**Program to reverse a string using methods of string**

**//title : to print reverse string with using reverse string class methods//**

**package** javaday3;

**import** java.util.\*;

**public** **class** StringRev{

// Function to reverse a string in Java using StringBuilder

**public** **static** String rev(String s){

**return** **new** StringBuilder(s).reverse().toString();

}

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

String s= "Welcome to jAVA"; // Note that string is immutable in Java

s= *rev*(s);

System.*out*.println("Result after reversing a string is : "+s);

}

}

Output :

Result after reversing a string is : AVAj ot emocleW

**Program to reverse a string without using any methods of string class.**

**//Title : to print reverse string without using any reverse string methods//**

**package** javaday3;

**import** java.util.\*;

**public** **class** StringRevss{

// Function to reverse a string in Java using StringBuilder

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

String str="Hello world";

String revstring="";

**for**(**int** i=str.length()-1;i>=0;--i){

revstring +=str.charAt(i);

}

System.*out*.println(revstring);

}

}

Output :

dlrow olleH

**Program to print alternate values of a 2-D array**

//Title : to print alternate elements in matrix//

**package** javaday3;

**import** java.util.\*;

**import** java.lang.\*;

**public** **class** Alternate {

**static** **void** printElements(**int**[][] mat, **int** n)

{

**for** (**int** i = 0; i < n; i++)

{

**if** (i % 2 == 0)

**for** (**int** j = 0; j < n; j += 2)

System.*out*.print(mat[i][j] + " ");

**else**

**for** (**int** j = 1; j < n; j += 2)

System.*out*.print(mat[i][j] + " ");

}

}

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

**int** n = 3; **int**[][] mat = **new** **int**[][]{ { 1, 5, 3 },{ 2, 4, 7 },{ 9, 8, 6 }

};

*printElements*(mat, n);

}

}

Output :

1 3 4 9 6

**What is IMMUTABLE**

In java, **string objects are immutable**. Immutable simply means unmodifiable or unchangeable.

Once string object is created its data or state can't be changed but a new string object is created.

For example, [String](https://www.journaldev.com/16928/java-string) is an immutable class and once instantiated its value never changes.

To create an immutable class in java, you have to do following steps.

1. Declare the class as final so it can’t be extended.
2. Make all fields private so that direct access is not allowed.
3. Don’t provide setter methods for variables
4. Make all mutable fields final so that it’s value can be assigned only once.
5. Initialize all the fields via a constructor performing deep copy.
6. Perform cloning of objects in the getter methods to return a copy rather than returning the actual object reference.