strings

In [1]:
```
s="hello"
```

In [3]:
```
type(s)
```

Out[3]:
```
str
```

In [5]:
```
len(s)
```

Out[5]:
```
5
```

In [7]:
```
s.upper()
```

Out[7]:
```
'HELLO'
```

In [9]:
```
s.lower()
```

Out[9]:
```
'hello'
```

In [11]:
```
s.title()
```

Out[11]:
```
'Hello'
```

In [13]:
```
s.swapcase()
```

Out[13]:
```
'HELLO'
```

In [15]:
```
s.capitalize()
```

Out[15]:
```
'Hello'
```

In [ ]:
```
s.
```

In [ ]:
```

```

In [ ]:
```

```

escap character

In [73]:
```
print("hii.\n how r u?\n")   #\n means return new line
```

```
hii.
 how r u?
```

In [75]:
```
print("hii.\t how r u?\t")  #\t means tab space
```

```
hii.     how r u?
```

In [77]:
```
print("hii.\b how r u?\b") #\b means backspace
```

```
hii how r u
```

In [79]:
```
print("hii.\r how r u?\r") #carriage returns
```

```
 how r u?
```

In [83]:
```
print("hii.\\ how r u?\\")
```

```
hii.\ how r u?\
```

indexing and silicing

In [86]:
```
s1="datascience"
```

In [90]:
```
s1[0]
```

Out[90]:
```
'd'
```

In [94]:
```
s1[5]
```

Out[94]:
```
'c'
```

In [96]:
```
s1[::]
```

Out[96]:
```
'datascience'
```

In [98]:
```
s1[4:10]
```

Out[98]:
```
'scienc'
```

In [100…
```
s1[0:11:2] #two step silicing
```

Out[100…
```
'dtsine'
```

"in" and "not in"

In [105…
```
"hello" in "helloworld"
```

Out[105…
```
True
```

In [107…
```
"hello" not in "helloworld"
```

Out[107…
```
False
```

- when we use "in" then if the given char or string is part of the main then it returns "true" elase "false"

- when we use "not in" then if the given char or string is part of the main string then it returns the " false" else "true".

"start with" and "end with"

In [116…
```
"good morning".endswith("good")
```

Out[116…
```
False
```

In [118…
```
"good morning".startswith("good")
```

Out[118…
```
True
```

join and split

In [123…
```
','.join(["hii","hello"])
```

Out[123…
```
'hii,hello'
```

In [127…
```
' @ '.join(["hii","hello"])
```

Out[127…
```
'hii @ hello'
```

In [129…
```
"hello how r u".split('h')
```

Out[129…
```
['', 'ello ', 'ow r u']
```

rjust,ljust,center

In [134…
```
"hello".rjust(10)
```

Out[134…
```
'     hello'
```

In [138…
```
"hello".ljust(20)
```

Out[138…
```
'hello               '
```

In [140…
```
"hello".ljust(10)
```

Out[140…
```
'hello     '
```

In [146…
```
"hello".center(10)
```

Out[146…
```
'  hello   '
```

removing the whitespace vth strip(),rstrip(),lstrip()

In [173…
```
s3="       hello good  evening        ".lstrip()
```

In [156…
```
s3
```

Out[156…
```
'hello  good evening'
```

In [175…
```
s3.rstrip()
```

Out[175…
```
'hello good  evening'
```

In [177…
```
s3.strip( )
```

Out[177…
```
'hello good  evening'
```

count

In [179…
```
s4="once up on a time"
```

In [181…
```
s4.count("on")
```

Out[181…
```
2
```

In [183…
```
s4.count("e")
```

Out[183…
```
2
```

replace

In [188…
```
text="hello world"
text.replace("world","planet")
```

Out[188…
```
'hello planet'
```

In [192…
```
fruit="apple,banana,pineapple,apple"
```

In [194…
```
fruit.replace("apple","orange",1)
```

Out[194…
```
'orange,banana,pineapple,apple'
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: