String Handling

String is a class in java.lang package.In java all classes are treated as data types.So String is a data type as well as a class.There are three ways to create Strings in Java.

*just by assigning a group of characters to a string type variable.
String s; S="Hai"; or String s="Hai";

*by allocating memory using new operator
String s=new String("Hai");

*by converting the character arrays into Strings
char arr[]={'H' , 'a' ,'i' };

Now to create a string object by passing the array name to it can be done as;
String s=new String(arr);

Some methods in String class

String method	Description	Example
String concat(String s)	Concatenates two strings	String s3=s1.concat(s2)
int length()	Returns number of characters of a string	int len=length(s1)
char charAt(int i)	Returns the character at the specified location i	S1.charAt(5)
int compareTo(String S)	Returns 0 if s1=s2 ,+ve value if s1>s2 and –ve value if s1<s2	S1.compareTo(s2)
int compareToIgnoreCase(String s)	Same as compareTo method but do not take the case of strings into consideration	S1.compareToIgnoreCase(s2)
boolean equals(String s)	Returns true if s1=s2 otherwise false	S1.equals(s2)
boolean equalsIgnoreCase(String s)	Same as equals mehod but not case sensitive	S1.equalsIgnoreCase(s2)
String replace(char c1,char c2)	Replaces all occurrences of c1 with c2	S1.replace('x','y')
String substring(int i)	Extracts substring from a main string staring from ith position	S1.substring(5)
String toLowerCase()	Converts string to lowercase	S1.toLowerCase()
String toUpperCase()	Converts string to uppercase	S1.toUpperCase()
String trim()	Removes spaces from the beginning and ending	S1.trim()

String class objects are immutable and hence their contents cannot be modified .There is a class

StringBuffer which is mutable so that they can be modified .Moreover the methods that directly ,manipulate data of the object are not available in String class but they are there in StringBuffer.

Creating StringBuffer objects--There are two ways to create a StringBuffer object,and fill the object with a string.

1.By creating a StringBuffer object by using a new operator and pass the string to the object as;

StringBuffer sb=new Stringbuffer("Hello");

2.By first allotting memory to the StringBuffer object using new operator and later storing the string to it as;

StringBuffer sb=new StringBuffer()------Here a StringBuffer object will be created with a default capacity of 16 characters

StringBuffer sb=new StringBuffer(50)-----StringBuffer object is created as an empty object with a capacity of storing 50 characters .

Some StringBuffer class methods are given below

StringBuffer method	Description	Example
StringBuffer append(x)	X may be any data type,it will be added to StringBuffer object	StringBuffer sb=new StringBuffer("Uni"); Sb.append("versity");
StringBuffer insert(int i,x)	X may be any data type,it will be inserted to StringBuffer object	StringBuffer sb=new StringBuffer("Christ University"); Sb.insert(5"Deemed");
StringBuffer reverse()	Reverses the content	Sb.reverse()
String toString()	Converts StringBuffer object to String object	Sb.toString()

Example program

//Example 1

//Java application to check whether a String is a palindrome or not using inbuilt classes and methods….Try to do the same without using inbuilt classes and methods

```
import java.io.*;
class palindrome
{
public static void main(String args[])throws IOException
                {
String s1;
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("Enter the string...:");
s1=br.readLine();      // reading a string and assigning to  s1
String temp=s1;        // storing s1 in temp
StringBuffer sb=new StringBuffer(s1);   // making a StringBuffer object of s1 to use reverse()
                                            which is in StringBuffer class
sb.reverse(); // reversing the string
s1=sb.toString();   // converting back to String from StringBuffer
```

```
if(s1.equalsIgnoreCase(temp)) // comparing s1 and temp ignoring the case
System.out.println("Its a palindrome");
else
System.out.println("Its  not a palindrome");
                }
}
```

Example program:2 To read an integer,a string and a character from keyboard

```
import java.io.*;
class inputs
{
public static void main(String args[])throws IOException
                {
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

System.out.println("Enter a string:");
String str=br.readLine();
System.out.println("Enter an integer:");
int integer=Integer.parseInt(br.readLine());
System.out.println("Enter a character:");
char character=(char)br.read();

System.out.println ("Entered  string is   :"+str);
System.out.println ("Entered  integer is  :"+integer);
System.out.println ("Entered  character is:"+character);

                }
}
```

Demonstration of Math class

```
import java.io.*;
class Mathlp
{
        public static void main(String args[]) throws IOException
        {

        char ch;
        do
        {
        BufferedReader br=new BufferedReader(new
InputStreamReader(System.in));
        System.out.println("Enter the choice");
        int c=Integer.parseInt(br.readLine());
        System.out.println("Enter the number");
        int a=Integer.parseInt(br.readLine());
        switch(c)
        {
        case 1: System.out.println("The square root for the given number is
"+Math.sqrt(a));
```

```
                break;
        case 2: System.out.println("Enter the  power");
                int b=Integer.parseInt(br.readLine());
                System.out.println("The power of the given number is
"+Math.pow(a,b));
        break;

        }
        System.out.println("Do u wish to continue");
         ch=(char)br.read();
        }while(ch=='y');
        }
}
```