C-Style Formatting

% - Quotient Operator or Modular Operator or Percent Operator

- **% - it is used for formatting Strings, it is also called Percent Operator**
- **%s,%d - it is format specifiers, or place holders.**
- **%s - string place holder**
- **%d,%i - integer place holder**
- **%f - floating Point**

In [1]:

```
# Writing In Sequence :-
print("My Name is %s and My age is %d" % ("Aadil",22))
                            # order must be maintained or it gives error.
```

```
My Name is Aadil and My age is 22
```

In [2]:

```
# Writing not in Sequence :- {key,Value}

print("My Name is %(nm)s and My age is %(ag)d" % {'nm':"Aadil",'ag': 22})
```

```
My Name is Aadil and My age is 22
```

In [3]:

```
print("My Name is %(nm)s and My age is %(ag)d" % {'ag': 22,'nm':"Aadil"})
```

```
My Name is Aadil and My age is 22
```

In [4]:

```
print("%d" % 432)
```

```
432
```

In [5]:

```
print("%d %d" % (432,678))
```

```
432 678
```

In [6]:

```
print("%f" % 432.32)
```

```
432.320000
```

In [7]:

```
print("%s" % "Aadil")
```

```
Aadil
```

In [8]:

```
print("%s %s" % ("Aadil","gta"))
```

```
Aadil gta
```

In [9]:

```
print("%s %s" % ("Aadil","345"))
```

```
Aadil 345
```

In [10]:

```
print("%s %d" % ("Aadil",345))
```

```
Aadil 345
```

In [11]:

```
print("%(nm)s %(ag)d" % {'nm' : "Aadil",'ag':345})
```

```
Aadil 345
```

In [12]:

```
print("%s" % (34))
```

```
34
```

In [13]:

```
print("%s" % 34)
```

```
34
```

In [14]:

```
a = "%s" % 34

print(a)
```

```
34
```

----------Throws Error ----------

In [15]:

```
print("%s %d"  %  (678, "Aadil"))
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-15-93244fb02ce9> in <module>
----> 1 print("%s %d"  %  (678, "Aadil"))

TypeError: %d format: a number is required, not str
```

Flag

In [16]:

```
# space

print("%d" % 34)
print("% d" % 34)      # ----- space
print("%  d" % 34)     # ----- double space
print("%    d" % 34)
print("%+d" % 34)
```

```
34
 34
 34
 34
+34
```

In [17]:

```
print("%  d" % 34)
print("%d""hello" % 34)
```

```
 34
34hello
```

In [18]:

```
print("              %d"                    "hello" % 34)
```

```
              34hello
```

In [19]:

```
print("              %d"                    "               hello" % 34)
```

```
              34               hello
```

In [20]:

```
print("              %d                    hello" % 34)
```

```
              34                    hello
```

In [21]:

```
print("%d hello" % 34)
```

```
34 hello
```

Width

In [22]:

```
print("% 8d" % 34)
                  # here width creates 8 boxes.
```

```
      34
```

In [23]:

```
print("%8d" % 34)         #   0  1  2  3  4  5  6  7  8
                          #____|__|__|__|__|__|__|__|___|
                          #  -| -| -| -| -| -| -| 3| 4
```

```
      34
```

Flag with width

In [24]:

```
print("%08d" % 34)        # flag fills zeros in empty boxes
```

```
00000034
```

Precision Of Decimals

In [25]:

```
print("%f" % 345.567)
```

```
345.567000
```

In [26]:

```
print("%.3f" % 345.567)      # 3 4 5 . 5 6 7
```

```
345.567
```

In [27]:

```
print("%.2f" % 345.567)
```

```
345.57
```

In [28]:

```
print("%.2f" % 345.128)        # Round off of decimal
```

```
345.13
```

Combination of width,flag & precision

In [31]:

```
# width creates empty boxes in memory location
```

In [32]:

```
print("%9.2f" % 345.567)     # 3 empty boxes
```

```
   345.57
```

In [33]:

```
print("%9.3f" % 345.567)
```

```
  345.567
```

In [35]:

```
print("%9.2f" % 345.128)       # round off
```

```
   345.13
```

In [37]:

```
print("%09.2f" % 345.567)
```

```
000345.57
```

In [38]:

```
a = ("My Name is %s and My age is %d" % ("Aadil",22))
print(a)
```

```
My Name is Aadil and My age is 22
```