)
```

```
25
25
```

```
float_value=float(input())
print(float_value)
```

```
15.7
15.7
```

```
float_value=float(input("Enter a float value:"))
print(float_value)
```

```
Enter a float value:15.7
15.7
```

```
a=input()
b=len(a)
print("*"*b+a+"*"*b)
```

```
kle
****kle ****
```

```
name="python"
print("*"*3+name+"*"*3)
```

```
***python***
```

```
name=input()
print("*"*3+name+"*"*3)
```

```
108
***108***
```

```
a="milk"
b="shake"
print(a+b)
```

```
milkshake
```

```
a="milk"
b="shake"
print(a+" "+b)
```

```
milk shake
```

```
a=" milk shake "
b=5
print(a*b)
```

```
 milk shake  milk shake  milk shake  milk shake  milk shake
```

```
a=" kle bca "
b=7
print(a*b)
```

```
 kle bca  kle bca  kle bca  kle bca  kle bca  kle bca  kle bca
```

```
a="robert"
b="21"
print(a+" is "+ b + " year old")
```

```
robert is 21 year old
```

```
a="24"
b=10
c=b+int(a)
print(c)
```

```
34
```

```
a="24"
b=int(a)
print(type(a))
print(type(b))
```

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

```
first_name=input()
number=int(input())
print(first_name+" "+"is"+" "+str(number*2)+" year old")
```

```
laxmiiiiiiiiiiiiiiiii
24
laxmiiiiiiiiiiiiiiiii is 50 year old
```

```
first_number= input()
second_number=input()
print("understand the type conversion")
print(int(first_number)+int(second_number))
print(first_number+second_number)
print(float(first_number)+float(second_number))
```

```
12
15
understand the type conversion
27
1215
27.0
```

```
name="kle bca"
print(name[4:7])
```

```
bca
```

```
name="kle bca"
print(name[4:7])
```

```
print(name[:7])
print(name[4:])
```

```
bca
kle bca
bca
```

```
name="kle bca engineering"
print(name[8:19])
```

```
engineering
```

```
name="kle chikodi bca"
print(name[4:12])
```

```
chikodi
```

```
name="laxmii"
print(name[0])
print(name[1:5])
print(name[2:5])
print(name[3:5])
print(name[4:5])
print(name[5:5])
```

```
l
axmi
xmi
mi
i
```

```
a=21
print(a[1]+a[0])
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-132791941.py in <cell line: 0>()
      1 a=21
----> 2 print(a[1]+a[0])

TypeError: 'int' object is not subscriptable
```

```
first_name=input()
```

```
happy coding
```

```
number=12345
c_in_string=str(number)
print(c_in_string[0])
print(c_in_string[-1])
```

```
1
5
```

```
number=12345
c_in_string=str(number)
first_char=c_in_string[0]
last_char=c_in_string[-1]
print(int(first_char)+int(last_char))
```

```
6
```

```
name=input()
print(len(name))
```

```
software
8
```

```
name=input()
print(len(name))
```

```
qwerty
6
```

```
name=input()
print(name[-1])
```

```
python
n
```

```
name=input()
b=name[0:3]
print(b)
```

```
kle collge
kle
```

```
number1=int(input())
number2=int(input())
total=2*(number1+number2)
print(total)
```

```
5
4
```

```
18
```

```
number1=int(input())
number2=int(input())
total=2*(number1+number2)
print(total)
```

```
4
2
12
```

```
length=int(input())
breadth=int(input())
total=2*(length+breadth)
print("perimeter of rectangle is:"+str(total))
```

```
2
4
perimeter of rectangle is:12
```

```
print("$"*1)
print("$"*2)
print("$"*3)
print("$"*4)
```

```
$
$$
$$$
$$$$
```

```
a=(-1)
b=(-2)
print(a+b)
```

```
-3
```

```
a=input()
b=a[-1]
c=a[-2]
e=int(b)
f=int(c)
print(e+f)
```

```
python28
10
```

```
#variables
a=15
b=4

#addition
print("addition:",a+b)
```

```
#substraction
print("substraction:",a-b)

#multiplication
print("multiplication:",a*b)

#division
print("division:",a/b)

#floor division
print("floor division:",a//b)

#modulus
print("modulus:",a%b)

#exponentiation
print("exponentiation:",a**b)
```

```
addition: 19
substraction: 11
multiplication: 60
division: 3.75
floor division: 3
modulus: 3
exponentiation: 50625
```

```
a=13
b=33

print(a>b)
print(a<b)
print(a==b)
print(a!=b)
print(a>=b)
print(a<=b)
```

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

```
a=True
b=False
print(a and b)
print(a or b)
print(not a)
```

```
False
True
False
```

```
print(123=="123")
```

```
False
```

```
print(4==5)
```

```
False
```

```
print(1*2*3==6)
```

```
True
```

```
print((a*b)>130 and(a*b)<130)
```

```
False
```

```
print(2*3-4+3*6-4==34)
```

```
False
```

```
m=int(input())
p=int(input())
c=int(input())
part1= (m>=60) and (p>=50) and (c>=45) and (m+p+c>=180)
part2=m+p>=120 or c+p>=110
print(part1 or part2)
```

```
10
20
30
False
```

```
m=int(input())
p=int(input())
c=int(input())
part1= (m>=60) and (p>=50) and (c>=45) and (m+p+c>=180)
part2=m+p>=120 or c+p>=110
print(part1 or part2)
```

```
50
60
70
True
```

```
first_name=input()
last_name=input()
last_three_1=first_name[-3:]
last_three_2=last_name[-3:]
print(last_three_1 == last_three_2)
```

```
python
python
True
```

```
a=laxmi
b=laxmi
if a==b
  print("the condition is true")
```

```
  File "/tmp/ipython-input-2573620965.py", line 3
    if a==b
           ^
SyntaxError: expected ':'
```

```
if true:
  print("the condition is true")
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/ipython-input-2698209680.py in <cell line: 0>()
----> 1 if true:
      2   print("the condition is true")

NameError: name 'true' is not defined
```

```
a=input()
b=input()
if len(a)==len(b):
    print
```

```
a=int(input())
if number>0:
  print("my number is even")
print("program ended")
```

```
2
my number is even
program ended
```

```
a=int(input())
if number>0:
  print("the number is positive")
print("program ended")
```

```
8
the number is positive
program ended
```

```
password= input("enter passward")

if password == "program":
  print("correct password")
```

```
enter passwardprogram
correct password
```

```
a = int(input())
if a>0:
  print("positive")
else:
  print("not positive")
print("end")
```

```
-60
not positive
end
```

```
password = input("enter pasword: ")

if password == "program":
  print("correct password")
else:
  print("not matching")
print("program ended")
```

```
enter pasword: progam
not matching
program ended
```

```
number_1 = int(input("enter the integer"))
number_2 = int(input("enter the integer:"))
if number_1>number_2:
  print(number_1)
else:
  print(number_2)
```

```
enter the integer10
enter the integer:25
25
```

```
marks = int(input("enter your marks:"))

if marks >=40:
  print("pass")
else:
  print("fail")
```

```
enter your marks:41
pass
```

```
age = int(input("enter your age:"))

if age >=18:
  print("you are eligible to vote")
else:
  print("not eligible for vote")
```

```
enter your age:17
not eligible for vote
```

```
number = int(input("enter a number:  "))

if number % 2 ==0:
  print("even number")
else:
  print("odd number")
```

```
enter a number:  57
odd number
```

```
charecter_1 = input()
charecter_2 = input()

if(charecter_1[0:3]==charecter_2[0:3]):
  print("matching")
else:
  print("not matching")
```

```
python
phyton
not matching
```

```
charecter_1 = input()
charecter_2 = input()

if(charecter_1[0]==charecter_2[-1]):
  print("matching")
else:
  print("not matching")
```

```
program
PrograM
not matching
```

```
charecter_1 = input()
charecter_2 = input()

if(charecter_1[0]==charecter_2[-1]):
  print("matching")
else:
  print("not matching")
```

```
AI
ai
not matching
```

```
size = input()
num=int(input())
if(size=="large" and num>300):
  print("buy a book")
else:
  print("do not buy a book")
```

```
large
290
do not buy a book
```

```
size = input()
num=int(input())
if(size=="large" and num>300):
  print("buy a book")
else:
  print("do not buy a book")
```

```
large
301
buy a book
```

```
a=int(input())
b=int(input())
if((a>300 or b<300) and a+b<500):
  print("can team up")
else:
  print("cannot team up")
```

```
308
108
can team up
```

```
a=int(input())
b=int(input())
if((a<500 or b>500) and a+b<1000):
  print("pair")
else:
  print('not a pair')
```

```
300
550
pair
```

```
a=int(input())
b=int(input())
if((a<=1000 or b<=1000) or b<500):
  print("pair")
```

```
else:
  print('not a pair')
```

```
500
600
pair
```

```
a=int(input())
if("number > 10"):
  print("n+5")
else:
  print("n+1")
```

```
9
n+5
```

```
a=int(input())
if(a<=10):
  print(a+1)
else:
  print(a+5)
```

```
11
16
```

```
first=int(input())
second=int(input())
third=int(input())
if(first+second+third==180):
  print("*")
  print("*"*2)
  print("*"*3)
else:
  print("not a valid triangle")
```

```
90
20
70
*
**
***
```

```
a=int(input())
b=int(input())


sum=a+b
if ((a<20 and b<20) or (sum<20) ):
  print(sum)
else:
  print(a)
  print(b)
```

```
18
15
```

```
33
```

```
a=int(input())
b=int(input())


sum=a+b
if ((a<20 and b<20) or (30<sum<20) ):
  print(sum)
else:
  print(a)
  print(b)
```

```
22
25
22
25
```

```
a=input()
b=len(a)
if(2<b<7 or a[0]!="a" ):
  print("valid string")
else:
  print("not a valid string")
```

```
5
valid string
```

```
if True:
  print("Block 1")
  if True:
   print("block 2")
  print("block 3")
print("Block 4")
```

```
Block 1
block 2
block 3
Block 4
```

```
matches_won = int(input())
goals = int(input())
if matches_won > 8:
  if goals > 20:
    print("hurray")
  print("winner")
print("program ended")
```

```
11
23
hurray
winner
program ended
```

```
a=2
b=3
c=1
is_a_greatest = (a > b) and (a > c)
if is_a_greatest:
  print(a)
else:
  is_b_greatest = (b > c)
  if is_b_greatest:
    print(b)
  else:
    print(c)
```

```
3
```

```
n=int(input("enter a number: "))

if n > 0:
  print("positive")
elif n < 0:
  print("negetive")
else:
  print("zero")
```

```
enter a number: 11
positive
```

```
number=5
is_divisible_by_10=(number%10==0)
is_divisible_by_5=(number%5==0)
if is_divisible_by_10:
  print("divisible by 10")
elif is_divisible_by_5:
  print("divisible by 5")
else:
  print("not divisible by 10 or 5")
```

```
divisible by 5
```

```
marks= int(input("Enter your marks: "))

if marks >=90:
  print("grade: A")
elif marks >= 75:
  print("grade: B")
elif marks >=60:
  print("grade: C")
elif marks >=40:
  print("grade: D")
else:
  print("grade:f(fail)")
```

```
Enter your marks: 56
grade: D
```

```
age = int(input("enter your age: "))

if age <13:
  print("you are a child")
elif age < 20:
  print("you are a teenager")
elif age < 60:
  print("you are an adult")
elif age >=60:
  print("you are a senior citizen")
```

```
enter your age: 17
you are a teenager
```

```
number=int(input("enter a number: "))
if number <=3:
  print("not polygon")
elif number==3:
  print("triangle")
elif number==4:
  print ("rectangle")
elif number ==5:
  print("round")
else:
  print("polygon")
```

```
enter a number: 8
polygon
```

```
n=float(input())
if (n>85):
  print("A")
elif(n>70):
  print("B")
elif(n>=60):
  print("C")
else:
  print("F")
```

```
82
B
```

```
A=int(input())
B=int(input())
C=int(input())
if(A>B and A>C):
  print(A)
elif(B>A and B>C):
  print(B)
else:
  print(C)
```

```
10
23
2
23
```

```
N=int(input("enter how many people:"))
X=int(input("enter how many pieces per head:"))
total_pices=N*X
if total_pices%4==0:
  print(int(total_pices/4))
else:
  print((total_pices/4)+1)
```

```
enter how many people:50
enter how many pieces per head:2
25
```

```
N=int(input("enter how many people:"))
X=int(input("enter how many price per subscription:"))

if N%6==0:
  print(int(N/6) *X)
else:
  print((int(N/6)+1) *x )
```

```
enter how many people:30
enter how many price per subscription:200
1000
```

```
R=int(input())
if(R<=3):
  print("one of top 3")
else:
  print("not top 3 but one of top 10")
```

```
18
not top 3 but one of top 10
```

```
R=int(input())
if(R<=3):
  print("one of top 3")
else:
  print("not top 3 but one of top 10")
```

```
2
one of top 3
```

```
B1=int(input())
B2=int(input())
B3=int(input())

if(B1+B2+B3 >=2):
  print()
```

```
3
4
5
```

```
bill_amount=int(input())
if bill_amount<50:
  bill=2*bill_amounnt
elif bill_amount<150:
  bill=(2*50)+(3*(bill_amount-50))
elif bill_amount<250:
  bill=(2*50)+(3*100)+(5*(bill_amount-150))
elif bill_amount>=250:
  bill=(2*50)+(3*100)+(5*100)+(8*(bill_amount-250))

print(bill+bill*0.2)
```

```
220
900.0
```

```
cp=int(input())
sp=int(input())

if sp>cp:
  print("profit")
elif sp<cp:
  print("loss")
else:
  print("no profit-no loss")
```

```
20
10
loss
```

```
T=float(input())
if T<0:
  print("freezing weather")
elif 0<=T<10:
  print("very cold weather")
elif(10<=T<20):
  print("cold weather")
elif(20<=T<30):
  print("normal")
elif(30<=T<40):
  print("hot")
else:
  print("very hot")
```

```
25
normal
```

```
a=int(input())
counter=0
while counter < 3:
```

```
  a=a+1
  print(a)
  counter=counter+1
```

```
2
3
4
5
```

```
a=int(input())
while counter < 3:
  a=a+1
  print(a)
  counter=counter+1
print("end")
```

```
4
end
```

```
#missing initialization
a=int(input())
while counter < 3:
  a=a+1
  print(a)
  counter=counter+1
print("end")
```

```
2
end
```

```
number=int(input())
counter=number+0
while(counter<10+number):
  counter=counter+1
  print(counter)
```

```
5
6
7
8
9
10
11
12
13
14
15
```

```
a=3
c=0
sum=0
while (c<a):
  number=int(input("enter the number: "))
  sum=sum+number
```

```
  c=c+1
print(sum)
```

```
enter the number: 10
enter the number: 20
enter the number: 30
60
```

```
a=3
c=0
while(c<a):
  number=int(input("enter the number: "))
  print(number)
  c=c+1
```

```
enter the number: 10
10
enter the number: 20
20
enter the number: 30
30
```

```
a=int(input())
c=0
sum=0
while(c<a):
  c=c+1
  sum=sum+c

print(sum/c)
```

```
10
5.5
```

```
 a="python"
 b=len(a)
print("*"*b+a+"*"*b)
```

```
******python******
```

```
a=input()
b=len(a)
print(a(c))
c=c+1
```

```
8
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-1211354379.py in <cell line: 0>()
      1 a=input()
      2 b=len(a)
----> 3 print(a(c))
      4 c=c+1

TypeError: 'str' object is not callable
```

```
number=int(input())
Q=number/4
convert_int=int(Q)

if(number%4==0):
   print("good")
else:
  print("not good")
```

```
16
good
```

```
word= "python"
for each_char in word:
  print(each_char)
```

```
p
y
t
h
o
n
```

```
word = "program"
for each_char in word:
  print(each_char)
```

```
p
r
o
g
r
a
m
```

```
for number in range(9):
    print (number)
```

```
0
1
2
3
4
5
6
7
8
```

```
name="manu is a good girl "
for i in range(0,4):
  print(name[i],end="")
```

```
manu
```

```
for i in range(1,20):
  print(i**2)
```

```
1
4
9
16
25
36
49
64
81
100
121
144
169
196
225
256
289
324
361
```

```
sum=0
for i in range(1,13):
  sum=sum+(i**2)
print(sum)
```

```
650
```

```
sum=0
for i in range(1,6):
  sum=sum+(i**n)
print(sum)
```

```
22204663316663778144114175563592293 79
```

```
n=int(input())
sum=0
for i in range(1,n):
  sum=sum+(i)
print(sum)
```

```
30
435
```

```
a=3
c=0
sum=0
for i in range(1,n):
    sum=sum+(i)
print(sum)
```

```
435
```

```
a=int(input())
c=0
sum=0
for i in range(a):
  number=int(input("enter the number: "))
  sum=sum+number

print(sum)
```

```
3
enter the number: 12
enter the number: 15
enter the number: 19
46
```

```
a=int(input())
b=int(input())
for i in range(a,b+1):
  print(i,end=" ")
```

```
10
20
10 11 12 13 14 15 16 17 18 19 20
```

```
n = int(input("enter the number"))
for i in range(1,n+1):
  print("*"*i)
```

```
enter the number5
*
**
***
****
*****
```

```
n=5
for i in range(n,0,-1):
  print("*"*i)
```

```
*****
****
***
**
*
```

```
a=int(input())
sum=0
for i in range(a):
  number=int(input("enter the number: "))
  sum=sum+number

print(sum)
```

```
6
enter the number: 6
enter the number: 12
enter the number: 18
enter the number: 24
enter the number: 30
enter the number: 36
126
```

```
a=int(input())

sum=1
for i in range(a):
  number=int(input("enter the number: "))
  sum=sum*number

print(sum)
```

```
2
enter the number: 12
enter the number: 13
156
```

```
name=input()
find_char=input()
count=0
for i in range(len(name)):
  if name[i]==find_char:
    count=count+1
if count>0:
  print("the char in the string")
else:
  print ("the char not in the string")
```

```
python
o
the char in the string
```

```
number = int(input())
for i in range(1,number+1):
  print(str(i)*i)
```

```
5
1
22
333
4444
55555
```

```
start=int(input())
end=int(input())
sum=0
for i in range(start,end+1):
  sum=sum+1
  print(sum)
```

```
5
10
1
2
3
4
5
6
```

```
number=int(input())
for i in range(1,number+1):
  if i==number:
    print("+"*i)
  else:
    print("*"*i)
```

```
5
*
**
***
****
+++++
```

```
number = int(input())
for i in range(1,number+1):
  print(str(i)*i)
```

```
for i in range(1,number+1):
  print(str(i)*i)
```

```
4
1
22
333
4444
1
22
333
4444
```

```
n = int(input("enter the number"))
for i in range(1,n+1):
  print("*"*i)

for i in range(1,n+1):
  print("*"*i)
```

```
enter the number5
*
**
***
****
*****
*
**
***
****
*****
```

```
secret_message = "-R-a-v-i-"
print(secret_message[1:8:2])
```

```
Ravi
```

```
a= "waterfall"
part = a[1:6:3]
print(part)
```

```
ar
```

```
is_digit="4756".isdigit()
print(is_digit)
```

```
True
```

```
mobile = "9876543210"
print(len(mobile))
mobile = mobile.strip()
```

```
print(len(mobile))
print(mobile)
```

```
10
10
9876543210
```

```
name = "ravi."
name = name.strip(".")
print(name)
```

```
ravi
```

```
name="RAVI"
print(name.lower())
```

```
ravi
```

```
name="ravi"
print(name.upper())
```

```
RAVI
```

```
url="ananya is a well_behaved girl"
is_secure_url=url.startswith("ananya")
print(is_secure_url)
```

```
True
```

```
gmail_id="kle123@gmail.com"
is_gmail=gmail_id.endswith("@gmail.com")
print(is_gmail)
```

```
True
```

```
sentence="teh cat and teh dog"
sentence=sentence.replace("teh","the")
print(sentence)
```

```
the cat and the dog
```

```
for i in range(2):
  print("outer:"+str(i))
  for j in range(2):
    print("inner:"+str(j))
```

```
outer:0
inner:0
inner:1
outer:1
inner:0
```

```
inner:1
```

```
for i in range(2):
  print("outer:"+str(i))
  for j in range(1,6):
    print(j)
```

```
outer:0
1
2
3
4
5
outer:1
1
2
3
4
5
```

```
for i in range(2):
  for j in range(1,4):
    print("*"*j)
```

```
*
**
***
*
**
***
```

```
a=int(input())
b=int(input())
for i in range(2):
    for i in range(1,4):
      print(str(j)*i)
```

```
1
2
3
33
333
3
33
333
```

```
rows=int(input("enter the rows : "))
columns=int(input("enter the columns: "))
```

```
for i in range(1,4):
  print("for loop i = ",i)


  j=1
  while j <=2:
    print("while loop j =",j)
    j=j+1
```

```
for loop i =  1
while loop j = 1
while loop j = 2
for loop i =  2
while loop j = 1
while loop j = 2
for loop i =  3
while loop j = 1
while loop j = 2
```

```
i=1
while i <=3:
  print("outer while i = ",i)


  j=1
  while j <=2:
    print("inner while j = ",j)
    j += 1


  i+=1
```

```
outer while i =  1
inner while j =  1
inner while j =  2
outer while i =  2
inner while j =  1
inner while j =  2
outer while i =  3
inner while j =  1
inner while j =  2
```

```
rows=int(input("enter the rows: "))
columns=int(input("enter columns: "))
for i in range(rows):#columns
  new_string=" "
  for j in range(1,columns+1): #rows
   new_string=new_string+str(j)+" "
  print(new_string)
```

```
enter the rows: 5
enter columns: 5
 1 2 3 4 5
 1 2 3 4 5
 1 2 3 4 5
 1 2 3 4 5
```

```
 1 2 3 4 5
```

```
rows=int(input("enter the rows: "))
columns=int(input("enter columns: "))
for i in range(rows):#columns
  new_string=" "
  for j in range(1,i+1): #rows
   new_string=new_string+str(j)+" "
  print(new_string)
```

```
enter the rows: 5
enter columns: 5

 1
 1 2
 1 2 3
 1 2 3 4
```

```
# print numbers 1 to 5 using while with break
num = 1

while num<10:        #infinite loop
  print(num)
  num = num + 1


  if num > 5:  #stop condition
    break      #exit loop
print("program ended")
```

```
1
2
3
4
5
program ended
```

```
# print numbers 1 to 5 using while with break
num = 1

while num<10:        #infinite loop
  print(num)
  num = num + 1


  if num > 5:  #stop condition
    continue      #exit loop
print("program ended")
```

```
1
2
3
4
5
6
```

```
7
8
9
program ended
```

```
for i in range(5):
  if i ==3:
    continue
  print(i)
print("END")
```

```
0
1
2
4
END
```

```
for i in range(5):
  if i ==3:
    break
  print(i)
print("END")
```

```
0
1
2
END
```

```
num = 0

if num > 0:
  print("the number is positive")
elif num == 0:
  pass  # do nothing if the number is zero
else:
  print("the number is negative")
```

```
num = 4

if num > 0:
  print("the number is positive")
elif num == 0:
  pass  # do nothing if the number is zero
else:
  print("the number is negative")
```

```
the number is positive
```

```
num = -4

if num > 0:
  print("the number is positive")
elif num == 0:
  pass  # do nothing if the number is zero
```

```
else:
  print("the number is negative")
```

```
the number is negative
```

```
number_range=int(input())
sum=0
for i in range(1,number_range):
  summ=sum+1


  if sum>15:
    break
print("program ended")
```

```
8
program ended
```

```
number_range=int(input())
sum=0
for i in range(1,number_range):
  summ=sum+1


  if sum>15:
    continue
print("program ended")
```

```
16
program ended
```

```
#factorial number
number=int(input())
factorial=1
for i in range(1,number+1):
  factorial=factorial*i
print(factorial)
```

```
5
120
```

```
a=chr(65)
print(a)
```

```
A
```

```
name="python"
for i in name:
  print(i)
```

```
p
y
t
h
```

```
o
n
```

```
a=ord("A")
print(a)
```

```
65
```

```
name="python"
sum=0
for i in name:
  value=ord(i)
  sum=sum+value
print(sum)
```

```
674
```

```
name="12345python"
b=name[5:]
sum=0
for i in b:
  value=ord(i)
  sum=sum+value
print(sum)
```

```
674
```

```
name="12345python"
b=name[5:]
sum=0
for i in b:
  value=ord(i)
  sum=sum+value
  if sum>200:
    break
print(sum)
```

```
233
```

```
list_a = [5, "six", 2, 8.2]
print(list_a)
```

```
[5, 'six', 2, 8.2]
```

```
a = 2
list_a = [5, "six", a, 8.2]
print(type(list_a))
print(list_a)
```

```
<class 'list'>
[5, 'six', 2, 8.2]
```

```
a = 2
list_a = [5, "six", a, 8.2]
list_b = [1, list_a]
print(list_b)
```

```
[1, [5, 'six', 2, 8.2]]
```

```
a = 2
list_a = [5, "six", a, 8.2]
print(len(list_a))
```

```
4
```

```
a = 2
list_a = [5, "six", a, 8.2]
print(list_a[1])
```

```
six
```

```
a=2
list_a=[5, "six", a, 8.2]
for item in list_a:
  print(item)
```

```
5
six
2
8.2
```

```
a=2
list_a=[5, "six", a, 8.2]
for item in range(len(list_a)):
  print(list_a[item])
```

```
5
six
2
8.2
```

```
list_a=[1, 2, 3]
list_b=["a", "b", "c"]
list_c=list_a+list_b
print(list_c)
```

```
[1, 2, 3, 'a', 'b', 'c']
```

```
list_a = []
print(list_a)
for i in range(1,4):
  list_a = list_a+[i]
print(list_a)
```

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

```
list_a = [4]
print(list_a)
for i in range(1,4):
  list_a = list_a+[i]
print(list_a)
```

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

```
list_a = [1,2]
list_b = list_a*3
print(list_b)
```

```
[1, 2, 1, 2, 1, 2]
```

```
list_a = ["R", "B", "G", "O", "W"]
list_b = list_a[0::3]
print(list_b)
```

```
['R', 'O']
```

```
list_a = ["R", "B", "G", "O", "W"]
list_b = list_a[1:5]
print(list_b)
```

```
['B', 'G', 'O', 'W']
```

```
name="kle college"
convert_list=list(name)
print(convert_list)
```

```
['k', 'l', 'e', ' ', 'c', 'o', 'l', 'l', 'e', 'g', 'e']
```

```
list_a = list(range(4))
print(list_a)
```

```
[0, 1, 2, 3]
```

```
list_a = [1, 2, 3, 5]
print(list_a)
list_a[3]=4
print(list_a)
```

```
[1, 2, 3, 5]
[1, 2, 3, 4]
```

```
message="sea you soon"
message[2]="e"
print(message)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-2704032270.py in <cell line: 0>()
      1 message="sea you soon"
----> 2 message[2]="e"
      3 print(message)

TypeError: 'str' object does not support item assignment
```

```
list_values=[1,2,"water",29,"water"]
find_string=input()
for i in list_values:
  if i==find_string:
    print("succesfully find the string")
    break

  else:
    pass
```

```
water
succesfully find the string
```

```
list_a=[10, 20, 40, 100]
b=int(input())
```

```
list_a=["A", "two", "M", "one", ]
print(list_a)
```

```
['A', 'two', 'M', 'one']
```

```
list_a = [5,"six", 2, 8.2]
list_b = [5,2,4,8]
list_c = (list_a+list_b)
print(list_c)
```

```
[5, 'six', 2, 8.2, 5, 2, 4, 8]
```

```
list_a=[1, 2, "two", 20]
list_b=list_a[0]
print(list_b)
```

```
1
```

```
number=int(input())
result=[]
for i in range(number):
```

```
  value=int(input())
  result=result+[value]
print(result)
```

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

```
number=int(input())
result=[]
for i in range(number):
  value=(input())
  result=result+[value]
print(result)
```

```
4
laxmi
anu
amu
mahi
['laxmi', 'anu', 'amu', 'mahi']
```

```
a=2
tuple_a=(5, "six", a, 8.2)
print(type(tuple_a))
print(tuple_a)
```

```
<class 'tuple'>
(5, 'six', 2, 8.2)
```

```
a=2
tuple_a=(5, "six", a, 8.2)
print(tuple_a[1])
```

```
six
```

```
tuple
```

```
color="red"
tuple_a=tuple(color)
print(tuple_a)
```

```
('r', 'e', 'd')
```

```
list_a=[1,2,3]
tuple_a=tuple(list_a)
print(tuple_a)
```

```
(1, 2, 3)
```

```
tuple_a=tuple(range(4))
print(tuple_a)
```

```
(0, 1, 2, 3)
```

```
number=int(input())
result=()
for i in range(number):
  value=input()
  result=result+(value,)
print(result)
```

```
4
anu
mahi
amu
laxmi
('anu', 'mahi', 'amu', 'laxmi')
```

```
tuple_a=(1, 2, 3, 4)
is_part=5 in tuple_a
print(is_part)
```

```
False
```

```
tuple_a=(1, 2, 3, 4)
is_part=2 in tuple_a
print(is_part)
```

```
True
```

```
tuple_a=(1,2,3,4)
is_part=1 not in tuple_a
print(is_part)
```

```
False
```

```
tuple_a=(1,2,3,4)
is_part=5 not in tuple_a
print(is_part)
```

```
True
```

```
list_a=[1,2,3,4]
is_part=2 not in list_a
print(is_part)
```

```
False
```

```
word="laxmi is a well-mannered girl"
part=input()
```

```
is_part=part in word
print(is_part)
```

```
girl
True
```

```
word="laxmi is a well-mannered girl"
part=input()
is_part=part in word
print(is_part)
```

```
anu
False
```

```
tuple_a=('R', 'e', 'd')
(s_1, s_2, s_3)=tuple_a
print(s_1)
print(s_2)
print(s_3)
```

```
R
e
d
```

```
tuple_a=('R', 'e', 'd')
(s_1, s_2,)=tuple_a
print(s_1)
print(s_2)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
/tmp/ipython-input-3934416236.py in <cell line: 0>()
      1 tuple_a=('R', 'e', 'd')
----> 2 (s_1, s_2,)=tuple_a
      3 print(s_1)
      4 print(s_2)

ValueError: too many values to unpack (expected 2)
```

```
tuple_a=('R', 'e', 'd')
(s_1, s_2, s_3, s_4)=tuple_a
print(s_1)
print(s_2)
print(s_3)
print(s_4)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
/tmp/ipython-input-1525336420.py in <cell line: 0>()
      1 tuple_a=('R', 'e', 'd')
----> 2 (s_1, s_2, s_3, s_4)=tuple_a
      3 print(s_1)
      4 print(s_2)
      5 print(s_3)

ValueError: not enough values to unpack (expected 4, got 3)
```

```
set_a={5, "six", 2, 8.2}
print(type(set_a))
print(set_a)
```

```
<class 'set'>
{8.2, 2, 5, 'six'}
```

```
set_a={"a", "d", "c", "d",1,3,2,2,2,1,3,}
print(set_a)
```

```
{1, 2, 3, 'd', 'c', 'a'}
```

```
set_a={"a", ["c", "a"]}
print(set_a)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-2340175852.py in <cell line: 0>()
----> 1 set_a={"a", ["c", "a"]}
      2 print(set_a)

TypeError: unhashable type: 'list'
```

```
# create a sample set
s={1, 2, 3, 4}
print("original set:", s)

#1. add()   adds a single element
s.add(10)
print("after add(10):",s)


#2. update()   adds multiple elements
s.update([11, 12, 13])
print("after update([11,12,13]):", s)
```

```
original set: {1, 2, 3, 4}
after add(10): {1, 2, 3, 4, 10}
after update([11,12,13]): {1, 2, 3, 4, 10, 11, 12, 13}
```

```
#1. add()   adds a single element
s.add(10)
print("after add(10):",s)
```

```
after add(10): {1, 2, 3, 4, 10, 11, 12, 13}
```

```
#2. update()   adds multiple elements
s.update([11, 12, 13])
print("after update([11,12,13]):", s)
```

```
after update([11,12,13]): {1, 2, 3, 4, 10, 11, 12, 13}
```

```
#3. remove()  removes element (error if not found)
s. remove(3)
print("after remove(3):", s)
```

```
after remove(3): {1, 2, 4, 10, 11, 12, 13}
```

```
#4. discard()  removes element (no error if not found)
s. discard(99)  #99 not in set, no error
print("after discard(99):", s)
```

```
after discard(99): {1, 2, 4, 10, 11, 12, 13}
```

```
#5. pop()  removes a random element
removed=s.pop()
print("after pop():", s, "(removed:", removed, ")")
```

```
after pop(): {2, 4, 10, 11, 12, 13} (removed: 1 )
```

```
#6.clear()  removes all elements
temp=s.copy()    #keep a copy
temp.clear()
print("after clear():", temp)

#let's recreate sets for next methods
s1={1,2,3,4}
s2={3,4,5,6}
```

```
after clear(): set()
```

```
#7. union()  combines sets
print("union:", s1.union(s2))    #{1,2,3,4,5,6}
```

```
union: {1, 2, 3, 4, 5, 6}
```

```
#8. intersection()  common elements
print("intersection:", s1.intersection(s2))  #{3,4}
```

```
intersection: {3, 4}
```

```
#9. difference()  elements in s1 not in s2
print("difference (s1-s2):", s1.difference(s2))  #{1,2}
```

```
difference (s1-s2): {1, 2}
```

```
#10. issubset()  check if all elements of s1 are i s2
print("issubset:", {1,2}.issubset(s1))    #true
```

```
issubset: True
```

```
#11. issperset()  check if s1 contains all elements of s2
print("issuoerset:", s1.issuperset({1,2}))    #true
```

```
issuoerset: True
```

```
list_a = [5, "six", [8, 6], 8.2]
print(list_a[2][0])
```

```
8
```

```
list_a = ["five", "six"]
print(list_a[0][1])
```

```
i
```

```
name = input()
age = int(input())
msg = ("Hi" +name +",you are" +str(age) + "years old.")
print(msg)
```

```
ammu
17
Hiammu,you are17years old.
```

```
name = input()
age = int(input())
msg = ("Hi" +name +",you are" +str(age) + "years old." +"ok")
print(msg)
```

```
ammu
17
Hiammu,you are17years old.ok
```

```
dict_a = {"name": "teju",
          "age":  15}
```

```
print(dict_a)
```

```
{'name': 'teju', 'age': 15}
```

```
dict_a = {
  'name': 'teju',
  'age': 17
}
print(dict_a['name'])
```

```
teju
```

```
dict_a = {
  'name': 'teju',
  'age': 17
}
print(dict_a.get('city'))
```

```
None
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
result = 'name' in dict_a
print(result)
```

```
True
```

```
dict_a = {'name': 'teju', 'age': 17}
dict_a['city'] = 'goa'
print(dict_a)
```

```
{'name': 'teju', 'age': 17, 'city': 'goa'}
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
dict_a['age']=18
print(dict_a)
```

```
{'name': 'teju', 'age': 18}
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
```

```
del dict_a['age']
print(dict_a)
```

```
{'name': 'teju'}
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
print(dict_a.keys())
```

```
dict_keys(['name', 'age'])
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
print(dict_a.values())
```

```
dict_values(['teju', 17])
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
print(dict_a.items())
```

```
dict_items([('name', 'teju'), ('age', 17)])
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
for key in dict_a.keys():
  print(key)
```

```
name
age
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
for key in dict_a.values():
  print(key)
```

```
teju
17
```

```
dict_a ={
    'name': 'teju',
    'age': 17

}
for key in dict_a.items():
  print(key)
```

```
('name', 'teju')
('age', 17)
```

```
dict_a ={
    'name1': 'teju',
    'name2': "anu",
    'name': "sukanya"

}
for key in dict_a.values():
  if key=="anu":
    print("your value is found")
    break
  else:
    pass
```

```
your value is found
```

```
name=input()
print(name[0]+"*"*(len(name)-2)+name[-1])
```

```
prajwala
p******a
```

```
def greet():
  print("hello")

name = input()
print(name)
```

```
ammu
ammu
```

```
def greet():
  print("hello")

name = input("enter the name:")
greet()
print(name)
```

```
enter the name:anu
hello
anu
```

```
name = input()
print_name1()
print(name)

def print_name1():
  print("hello")
```

```
ashu
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/ipython-input-649781706.py in <cell line: 0>()
      1 name = input()
----> 2 print_name1()
      3 print(name)
      4
      5 def print_name1():

NameError: name 'print_name1' is not defined
```

```
def greet(word):
  msg = "hello " + word
  print(msg)

name = input("enter the name: ")
greet(word=name)
```

```
enter the name: shahin
hello shahin
```

```
def greet(word):
  msg = "hello " + word
  print(msg)

name = input("enter the name: ")
greet(word=name)
print(msg)
```

```
enter the name: amulya
hello amulya
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/ipython-input-1912108856.py in <cell line: 0>()
      5 name = input("enter the name: ")
      6 greet(word=name)
----> 7 print(msg)

NameError: name 'msg' is not defined
```

```
def greet(word):
  msg = "hello " + word
```

```
name = input()
greet(word=name)
print(msg)
```

```
anu
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
/tmp/ipython-input-4281804241.py in <cell line: 0>()
      5 name = input()
      6 greet(word=name)
----> 7 print(msg)

NameError: name 'msg' is not defined
```

```
def greet(word):
  msg = "hello " + word
  return msg

name = input("enter the name: ")
result = greet(word=name)
print(result)
```

```
enter the name: ammu
hello ammu
```

```
def greet(word):
  msg = "hello " + word
  return msg
  print("passing return")

name = input()
result = greet(word=name)
print(result)
```

```
vasu
hello vasu
```

```
def greet(arg_1, arg_2):
  print(arg_1 + " " + arg_2)

greting = input("enter greeting :")
name = input("enter name : ")
greet(arg_1=greting,arg_2=name)
```

```
enter greeting :hii
enter name : ammu
hii ammu
```

```
def greet(arg_1, arg_2):
  print(arg_1 + " " + arg_2)

greting = input("enter greeting :")
```

```
name = input("enter name : ")
greet(arg_1=greting,)
```

```
enter greeting :hey
enter name : ashu
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
/tmp/ipython-input-1911453636.py in <cell line: 0>()
      4 greting = input("enter greeting :")
      5 name = input("enter name : ")
----> 6 greet(arg_1=greting,)

TypeError: greet() missing 1 required positional argument: 'arg_2'
```

```
def greet(arg_1, arg_2):
  print(arg_1 + " " + arg_2)

greeting = input("enter first name:")
name = input("enter the last name: ")
greet(greeting,name)
```

```
enter first name:hii
enter the last name: vaishu
hii vaishu
```

```
def greet(arg_1="hey", arg_2="bagya"):
  print(arg_1 + " " + arg_2)


greet()
```

```
hey bagya
```

```
def greet(arg_1="hey", arg_2="ashu"):
  print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(greeting)
```

```
hii
ashu
hii ashu
```

```
def greet(arg_2, arg_1="hii"):
  print(arg_1 + " " + arg_2)

greeting = input()
name = input()
greet(name)
```

```
hello
preeti
```

```
hii preeti
```

```
def divisible_by_seven(arg_1):
  #write your code here
  if arg_1%7 == 0:
    print(True)
  else:
    print(False)
```

```
def divisible_by_seven(arg_1):
  #write your code here
  if arg_1%7 == 0:
    print(True)
  else:
    print(False)


n = int(input())
divisible_by_seven(n)
divisible_by_seven(78)
```

```
70
True
False
```

```
def in between 200 and 500(arg_1):
  #write your code here
    if (number>200 and number<500):
    print("Yes")
  else:
    print("No")


n = int(input())

#call the is_between_200_and_500 fuction
```

```
  File "<tokenize>", line 5
    else:
    ^
IndentationError: unindent does not match any outer indentation level
```

```
def numbers():
for i in range(5):
  print(i)
numbers()
```

```
0
1
```

```
2
3
4
```

```
#defining a class
class student:
  def __init__(self):      #constructor
     self.my_name = "vasu"            #instance variable
     self.my_age = 24


  def display(self):
    print("name",self.my_name,    "age:",self.my_age)



#creating objects
s1 = student()

#calling methods using objects
s1. display()
```

```
name vasu age: 24
```

```
#defining a class
class student:
  def __init__(self, name, age):    #constructor
     self.my_name = name           #instance variable
     self.my_age = age


  def display(self):           #method
    print("name",self.my_name, "age:",self.my_age)


#creating objects
s1 = student("ashu", 22)
s2 = student("pooja", 25)

#calling methods using objects
s1. display()
s2. display()
```

```
name ashu age: 22
name pooja age: 25
```

```
class bank:
  def __init__(self,balance): #constructor
   self.balance=balance

  def deposit(self,amount): #method
   self.balance=self.balance+amount
   #self.balance=balance+amount

  def get_balance(self): #method
    return self.balance
```

```
acc=bank(1000)  #obj creation
acc.deposit(500)  #method calling
print("balance:",acc.get_balance())
```

```
balance: 1500
```

```
class calculation:
  def add(self, a, b):
     print("parent class: adding numbers")
     return a+b

class advancecalculator(calculator):  #child inherits calulation
def add(self, a, b):
   #cell parent class method first
   result = super().add(a,b)
   # add child-specific behavior
   print("child class: adding 10 extra")
   return result = 10


#create object of child class
calc = advancedcalculator()
print("final result:", calc.add(5, 3))
```

```
  File "/tmp/ipython-input-1185556991.py", line 12
    return result = 10
                  ^
SyntaxError: invalid syntax
```

```
# Grocery store project (beginner friendly)

#items with prices
grocery_items= {
    "rice":75,
    "oil":120,
    "wheat":40,
    "sugar":42,
    "flour":50,
    "soap":30,
    "salt":17,
    "chocolate":110,
    "chips":50,
    "pickle":85,
    "milk":60,
    "chilli powder":90,
    "cheese":45,

}
print("welcome to the smart bazar!")
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")
```

```
welcome to the smart bazar!
available items:
```

```
rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
salt-17per unit
chocolate-110per unit
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit
```

```
cart = {"oil":2}
choice = "rice"
qty = 3
cart[choice]= qty  # 2 + 3 = 5
print(cart)
```

```
{'oil': 2, 'rice': 3}
```

```
cart = {"oil":2}
choice = "oil"
qty = 3
cart[choice]= cart.get(choice, 0)+ qty  # 2 + 3 = 5
print(cart)
```

```
{'oil': 5}
```

```
cart = {"rice":2}
choice = "rice"
qty = 3
cart[choice]= qty  # 2 + 3 = 5
print(cart)
```

```
{'rice': 3}
```

```
print("welcome to the more!")
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")


#cart to store items
cart={}

while True:
  choice = input("\nenter item name(or type 'done' to finish):")

  if choice.lower()=="done":
    break

  if choice in grocery_items:
```

```
    qty=int(input("enter qty of"+choice +":"))
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
  else:
    print("item not available!")
```

```
welcome to the more!
available items:

rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
salt-17per unit
chocolate-110per unit
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit

enter item name(or type 'done' to finish):milk
enter qty ofmilk:3
milk added to cart.

enter item name(or type 'done' to finish):done
```

```
print("welcome to the more!")
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")


#cart to store items
cart={}

while True:
  choice = input("\nenter item name(or type 'done' to finish):")

  if choice.lower()=="done":
    print("just a shop again  ")
    break

  if choice in grocery_items:
    qty=int(input("enter qty of"+choice +":"))
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
  else:
    print("item not available!")
```

```
welcome to the more!
available items:
```

```
rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
salt-17per unit
chocolate-110per unit
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit

enter item name(or type 'done' to finish):oil
enter qty ofoil:5
oil added to cart.

enter item name(or type 'done' to finish):done
just a shop again
```

```
grocery_items= {
    "rice":75,
    "oil":120,
    "wheat":40,
    "sugar":42,
    "flour":50,
    "soap":30,
    "salt":17,
    "chocolate":110,
    "chips":50,
    "pickle":85,
    "milk":60,
    "chilli powder":90,
    "cheese":45,

}
print(grocery_items["oil"])
```

```
120
```

```
print("welcome to the more!")
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")


#cart to store items
cart={}

while True:
  choice = input("\nenter item name(or type 'don' to finish):")

  if choice.lower()=="done":
    print("thank you for shopping")
    break
```

```
  if choice in grocery_items:
    qty=int(input("enter qty of"+choice +":"))
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
else:
    print("item not available!")
    print("\nYour bill:")
    total=0

    for item, qty in cart.items():
      price = grocery_items[items] * qty
      print(item, "-", qty, "x", grocery_items[item], "=", price)
      total += price

      print("total amount:",total)
      print("thank you! visit again")
```

```
welcome to the more!
available items:

rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
salt-17per unit
chocolate-110per unit
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit

enter item name(or type 'don' to finish):oil
enter qty ofoil:4
oil added to cart.

enter item name(or type 'don' to finish):done
thank you for shopping
```

```
# Grocery store project (beginner friendly)

#items with prices
grocery_items= {
    "rice":75,
    "oil":120,
    "wheat":40,
    "sugar":42,
    "flour":50,
    "soap":30,
    "salt":17,
    "chocolate":110,
    "chips":50,
    "pickle":85,
    "milk":60,
    "chilli powder":90,
```

```
    "cheese":45,
}
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")

#cart to store items
  cart={}

while True:
  choice = input("\nenter item name(or type 'done' to finish):")

  if choice.lower()=="done":
    break

  if choice in grocery_items:
    qty=int(input("enter qty of"+choice +":"))
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
  else:
    print("item not available!")
    print("\nYour bill:")
total=0

for item, qty in cart.items():
  price = grocery_items[item] * qty
  print(item, "-", qty, "x", grocery_items[item], "=", price)
  total += price

  print("total Amount:",total)
  print("thank you for shopping!")

if total > 300:
  discount = total * 0.10  #10% discount
  after_discount = total - discount
  gst = after_discount * 0.05   # 5% GST
  final_amount = after_discount + gst

  print("discount (10%): -", discount)
  print("amount after discout:", after_discount)
  print("GST (5%): +", gst)
  print("final amount to pay:", final_amount)


else:
  gst = total*0.05  # 5% GST
  final_amount = gst
  print("final amount to pay:", final_amount)
  print("reminder: shop more next time! if your bill exceeds 300, you will get 10
```

```
available items:

rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
```

```
salt-17per unit
chocolate-110per unit
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit

enter item name(or type 'done' to finish):oil
enter qty ofoil:5
oil added to cart.

enter item name(or type 'done' to finish):donr
item not available!

Your bill:

enter item name(or type 'done' to finish):done
oil - 5 x 120 = 600
total Amount: 600
thank you for shopping!
discount (10%): - 60.0
amount after discout: 540.0
GST (5%): + 27.0
final amount to pay: 567.0
```

```
# Grocery store project (beginner friendly)
import random
#items with prices
grocery_items= {
    "rice":75,
    "oil":120,
    "wheat":40,
    "sugar":42,
    "flour":50,
    "soap":30,
    "salt":17,
    "chocolate":110,
    "chips":50,
    "pickle":85,
    "milk":60,
    "chilli powder":90,
    "cheese":45,
}
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")

#cart to store items
  cart={}

while True:
  choice = input("\nenter item name(or type 'done' to finish):")

  if choice.lower()=="done":
    break
```

```
  if choice in grocery_items:
    qty=int(input("enter qty of"+choice +":"))
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
  else:
    print("item not available!")
    print("\nYour bill:")
total=0

for item, qty in cart.items():
  price = grocery_items[item] * qty
  print(item, "-", qty, "x", grocery_items[item], "=", price)
  total += price

  print("total Amount:",total)
  print("thank you for shopping!")

if total > 300:
  discount = total * 0.10  #10% discount
  after_discount = total - discount
  gst = after_discount * 0.05   # 5% GST
  final_amount = after_discount + gst

  print("discount (10%): -", discount)
  print("amount after discout:", after_discount)
  print("GST (5%): +", gst)
  print("final amount to pay:", final_amount)


else:
  gst = total*0.05  # 5% GST
  final_amount = gst
  print("final amount to pay:", final_amount)
  print("reminder: shop more next time! if your bill exceeds 300, you will get 10


#  free gift feature
if total > 300:
  gifts = ["shopping bag", "tiffin box", "dairy"]
  free_gift = random.choice(gifts)
  print("\ncongratulations! you get a free gift:", free_gift)


#  loyalty points feature
loyalty_points = total // 100  # point for every 100 spent
print("you earned", loyalty_points, "loyalty points today!")

print("\nthank you shopping with us!")
```

```
available items:

rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
salt-17per unit
chocolate-110per unit
```

```
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit

enter item name(or type 'done' to finish):done
final amount to pay: 0.0
reminder: shop more next time! if your bill exceeds 300, you will get 10% di
you earned 0 loyalty points today!

thank you shopping with us!
```

```
class dog:
  def sound(self):
    return "bark"


class cat:
  def sound(self):
    return "meow"

class cow:
  def sound(self):
    return "moo"

animals = [dog(), cat(), cow()]

for animal in animals:
  print(animal.sound())
```

```
bark
meow
moo
```

```
class dog:
  def sound(self):
    return "bark"


class cat:
  def sound(self):
    return "meow"

class cow:
  def sound(self):
    return "moo"

animals = [dog(), cat(),]

for animal in animals:
  print(animal.sound())
```

```
bark
meow
```

```
#installation
!pip install numpy

#importing NumPy
import numpy as np
```

```
Requirement already satisfied: numpy in /usr/local/lib/python3.12/dist-packa
```

```
# Grocery store project (beginner friendly)
import random
# step 1: item with prices
#  ----------------------------
grocery_items= {
    "rice":75,
    "oil":120,
    "wheat":40,
    "sugar":42,
    "flour":50,
    "soap":30,
    "salt":17,
    "chocolate":110,
    "chips":50,
    "pickle":85,
    "milk":60,
    "chilli powder":90,
    "cheese":45,
}

print("welcom to the more!")
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")

#cart to store items
  cart={}

while True:
  choice = input("\nenter item name(or type 'done' to finish):")

  if choice.lower()=="done":
    break

  if choice in grocery_items:
    qty=int(input("enter qty of"+choice +":"))
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
  else:
    print("item not available!")
    print("\nYour bill:")
total=0

for item, qty in cart.items():
  price = grocery_items[item] * qty
  print(item, "-", qty, "x", grocery_items[item], "=", price)
  total += price

  print("total Amount:",total)
  print("thank you for shopping!")
```

```
if total > 300:
  discount = total * 0.10  #10% discount
  after_discount = total - discount
  gst = after_discount * 0.05   # 5% GST
  final_amount = after_discount + gst

  print("discount (10%): -", discount)
  print("amount after discout:", after_discount)
  print("GST (5%): +", gst)
  print("final amount to pay:", final_amount)


else:
  gst = total*0.05  # 5% GST
  final_amount = gst
  print("final amount to pay:", final_amount)
  print("reminder: shop more next time! if your bill exceeds 300, you will get 10


#  free gift feature
if total > 300:
  gifts = ["shopping bag", "tiffin box", "dairy"]
  free_gift = random.choice(gifts)
  print("\ncongratulations! you get a free gift:", free_gift)


#  loyalty points feature
loyalty_points = total // 100  # point for every 100 spent
print("you earned", loyalty_points, "loyalty points today!")

print("\nthank you shopping with us!")
```

```
welcom to the more!
available items:

rice-75per unit
oil-120per unit
wheat-40per unit
sugar-42per unit
flour-50per unit
soap-30per unit
salt-17per unit
chocolate-110per unit
chips-50per unit
pickle-85per unit
milk-60per unit
chilli powder-90per unit
cheese-45per unit

enter item name(or type 'done' to finish):oil
enter qty ofoil:5
oil added to cart.

enter item name(or type 'done' to finish):done
oil - 5 x 120 = 600
total Amount: 600
thank you for shopping!
```

```
discount (10%): - 60.0
amount after discout: 540.0
GST (5%): + 27.0
final amount to pay: 567.0

congratulations! you get a free gift: tiffin box
you earned 6 loyalty points today!

thank you shopping with us!
```

```
# Grocery store project (beginner friendly)
import random
import numpy as np  #for simple
average feature

# step 1: item with prices
#  ----------------------------
grocery_items= {
    "rice":75,
    "oil":120,
    "wheat":40,
    "sugar":42,
    "flour":50,
    "soap":30,
    "salt":17,
    "chocolate":110,
    "chips":50,
    "pickle":85,
    "milk":60,
    "chilli powder":90,
    "cheese":45,
}

print("welcom to the more!")
print("available items:\n")
for item, price in grocery_items.items():
  print(item+"-"+str(price)+ "per unit")

# ---------------------------------------
# step 2: customer details
# ---------------------------------------
name = input("\nenter your name: ")
mobile = input("enter your mobile number: ")


# -----------------------------------------
# step 3: cart system
# -----------------------------------------
cart = {}


while True:
  choice = input("\nenter item name(or type 'done' to finish):")

  if choice.lower()=="done":
    break

  if choice in grocery_items:
    qty=int(input("enter qty of"+choice +":"))
```

```
    cart[choice]=cart.get(choice,0)+qty
    print(choice,"added to cart.")
  else:
    print("item not available!")


#-----------------------------
#step 4: bill calculation
# --------------------------
total = 0
bill_item = []


for item, qty in cart.items():
  price = grocery_items[item]
  amount = price * qty
  bill_items.append([item, qty, price, amount])
  total += amount


# discount + GST
discount = 0
if total >1000:
  discount = total *0.10
  after_discount = total-discount

else:
  after_discount = total

gst = after_dicount * 0.05
final_amount = after_discount + gst


#  free gift feature
if total > 300:
  gifts = ["shopping bag", "tiffin box", "dairy"]
  free_gift = random.choice(gifts)
  print("\ncongratulations! you get a free gift:", free_gift)

#  loyalty points feature
loyalty_points = total //100

#NumPy feature-average price
if cart:
  prices_list=[grocery_items[item] for item in cart.keys()]
  avg_price=np.mean(prices_list)
else:
  avg_price=0
# ------------------------------------------------------

#step 5: print bill reciept
#-----------------------------------
print("\n"+ "="*40)
print("            more smart market")
print("="*40)
print("customer name :", name)
print("mobile number :", mobile)
print("-"*40)
print("item          qty     price      amount")
print("-"*40)
```

```
for item,qty,price,amount in bill_item:
  print(f"{item:10} {qty:<5} {price:<7} {amount}")
  print(item,qty,price)


print("-"*40)
print("total amount          :",total)
print("discount  (10%)       :",round(discount,2))
print("amount after disc.    :",round(after_discount,2))
print("GST (5%)              :",round(gst,2))
print("final amount to pay   :",round(final_amount,2))
print("-"*40)

if free_gift:
  print(" free gift:", free gift)
```