

**1.Write a simple String program to take input from user.**

=>Here is a simple String program to take input from user –

public class ToTakeInput {  
 public static void main(String[] args) {  
 Scanner sc=new Scanner(System.*in*);  
 System.*out*.print("Enter a String : ");  
 String str=sc.nextLine();  
 System.*out*.print("You have entered : "+str);  
 }  
}

**2.How do you concatenate two strings in Java? Give an example.**

=> Concatenation is the process of combining two or more strings into a single string. This can be done in multiple ways, including using the "+" operator or the concat() method.

**Example 1:**  
Using the "+" operator:   
String str1 = "Hello";   
String str2 = "World";   
String str3 = str1 + " " + str2;

System.out.println(str3);

**Output**: Hello World

**Example 2**:  
Using the concat() method:   
String str1 = "Hello";   
String str2 = "World";   
String str3 = str1.concat(" ").concat(str2);

System.out.println(str3);

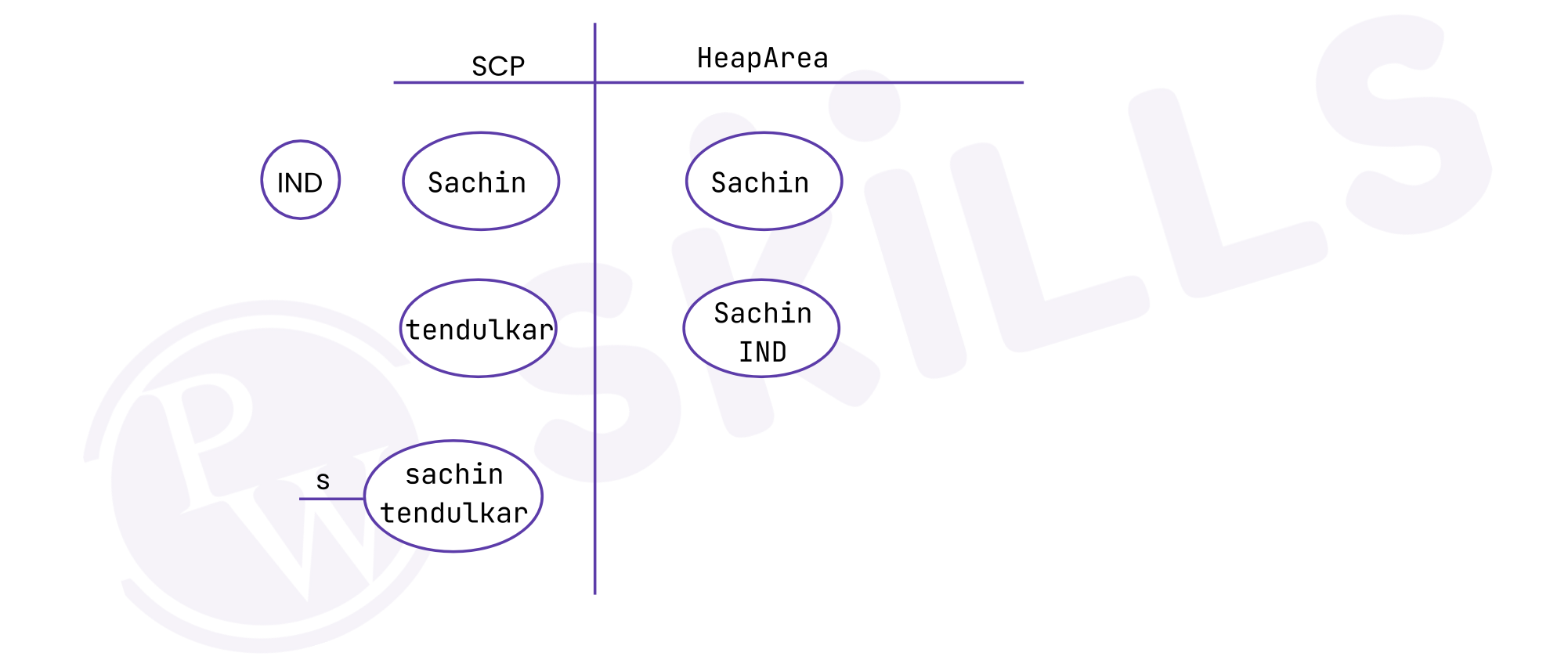
**Output:** Hello World

**Example3:**   
String s = new String("sachin"); s.concat("tendulkar");

s=s.concat("IND");

s="sachintendulkar";

**Output**   
Direct literals are always placed in SCP. Because of runtime operations, if an object is required to be created compulsorily, that object should be placed on the heap, not in SCP.

  
**Example 4:**   
String s1= new String("sachin");

s1.concat("tendulkar");

s1+="IND";   
String s2=s1.concat("MI");

System.out.println(s1) ;

**We can also concatenate 2 strings without using library function-**

String s1 = "Dipayan";  
String s2 =" Rana";  
  
for(int i=0; i<s2.length(); i++) {  
 s1 = s1 + s2.charAt(i);  
}  
System.*out*.println(s1);

**Ouput:**Dipayan Rana

**3.How do you find the length of a string in Java Explain with an example?**

=> These following steps to find the length of a String in Java:

1. Declare a variable of type String
2. Initialize the String variable to a non-null value
3. Call the Java String length() method
4. Hold the value of the String length in a variable for future use.

***Example***

public class Main {  
 public static void main(String[] args) {

String str = "Dipayan Rana";  
 int stringSize= str.length();  
 System.*out*.println(stringSize);  
  
 }  
}

***Output:***

12

**4.How do you compare two strings in Java? Give an Example.**

=> We can compare String in Java on the basis of content and reference.

There are three ways to compare String in Java:

1. By Using equals() Method
2. By Using == Operator
3. By compareTo() Method

## 1) By Using equals() Method

The String class equals() method compares the original content of the string. It compares values of string for equality. String class provides the following two methods:

* **equals()** compares this string to the specified content.
* **equalsIgnoreCase()** compares this string to another string, ignoring case.

**Example:**

**class** Teststringcomparison{

**public** **static** **void** main(String args[]){

   String s1="Sachin";

   String s2="SACHIN";

   System.out.println(s1.equals(s2));//false

   System.out.println(s1.equalsIgnoreCase(s2));//true

 }

}

## 2) By Using == operator

The == operator compares references not values.

**Example:**

**class** Teststringcomparison{

**public** **static** **void** main(String args[]){

   String s1="Sachin";

    String s2="Sachin";

   String s3=**new** String("Sachin");

   System.out.println(s1==s2);//true (because both refer to same instance)

   System.out.println(s1==s3);//false(because s3 refers to instance created in non-SCP)

 }

}

## 3) By Using compareTo() method

The String class compareTo() method compares values lexicographically and returns an integer value that describes if first string is less than, equal to or greater than second string.

Suppose s1 and s2 are two String objects. If:

* **s1 == s2** : The method returns 0.
* **s1 > s2** : The method returns a positive value.
* **s1 < s2** : The method returns a negative value.

**Example:**

**class** Teststringcomparison{

**public** **static** **void** main(String args[]){

   String s1="Sachin";

    String s2="Sachin";

    String s3="Ratan";

   System.out.println(s1.compareTo(s2));//0

  System.out.println(s1.compareTo(s3));//1(because s1>s3)

   System.out.println(s3.compareTo(s1));//-1(because s3 < s1 )

 }

}

**5.Write a program to find the length of the string "refrigerator".**

=> public class Main {  
 public static void main(String[] args) {

String str = "Dipayan Rana";  
 int stringSize= str.length();  
 System.*out*.println(stringSize);  
  
 }  
}

***Output:***

12

**6.Write a program to check if the letter 'e' is present in the word 'Umbrella'.**

=> public class CheckLetter {  
 public static void main(String[] args) {  
 String str = "Umbrella";  
 boolean n = false;  
 for(int i = 0;i<str.length();i++)  
 {  
 if(str.charAt(i) == 'e')  
 {  
 n=true;  
 break;  
 }  
 }  
 System.*out*.println(n);  
 }  
 }

**7.Write a program to delete all consonants from the string "Hello, have a good day".**

=>public class DeleteAllConsonents {  
 public static void main(String[] args) {  
 String s1="Hello, have a good day";  
 String s2="";  
 for (int i = 0; i < s1.length(); i++) {  
   
 if(s1.charAt(i)!='B' && s1.charAt(i)!='b'  
 && s1.charAt(i)!='C' && s1.charAt(i)!='c'  
 && s1.charAt(i)!='D' && s1.charAt(i)!='d'  
 && s1.charAt(i)!='F' && s1.charAt(i)!='f'  
 && s1.charAt(i)!='G' && s1.charAt(i)!='g'  
 && s1.charAt(i)!='H' && s1.charAt(i)!='h'  
 && s1.charAt(i)!='J' && s1.charAt(i)!='j'  
 && s1.charAt(i)!='K' && s1.charAt(i)!='k'  
 && s1.charAt(i)!='L' && s1.charAt(i)!='l'  
 && s1.charAt(i)!='M' && s1.charAt(i)!='m'  
 && s1.charAt(i)!='N' && s1.charAt(i)!='n'  
 && s1.charAt(i)!='P' && s1.charAt(i)!='p'  
 && s1.charAt(i)!='Q' && s1.charAt(i)!='q'  
 && s1.charAt(i)!='R' && s1.charAt(i)!='r'  
 && s1.charAt(i)!='S' && s1.charAt(i)!='s'  
 && s1.charAt(i)!='T' && s1.charAt(i)!='t'  
 && s1.charAt(i)!='V' && s1.charAt(i)!='v'  
 && s1.charAt(i)!='W' && s1.charAt(i)!='w'  
 && s1.charAt(i)!='X' && s1.charAt(i)!='x'  
 && s1.charAt(i)!='Y' && s1.charAt(i)!='y'){  
  
 s2=s2+s1.charAt(i);  
 }  
 }  
 System.*out*.println(s2);  
 }  
}

**Output-** eo, ae a oo a