StringBuilder:-

Java StringBuilder class is used to create mutable (modifiable) string. The Java StringBuilder class is same as StringBuffer class except that it is non-synchronized. It is available since JDK 1.5.

Important Constructors of StringBuilder class

Constructor	Description
StringBuilder()	creates an empty string buffer with the initial capacity of 16.
StringBuilder(String str)	creates a string buffer with the specified string.
StringBuilder(int capacity)	creates an empty string buffer with the specified capacity as length.

Important methods of StringBuilder class

Modifier and Type	Method	Description
public synchronized StringBuilder	append(String s)	is used to append the specified string with this string. The append() method is overloaded like append(char), append(boolean), append(int), append(float), append(double) etc.
public synchronized StringBuilder	insert(int offset, String s)	is used to insert the specified string with this string at the specified position. The insert() method is overloaded like insert(int, char), insert(int, boolean), insert(int, int), insert(int, float), insert(int, double) etc.
public synchronized StringBuilder	replace(int startIndex, int endIndex, String str)	is used to replace the string from specified startIndex and endIndex.
public synchronized StringBuilder	delete(int startIndex, int endIndex)	is used to delete the string from specified startIndex and endIndex.
public synchronized StringBuilder	reverse()	is used to reverse the string.
public int	capacity()	is used to return the current capacity.
public void	ensureCapacity(int minimumCapacity)	is used to ensure the capacity at least equal to the given minimum.
public char	charAt(int index)	is used to return the character at the specified position.

Chandrakant Sir (M) 7744822228/7743822228
Janaee Plaza 2nd Floor, Bhande Plot Sq., Umred Road, Nagpur 440024

public int	length()	is used to return the length of the string i.e. total number of characters.
public String	substring(int beginIndex)	is used to return the substring from the specified beginIndex.
public String	substring(int beginIndex, int endIndex)	is used to return the substring from the specified beginIndex and endIndex.

What is mutable string

A string that can be modified or changed is known as mutable string. StringBuilder and StringBuilder classes are used for creating mutable string.

1) StringBuilder append() method

The append() method concatenates the given argument with this string.

```
class StringBuilderExample{

public static void main(String args[]){

StringBuilder sb=new StringBuilder("Hello ");

sb.append("Java");//now original string is changed

System.out.println(sb);//prints Hello Java

}

}
```

2) StringBuilder insert() method

The insert() method inserts the given string with this string at the given position.

```
class StringBuilderExample2{

public static void main(String args[]){

StringBuilder sb=new StringBuilder("Hello ");

sb.insert(1,"Java");//now original string is changed

System.out.println(sb);//prints HJavaello

} }
```

3) StringBuilder replace() method

The replace() method replaces the given string from the specified beginIndex and endIndex.

```
class StringBuilderExample3{

public static void main(String args[]){

StringBuilder sb=new StringBuilder("Hello");

sb.replace(1,3,"Java");

System.out.println(sb);//prints HJavalo

}

}
```

4) StringBuilder delete() method

The delete() method of StringBuilder class deletes the string from the specified beginIndex to endIndex.

```
class StringBuilderExample4{

public static void main(String args[]){

StringBuilder sb=new StringBuilder("Hello");

sb.delete(1,3);

System.out.println(sb);//prints Hlo

} }
```

5) StringBuilder reverse() method

The reverse() method of StringBuilder class reverses the current string.

```
class StringBuilderExample5{

public static void main(String args[]){

StringBuilder sb=new StringBuilder("Hello");

sb.reverse();

System.out.println(sb);//prints olleH

} }
```

6) StringBuilder capacity() method

The capacity() method of StringBuilder class returns the current capacity of the buffer. The default capacity of the buffer is 16. If the number of character increases from its current capacity, it increases the capacity by (oldcapacity*2)+2. For example if your current capacity is 16, it will be (16*2)+2=34.

```
class StringBuilderExample6{

public static void main(String args[]){

StringBuilder sb=new StringBuilder();

System.out.println(sb.capacity());//default 16

sb.append("Hello");

System.out.println(sb.capacity());//now 16

sb.append("java is my favourite language");

System.out.println(sb.capacity());//now (16*2)+2=34 i.e (oldcapacity*2)+2

}

}
```