8. What are different ways to reverse a string in Java?

In Java, a string is an object, which is a collection of characters. Java allows you to perform various operations on String objects. String Reverse is one of the most commonly used operations on string objects. The Java language offers five possible methods for reversing Strings. These include:

1. String Reversal using CharAt: CharAt() returns the character at the given index in a String.

Code:

import java.util.\*;

public class Main

{

public static void main(String args[])

{

String instr, revstr="";

Scanner sc=new Scanner(System.in);

System.out.println("Enter the string: ");

instr=sc.nextLine();

int length=instr.length();

for(int i=length-1; i>=0; i--)

revstr=revstr+instr.charAt(i);

System.out.println("\nReversed string: "+revstr);

}

}

Output:

Enter the string:

Scaler Academy

Reversed string: ymedacA relacS

2. String Reversal using Recursion: In essence, recursion is a function calling itself. So, we will create a method that reverses the String by recursively calling itself.

Code:

import java.util.\*;

public class Main

{

String rev(String instr)

{

if(instr.length() == 0)

return " ";

return instr.charAt(instr.length()-1) + rev(instr.substring(0,instr.length()-1));

}

public static void main(String[ ] args)

{

Main r=new Main();

Scanner sc=new Scanner(System.in);

System.out.print("Enter the string: ");

String instr=sc.nextLine();

System.out.println("Reversed String: "+ r.rev(instr)); }

}

Output:

Enter the string: Scaler

Reversed String: relacS

3. String Reversal using Reverse Iterative: Here, we will turn the given String into a Character Array using the CharArray() method.

Code:

import java.util.\*;

class Main

{

public static void main(String[] args)

{

String instr;

Scanner in=new Scanner(System.in);

System.out.println("Enter the string: ");

instr=in.nextLine();

System.out.println("\nReversed String: ");

char[] c = instr.toCharArray();

for (int i = c.length - 1; i >= 0; i--)

System.out.print(c[i]);

}

}

Output:

Enter the string:

Scaler

Reversed String:

relacS

4. String Reversal using String Builders/String Buffers: Both StringBuffer and StringBuilder have a built-in reverse() method that can be used to reverse the characters in a string.

Code1: Using StringBuffer

//using StringBuffer

import java.util.\*;

public class Main

{

public static void main(String[] args)

{

String instr;

Scanner in=new Scanner(System.in);

System.out.println("Enter the string: ");

instr=in.nextLine();

String revstr = new StringBuffer(instr).reverse().toString();

System.out.println("\nReversed String: "+ revstr);

}

}

Output:

Enter the string:

StringBuffer Example

Reversed String: elpmaxE reffuBgnirtS

Code2: Using StringBuilder

//using StringBuilder

import java.util.\*;

public class Main

{

public static void main(String[] args)

{

String instr;

Scanner in=new Scanner(System.in);

System.out.println("Enter the string: ");

instr=in.nextLine();

String revstr = new StringBuilder(instr).reverse().toString();

System.out.println("\nReversed String: "+ revstr);

}

}

Output:

Enter the string:

StringBuilder Example

Reversed String: elpmaxE redliuBgnirtS

5. Reverse the letters of the String: Unlike previous approaches, this one does not reverse the entire String. Example: Hello Scaler will be reversed into olleH relacS

Code:

import java.util.\*;

public class Main

{

public static void main(String[] args)

{

String instr;

Scanner sc=new Scanner(System.in);

System.out.println("Enter the string: ");

instr=sc.nextLine();

System.out.println("\n");

String[] strArray = instr.split(" ");

for (String temp: strArray)

{

System.out.println(temp);

}

System.out.println("\n");

for(int i=0; i<3; i++)

{

char[] s = strArray[i].toCharArray();

for (int j = s.length-1; j>=0; j--)

{

System.out.print(s[j]);

}

System.out.print(" ");

}

}

}

Output:

Enter the string:

Scaler by InterviewBit

Scaler

by

InterviewBit

relacS yb tiBweivretnI