String in Java

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

   Answer:

   Program:

```
import java.util.Scanner;

public class StringEx1 {
    public static void main (String [] args) {
        Scanner sc = new Scanner (System.in);
        System.out.print("Enter your name: ");
        String name = sc.nextLine();
        sc.close();
        System.out.print("Your name is: "+name);
    }
}
```

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

   Answer: We can concatenate two string with the help of 'concat()' method or with the use of '+'(+=) operator.

   Program:

```
public class Concatenation {
    public static void main (String [] args) {
        String str1 = "Rajan";
        String str2 = " Rajbhar";
// By using + operator-

        String name = str1 + str2;
        System.out.println(name);

        name = str1 + str2 + " From" + " Azamgarh";
        System.out.println(name);

// By using concat() method-

        String str = "Pw ";
        str = str.concat("Skills");
        System.out.println(str);

    }
  }
```

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

Answer: To find the length of a string in java we use an inbuilt **length()** method of the Java String class.
In Java, strings are objects created using the string class and the **length()** method is a public member method of this class. So, any variable of type **String** can access this method using the .(dot) operator.

The **length()** method counts the total number of characters in a String.

The signature of the string length () method is given below:

- public int length ()

Program:

```
public class Length {

    public static void main(String[] args) {

        String str = "Rajan Rajbhar";

        int len = str.length(); // Spces are also counts.

         System.out.println("The length of the string "+str+ " is: "+len);

    }

}
```

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

Answer:

Below are 5 ways to compare two Strings in Java:

I. **Using user-defined function:** Define a function to compare values with following conditions:
   1. if (string1 > string2) it returns a **positive value**.
   2. if both the strings are equal lexicographically
      i.e. (string1 == string2) it returns **0**.

3. if (string1 < string2) it returns a **negative value**.

The value is calculated as **(int)str1.charAt(i) – (int)str2.charAt(i)**

II. **Using String.equals**(): In Java, string equals() method compares the two given strings based on the data/content of the string. If all the contents of both the strings are same then it returns true. If any character does not match, then it returns false.

Syntax:

- str1.equals(str2);

Here str1 and str2 both are the strings which are to be compared.

III. **Using String.equalsIgnoreCase():** The String.equalsIgnoreCase() method compares two strings irrespective of the case (lower or upper) of the string. This method returns true if the argument is not null and the contents of both the Strings are same ignoring case, else false.

Syntax:

- str2.equalsIgnoreCase(str1);

Here str1 and str2 both are the strings which are to be compared.

IV. **Using Objects.equals():** Object.equals(Object a, Object b) method returns true if the arguments are equal to each other and false otherwise. Consequently, if both arguments are null, true is returned and if exactly one argument is null, false is returned. Otherwise, equality is determined by using the equals() method of the first argument.

Syntax:

- public static boolean equals(Object a, Object b)

Here a and b both are the string objects which are to be compared.

V. Using String.compareTo():

Syntax:

- int str1.compareTo(String str2)

Working:

It compares and returns the following values as follows:

- if (string1 > string2) it returns a positive value.
- if both the strings are equal lexicographically
  i.e. (string1 == string2) it returns 0.

- if (string1 < string2) it returns a negative value.

Program:

```
      import java.util.Objects;


public class Compare {

    public static void main(String[] args) {

        String str1 = new String("Java");

        String str2 = new String("Java");

        String str3 = "Java";

        String str4 = "Java";



        String str5 = "";

        String str6 = "";

        String str7 = "java";


 // Using == operator

        System.out.println(str1 == str2); // false because both s1 and s2
refers to different objetcs

        System.out.println(str3 == str4); // true because str3 and str4
refers same objects value

        System.out.println("--------------------------------------");


// Using  .equals() method

        System.out.println(str1.equals(str2)); // true because .equals() method
compare the value of the object.

        System.out.println(str3.equals(str4)); // true
        System.out.println("------------------------------------");


// Using .equalsIgnorCase() method


        System.out.println(str1.equalsIgnoreCase(str7)); //true

        System.out.println("------------------------------------");


// Using Objects.equals(Object a, Object b)

        System.out.println(Objects.equals(str1, str2)); // true

        System.out.println(Objects.equals(str1, str3)); // true

        System.out.println(Objects.equals(str3, str4)); // true

        System.out.println(Objects.equals(str5, str6)); // true

        System.out.println(Objects.equals(str1, str7)); // false

        System.out.println(Objects.equals(str1, str6)); // false


// Using compareTo() method

        System.out.println(str1.compareTo(str2)); // positive value

        System.out.println(str1.compareTo(str3)); // positive value

        System.out.println(str1.compareTo(str7)); // negative value

   }

}
```

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

   Program:

```
public class Length {
    public static void main(String[] args) {
        String str = "refrigeratorr";
        int len = str.length(); // Spces are also counts.
        System.out.println("The length of the string "+str+ " is: "+len);
    }
}
```

Output: 13

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

Program:

```
public class FindChar {
    public static void main (String[] args) {
        String str = "umbrella";
        char ch = 'e';
        for (int i = 0; i<str.length(); i++){
            if(str.charAt(i) == ch){
                System.out.println("Letter is present.");
                break;
            }
        }
    }
}
```

Output: Letter is present.

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

Program:

```
public class RmoveCons {
    public static void main(String[] args) {
        String str = "Hello, have a good day";
        String newstr = "";
        for(int i = 0; i<str.length(); i++) {
            if (str.charAt(i)=='a' || str.charAt(i)=='e' || str.charAt(i)=='i' ||
str.charAt(i)=='o' || str.charAt(i)=='u' || str.charAt(i)=='A' ||
str.charAt(i)=='E' || str.charAt(i)=='I' || str.charAt(i)=='O' ||
str.charAt(i)=='a')
        {
                newstr = newstr + str.charAt(i);
            }
        }
        System.out.println(newstr);
    }
}
```

OR

Program:

```
public class RmoveCons {
    public static void main (String[] args) {
        String str = "Hello, have a good day";
        String newstr = "";
        for (int i = 0; i<str.length(); i++){
            if(str.charAt(i)=='a' || str.charAt(i)=='e' || str.charAt(i)=='i' ||
str.charAt(i)=='o' || str.charAt(i)=='u' || str.charAt(i)=='A' ||
str.charAt(i)=='E' || str.charAt(i)=='I' || str.charAt(i)=='O' ||
str.charAt(i)=='a'){
                newstr = newstr + str.charAt(i);
            }
            else {
                newstr = newstr + "_";
            }
        }
        System.out.println(newstr);
    }
}
```