DAY-27

------

EXAMPLE:1

----------

class Demo

{

public static void main(String[] args)

{

String s1 = new String("sagar");

s1.concat("ram");

System.out.println(s1);

}

}

OUTPUT:

-------

sagar

EXAMPLE:2

---------

class Demo

{

public static void main(String[] args)

{

String s1 = "sagar";

s1.concat("ram");

System.out.println(s1);

}

}

OUTPUT:

-------

sagar

intern() method:

----------------

It is used to create the copy of string object in the constant pool with the help of non-constant pool reference.

EXAMPLE:1

--------

class Demo

{

public static void main(String[] args)

{

String s1 = new String("sagar");

String s2 = s1.intern();

System.out.println(s1);

System.out.println(s2);

System.out.println(s1 == s2);

}

}

OUTPUT:

-------

sagar

sagar

false

EXAMPLE:2

----------

class Demo

{

public static void main(String[] args)

{

String s1 = "sagar";

String s2 = new String("sagar");

String s3 = s2.intern();

System.out.println(s1);

System.out.println(s2);

System.out.println(s3);

System.out.println(s1 == s2);

System.out.println(s2 == s3);

System.out.println(s1 == s3);

}

}

OUTPUT:

-------

sagar

sagar

sagar

false

false

true

trim() method.

--------------

It is used to remove the blank spaces in the begging and ending of the string.However it will not remove the spaces in b/w the strings.

EXAMPLE:1

---------

class Demo

{

public static void main(String[] args)

{

String s1 = " Study Online ";

System.out.println(s1.trim());

}

}

OUTPUT:

-------

Study Online

charAt() method.

-----------------

The character in the string cannot be accessed directly.It can be accessed only using charAt() method.

EXAMPLE:1

---------

class Demo

{

public static void main(String[] args)

{

String s1 = "Study Online";

System.out.println(s1[2]);

}

}

OUTPUT:

-------

COMPILATION ERROR

EXAMPLE:2

---------

class Demo

{

public static void main(String[] args)

{

String s1 = "Study Online";

System.out.println(s1.charAt(2));

}

}

OUTPUT:

-------

u

compareTo() method.

-------------------

refer diagram:

It is used to compare two strings lexio graphically ie each character in both the strings are converted in to unicode and compared.

when this method is used result will be in fallowing way:

if the res is zero --> strings are equal

if the res is positive --> string s1 is greater than string s2

if the res is negative --> String s1 is lesser than String s2

EXAMPLE:

--------

class Demo

{

public static void main(String[] args)

{

String s1 = "abhilash";

String s2 = "asha";

int res = s1.compareTo(s2);

if(res == 0)

{

System.out.println("Strings are equal");

}

else if (res > 0)

{

System.out.println("String s1 is greater than String s2");

}

else

{

System.out.println("String s1 is lesser than String s2");

}

}

}

OUTPUT:

-------

String s1 is lesser than String s2.

constructors of String class.

-----------------------------

1. String s1 = new String(); --> for empty string

2. String s1 = new String("String value"); -->String s1 = new String("sagar");

3.String s1 = new String(StringBuffer sb);

4.char c[] = {"a","b","c","d"}; -->String s1 = new String(c);

5.byte b[] = {97,98,99,100}; --> String s1 = new String(b)