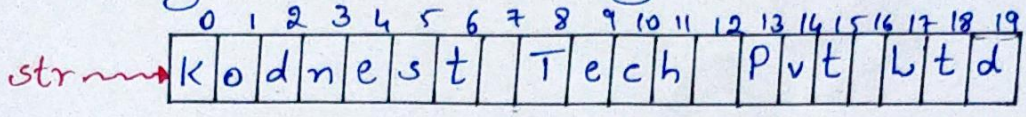
# Inbuilt methods present in String

Consider:

```
String str = "Kodnest Tech Pvt Ltd";

String s1 = str.toUpperCase();
String s2 = str.toLowerCase();
String s3 = str.substring(8, 12);
String s4 = str.substring(8);
```




.toUpperCase(): converts all the letters in the String to Lowercase letters.

s1 → KODNEST TECH PVT LTD

.toLowerCase: converts all the letters in the String to Uppercase letters.

s2 → kodnest tech pvt ltd

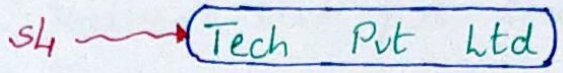
.substring(starting index, Ending index):

returns substring from the given String from starting index to ending index that is given as parameters (input) excluding ending index.

s3 → Tech

.substring(starting index):

returns the substring from the given string from the entered starting string index to the end of the string.




Program:

```
public class StringMethodsDemo {
  public static void main(String []args)
  {
    String str = "Kodnest Tech Pvt Ltd";
    String s1 = str.toUpperCase();
    String s2 = str.toLowerCase();
    String s3 = str.substring(8,12);
    String s4 = str.substring(8);
    System.out.println("Original string = "+str);
    System.out.println("s1 = "+s1);
    System.out.println("s2 = "+ s2);
    System.out.println("s3 = "+ s3);
    System.out.println("s4 = "+s4);
  }}
```

Output:

Original string = Kodnest Tech Pvt Ltd

s1 = KODNEST TECH PVT LTD

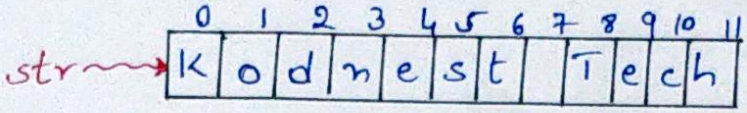
s2 = kodnest tech pvt ltd

s3 = Tech

s4 = Tech Pvt Ltd.

Consider:

```
String  str = "Kodnest Tech";
int l = str.length();
boolean pr = str.contains("T");
char ch[] = str.toCharArray();
String s[] = str.split("_");
String str2 = str.concat("_Software");
```



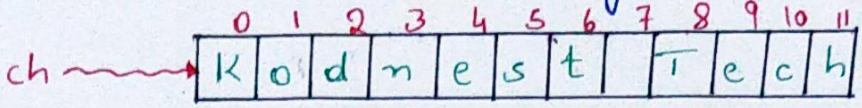

.length(): returns number of characters present in the string in integer type (number)

returns "12" for string "Kodnest Tech"

.contains("input"): returns true if given character or set of characters present in the string or returns false if it is not present

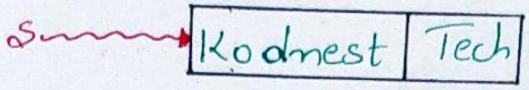
returns "true" for string "Kodnest Tech"

.toCharArray(): converts the given string into a character array



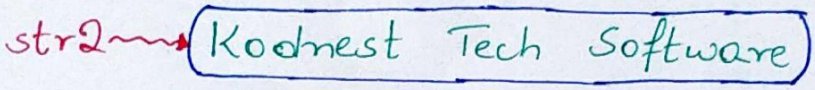

---

.split("delimiter/input"): converts the given string into a string array by splitting the string according to the input/delimiter.




---

.concat("input string"): joins the input string with the given string



Program:

```
public class StringMethodsDemo
{
public static void main(String []args)
{
String str = "Kodnest Tech";
int l = str.length();
boolean pr = str.contains("T");
char ch[] = str.toCharArray();
String s[] = str.split(" ");
String str2 = str.concat(" software");
System.out.println("original string:"+str);
System.out.println("length of string:"+l);
System.out.println("is given character present:"
                                    +pr);
System.out.println("printing elements of
                    array ch:");
```

```
for(int i=0; i<=ch.length-1; i++)
{
 System.out.println(ch[i]);
}
System.out.println("printing elements of
                    array s:");
 for(int i=0; i<s.length; i++)
{
  System.out.println(s[i]);
 }
System.out.println("concatenated string:"+str2);
 }
}
```

Output:

original string: Kodnest Tech
length of string: 12
is given character present: true
printing elements of array ch:
K
o
d
n
e
s

t

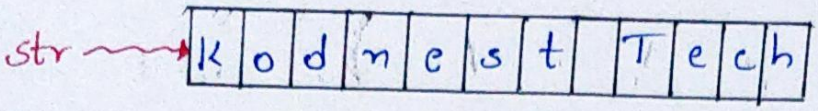
T
e
c
h

printing elements of array s:

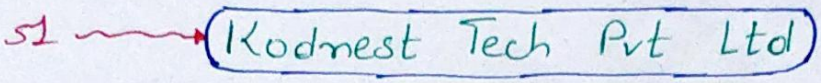
Kodnest
Tech

concatenated string: Kodnest Tech Softwares

```
String str = "Kodnest Tech";
String s1 = str + " Pvt Ltd";
char ch = str.charAt(5);
```




"+": + operater usage is allowed in java to concatinate two strings.




.charAt(index): returns the character that is present at the index given to .charAt( ) as input.


Program:

```
public class StringMethodsDemo
{
public static void main(String []args)
{
String str = "Kodnest Tech";
String s1 = str + "Pvt Ltd";
char ch = str.charAt(5);
System.out.println(s1);
System.out.println(ch);
}}
```

Output:

```
Kodnest Tech Pvt Ltd
s
```