String formatting or formatting string

Format string is used for formatting output.
Python supports formatting string/formatting output in different ways.

1. Old style string formatting
2. New style string formatting
3. F-string

Old-style string formatting
Old style string formatting is similar to C-Style String formatting.
In this formatting, string contains replacement fields/formatting fields, which replaced with values.

Syntax: "formatting fields"%(value,value,value,…)

% → string interpolation operator

%d → decimal integer
%o → octal integer
%x → hexadecimal integer
%f → float in fixed
%e → float in exponent
%s → String
%c → Character

Example:

```
n1=10
n2=20
n3=n1+n2

print("sum of ",n1,"and",n2,"is",n3)

print("sum of %d and %d is %d"%(n1,n2,n3))
print("sum of %o and %o is %o"%(n1,n2,n3))
print("sum of %x and %x is %x"%(n1,n2,n3))
```

Output

```
sum of  10 and 20 is 30
sum of 10 and 20 is 30
sum of 12 and 24 is 36
sum of a and 14 is 1e
```

Example:

 # Write a program to find area of triangle

```
base=float(input("Enter Base of the triangle "))
height=float(input("Enter Hieght of the triangle "))

area=0.5*base*height

print("Area of triangle with base=%.2f and height=%.2f is
%.2f"%(base,height,area))
```

Output

```
Enter Base of the triangle 1.2
Enter Hieght of the triangle 1.3
Area of triangle with base=1.20 and height=1.30 is 0.78
```

Example

```
# Write a program to input rollno,name,course,fee and
display/output
rollno=int(input("Enter Rollno "))
name=input("Enter Name ")
course=input("Enter Course ")
fee=float(input("Enter Fee "))


print('''Rollno\t%d
Name\t%s
Course\t%s
Fee\t%f'''%(rollno,name,course,fee))
```

Output

```
Enter Rollno 101
Enter Name Naresh
Enter Course Python
Enter Fee 5000
Rollno      101
Name        Naresh
Course      Python
Fee   5000.000000
```

New style string formatting

New style string formatting is done using format method of string data type/class.

"<formatting>".format(value/variable,…)

In style string formatting replacement fields/formatting fields are represented using {}

Example:

```
n1=10
n2=20
n3=n1+n2

print("sum of {} and {} is {}".format(n1,n2,n3))

print("sum of {:d} and {:d} is {:d}".format(n1,n2,n3))
print("sum of {:o} and {:o} is {:o}".format(n1,n2,n3))
print("sum of {:x} and {:x} is {:x}".format(n1,n2,n3))
print("sum of {:b} and {:b} is {:b}".format(n1,n2,n3))
```

Output

```
sum of 10 and 20 is 30
sum of 10 and 20 is 30
sum of 12 and 24 is 36
sum of a and 14 is 1e
sum of 1010 and 10100 is 11110
```

Example:

```
print("{a:d},{b:d}".format(a=10,b=20))
print("{a:d},{b:o},{c:x},{d:b}".format(a=10,b=20,c=30,d=40))

n1=int(input("Enter first number "))
n2=int(input("Enter second number "))
n3=n1+n2

print("sum of {a:d} and {b:d} is {c:d}".format(c=n3,a=n1,b=n2))
```

Output

```
10,20
10,24,1e,101000
Enter first number 10
Enter second number 20
sum of 10 and 20 is 30
```

Example:

```
# Write a program to find simple interest


amt=float(input("Enter Amount "))
rate=float(input("Enter Rate "))
time=int(input("Enter Time "))

si=amt*rate*time/100

print('''Simple Interest with
Amount={:.2f}
Rate={:.2f}
Time={:d} is {:.2f}'''.format(amt,rate,time,si))
```

Output

```
Enter Amount 5000
Enter Rate 1.2
Enter Time 12
```

Simple Interest with
Amount=5000.00
Rate=1.20
Time=12 is 720.00

f-string

f-string is introduced in python 3.8 version.
Any string prefix with “f” or “F” is called format string.
This string contains formatting fields, which are replaced with values.

Syntax: f‘format string‘
Syntax: F’format string’

Example

```
n1=10
n2=20
n3=n1+n2

print("sum of %d and %d is %d"%(n1,n2,n3))
print("sum of {} and {} is {}".format(n1,n2,n3))
print(f'sum of {n1} and {n2} is {n3}')
print(f'sum of {n1:d} and {n2:d} is {n3:d}')
print(f'sum of {n1:o} and {n2:o} is {n3:o}')
print(f'sum of {n1:x} and {n2:x} is {n3:x}')
print(f'sum of {n1:b} and {n2:b} is {n3:b}')

f1=1.5
f2=1.2
f3=f1+f2
print(f'sum of {f1} and {f2} is {f3}')
print(f'sum of {f1:f} and {f2:f} is {f3:f}')
print(f'sum of {f1:.2f} and {f2:.2f} is {f3:.2f}')
print(f'sum of {f1:e} and {f2:e} is {f3:e}')

rollno=1
name="naresh"
```

print(f"My name is {name:s} and Rollno is {rollno:d}")

Output

```
sum of 10 and 20 is 30
sum of 10 and 20 is 30
sum of 10 and 20 is 30
sum of 10 and 20 is 30
sum of 12 and 24 is 36
sum of a and 14 is 1e
sum of 1010 and 10100 is 11110
sum of 1.5 and 1.2 is 2.7
sum of 1.500000 and 1.200000 is 2.700000
sum of 1.50 and 1.20 is 2.70
sum of 1.500000e+00 and 1.200000e+00 is 2.700000e+00
My name is naresh and Rollno is 1
```

Nested if

Defining if within if is called nested if (OR) if followed by if is called nested if.

Syntax:
if <condition1>: → Outer if
  if <condition2>: → Inner if
  statement-1
  else:
  statement-2
elif <condition3>:
  if <condition4>:
  statement-3
  else:
  statement-4
else:
  statement-6
	If condition1, condition2 are True, PVM executes statement-1
	If condition1 is True and condition2 is False, PVM executes statement-2

Example # Login Application	Output UserName :nit Password :nit123 welcome

```
uname=input("UserName :")
pwd=input("Password :")

if uname=="nit":
    if pwd=="nit123":
        print("welcome")
    else:
        print("invalid password")
else:
    print("invalid username")
```

```
UserName :nit
Password :n234
invalid password

UserName :xy
Password :abc
invalid username
```