**STRING**

String str1=” Hello ”;

String str2=” Hello ”;

String str3=new String (“Hello”);

//here string str1=str2

//for comparing content => str3.equals(str2);

//compare content only

String str="Hello";

String str1="Hello";

String str2=**new** String("Hello");

System.***out***.println(str1==str);

System.***out***.println(str1==str2);

Output=> True

False

Q -> HOW to access String as Array

Str.charAt(i);

//but we can not do =>

Str.charAt(i) = ‘z’;

//give error

**Strings are immutable in java**

i.e once value is defined => later it can not be changed

String st1=”Hello”;

Str1=”abc”;

Here only reference of str1 is changed from reference of Hello to reference of abc but value still remain same

And strings are still immutable

* == compare references
* Equals compare content
* **char** ch = str1.charAt(1);

System.***out***.println(ch);

// it is used to get the char at ith index

* System.***out***.println(str2.length());
* It is used to get length of a string

=============================================

* System.***out***.println(str2.substring(2, 9));

(it has both begin and end index)

* substring is used to get string from

starting end to endingIndex-1

* System.***out***.println(str2.substring(2));

(it has only begin index)

==========================

If you try to access index beyond length

Then you will get => StringIndexOutOfBoundIndex

===========================

s3=s.concat(s3);

System.***out***.println(s3);

It Is used to concat two strings

==========================================

Scanner sc = **new** Scanner(System.***in***);

String str=sc.next();

System.***out***.println(str);

It is used to get one word only

String str=sc.nextLine();

It is used to get one line only

Scanner sc = **new** Scanner(System.***in***);

String str=sc.next();

**int** k=0;

**for**(**int** i=0;i<str.length()/2;i++)

{

**if**(str.charAt(i)!=str.charAt(str.length()-i-1))

k=1;

}

**if**(k==0) System.***out***.println("pallindrome");

**else**

System.***out***.println("not a pallindrome");