

# String Formatting

Yankun (Alex) Meng

There are three ways to format strings:

1. “Old style” string formatting
2. Using “.format()” Method
3. Using python f-strings

# Old Style string formatting with %

```
1 # data type comes after a modulo sign
2 # %s - string, and it is inside quotation marks
3 # to assign value, you need to call % outside string
4 print("Hello, %s" % "Alex!")
```

Hello Alex!

```
1 # Use %d for integers and %f for floating point numbers
2 # In order to insert multiple values, use a tuple
3 name = "Alex"
4 age = 20.2
5 money = 1000.56789
6
7 print("%s is %d years old and has $%f." % (name, age, money))
```

Alex is 20 years old and has \$1000.567890.

```
1 # Write it this way to be cleaner
2 fstring = "Hello %s, your current balance is $%f"
3 data = ("Alex", 1000.56789)
4 print(fstring % data)
```

Hello Alex, your current balance is \$1000.567890

# General Arguments

`%[key][flags][width][.precision][length_type][conversion type] %  
[value]`

```
1 # Simplest string formatting  
2 print("%s %s" % ("foo", "bar"))
```

foo bar

```
1 # Floats  
2 # %[type]  
3 print("%f %f" % (1, 2))
```

1.000000 2.000000

```
1 # Working with dictionaries  
2 # %([key])[type] → prints out [value]  
3 dct = {"Alex":20, 'Bob':18}  
4 print("%(Alex)s" % dct)
```

20

```
1 # We can specify the minimum width given to the inputs  
2 # width → how much space to reserve for the value  
3 # (C style programming)  
4 # 2 and 4 are widths  
5 # %[key][width][type]  
6
```



# Using `.format()` Method

# Using f-strings

# Resources

- [Old Style Tutorial](#)
- [.format\(\) Method Tutorial](#)
- [fString Tutorial](#)