String

- **String are Immutable.**

Memory Allocation of Strings

In [42]:

```
str1 = "Aadil"
```

In [43]:

```
id(str1)
```

Out[43]:

```
85295920
```

In [44]:

```
str2= "Aadil"
```

In [45]:

```
id(str2)
```

Out[45]:

```
85295920
```

In [46]:

```
str3 = "python"
```

In [47]:

```
id(str3)
```

Out[47]:

```
55719216
```

In [48]:

```
str4 = str3
```

In [49]:

```
id(str4)      # memory allocation of str 3 and str4 are same
```

Out[49]:

```
55719216
```

In [50]:

```
str_1 = "Sam"
```

In [51]:

```
str_1 = "gem"
```

In [52]:

```
str_1
```

Out[52]:

```
'gem'
```

In [53]:

```
id(str_1)
```

Out[53]:

```
85271024
```

- **Remember that id of old str_1 sam is delted by garbage collector of python.**
  index starts from zero.

In [54]:

```
Strn = "aadil"
```

In [55]:

```
Strn[-1]
```

Out[55]:

```
'l'
```

Comparision of strings

In [56]:

```
str = "A"
```

In [57]:

```
str1 = "B"
```

In [58]:

```
str1<str
```

Out[58]:

```
False
```

In [59]:

```
str11 = "Aa"
str22 = "Bb"
str22 > str11
```

Out[59]:

```
True
```

In [60]:

```
str11 = "Aadil"
str22 = "Sahib"
str22 > str11
```

Out[60]:

```
True
```

In [61]:

```
str11 = "Aadil"
```

```
str22 = "Sahib"
str22 == str11
```

Out[61]:

```
False
```

Repetition Operators

In [62]:

```
print(str11 * 5)
```

```
AadilAadilAadilAadilAadil
```

In [63]:

```
str22 = "Sahib "
(str11 * 5)
```

Out[63]:

```
'AadilAadilAadilAadilAadil'
```

In [64]:

```
str22 = "Sahib "
print(str11 * 5)
```

```
AadilAadilAadilAadilAadil
```

In [65]:

```
str22 = "Sahib\n"
print(str22 * 5)
```

```
Sahib
Sahib
Sahib
Sahib
Sahib
```

Accessing String using Loops

In [66]:

```
str_loops = "Aadil"
for i in str_loops:
    print(i)
```

```
A
a
d
i
l
```

In [67]:

```
str_loops = "Aadil"
for i in range(len(str_loops)):
    print(f'{i} = {str_loops[i]}')
```

```
0 = A
1 = a
2 = d
3 = i
4 = l
```

In [68]:

```
str_loops = "Aadil"
n = len(str_loops)
for i in (range(n)):
    print(str_loops[i])
```

```
A
a
d
i
l
```

In [69]:

```
str = "Aadil"
n = len(str)
i = 0
while i<n:
    print(str[i])
    i += 1
```

```
A
a
d
i
l
```

In [70]:

```
str = "Aadil"
i = 0
while i<len(str):
    print(str[i])
    i += 1
```

```
A
a
d
i
l
```

In [71]:

```
str = "Aadil"
i = 0
while i<len(str):
    print(f'{i} = {str[i]}')
    i += 1
                # range function cant be used with for loops so, we use starter & incrime
nter
```

```
0 = A
1 = a
2 = d
3 = i
4 = l
```

Strip Function ()

In [72]:

```
name = "        Aadil"
name.lstrip()
```

Out[72]:

```
'Aadil'
```

In [73]:

```
name = "        Aadil"
print(name.lstrip())   # left space remove
```

```
Aadil
```

In [74]:

```
name = "    Aadil                    "
print(name.rstrip())        # right space remove
```

```
    Aadil
```

In [75]:

```
name = "    Aadil                    "
print(name.strip())    # all side space remove
```

```
Aadil
```

Replace Function

In [76]:

```
name = "aadilShows"
old = "aadil"
new = "gta"
str1 = name.replace(old,new)
print(name)
print(str1)
```

```
aadilShows
gtaShows
```

In [77]:

```
name = "aadilShows"
str1 = name.replace("aadil",'new')
print(name)
print(str1)
```

```
aadilShows
newShows
```

split() Function

In [78]:

```
name = "Hello-How-Are-you"
str1 = name.split('-')
print(str1)

             # it writtens a list
```

```
['Hello', 'How', 'Are', 'you']
```

In [79]:

```
name = "Aadil Ji"
str1 = name.split(' ')
print(str1)
```

```
['Aadil', 'Ji']
```

Join() Function

In [80]:

```
name = ('aadil','kaise','ho')
str1 = '_'.join(name)
print(str1)
                # seperator is '_'
```

```
aadil_kaise_ho
```

In [81]:

```
name = ('aadil','kaise','ho','sb','badhiya')
str1 = ' '.join(name)
print(str1)
                 # seperator is '<space>'
```

```
aadil kaise ho sb badhiya
```

In [82]:

```
name = ('aadil','kaise','ho','sb','badhiya')
str1 = '-'.join(name)
print(str1)
              # seperator is '-'
```

```
aadil-kaise-ho-sb-badhiya
```