**Program 1: Reverse a number using while Loop**

The program will prompt user to input the number and then it will reverse the same number using while loop.

import java.util.Scanner;

class ReverseNumberWhile

{

public static void main(String args[])

{

int num=0;

int reversenum =0;

System.out.println("Input your number and press enter: ");

//This statement will capture the user input

Scanner in = new Scanner(System.in);

//Captured input would be stored in number num

num = in.nextInt();

//While Loop: Logic to find out the reverse number

while( num != 0 )

{

reversenum = reversenum \* 10;

reversenum = reversenum + num%10;

num = num/10;

}

System.out.println("Reverse of input number is: "+reversenum);

}

}

Output:

Input your number and press enter:

145689

Reverse of input number is: 986541

# Java Program to calculate area and circumference of circle

In this tutorial we will see **how to calculate area and circumference of circle in Java**. There are two ways to do this:

1) With user interaction: Program will prompt user to enter the radius of the circle  
2) Without user interaction: The radius value would be specified in the program itself.

**Program 1:**

/\*\*

\* @author: BeginnersBook.com

\* @description: Program to calculate area and circumference of circle

\* with user interaction. User will be prompt to enter the radius and

\* the result will be calculated based on the provided radius value.

\*/

import java.util.Scanner;

class CircleDemo

{

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

public static void main(String args[])

{

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

/\*We are storing the entered radius in double

\* because a user can enter radius in decimals

\*/

double radius = sc.nextDouble();

//Area = PI\*radius\*radius

double area = Math.PI \* (radius \* radius);

System.out.println("The area of circle is: " + area);

//Circumference = 2\*PI\*radius

double circumference= Math.PI \* 2\*radius;

System.out.println( "The circumference of the circle is:"+circumference) ;

}

}

Output:

Enter the radius: 1

The area of circle is: 3.141592653589793

The circumference of the circle is:6.283185307179586

# Java program to calculate area of Triangle

 Here we will see how to calculate area of triangle. We will see two following programs to do this:  
1) Program 1: Prompt user for base-width and height of triangle.  
2) Program 2: No user interaction: Width and height are specified in the program itself.

**Program 1:**

/\*\*

\* @author: BeginnersBook.com

\* @description: Program to Calculate area of Triangle in Java

\* with user interaction. Program will prompt user to enter the

\* base width and height of the triangle.

\*/

import java.util.Scanner;

class AreaTriangleDemo {

public static void main(String args[]) {

Scanner userInput = new Scanner(System.in);

System.out.println("Enter the width of the Triangle:");

double base = userInput.nextDouble();

System.out.println("Enter the height of the Triangle:");

double height = userInput.nextDouble();

//Area = (width\*height)/2

double area = (base\* height)/2;

System.out.println("Area of Triangle is: " + area);

}

}

Output:

Enter the width of the Triangle:

2

Enter the height of the Triangle:

2

Area of Triangle is: 2.0

# Java program to sum the elements of an array

 In this tutorial we will see **how to sum up all the elements of an array**.

Program 1: No user interaction

/\*\*

\* @author: BeginnersBook.com

\* @description: Get sum of array elements

\*/

class SumOfArray{

public static void main(String args[]){

int[] array = {10, 20, 30, 40, 50, 10};

int sum = 0;

//Advanced for loop

for( int num : array) {

sum = sum+num;

}

System.out.println("Sum of array elements is:"+sum);

}

}

Output:

Sum of array elements is:160

Program 2: User enters the array’s elements

/\*\*

\* @author: BeginnersBook.com

\* @description: User would enter the 10 elements

\* and the program will store them into an array and

\* will display the sum of them.

\*/

import java.util.Scanner;

class SumDemo{

public static void main(String args[]){

Scanner scanner = new Scanner(System.in);

int[] array = new int[10];

int sum = 0;

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

for (int i=0; i<10; i++)

{

array[i] = scanner.nextInt();

}

for( int num : array) {

sum = sum+num;

}

System.out.println("Sum of array elements is:"+sum);

}

}

Output:

Enter the elements:

1

2

3

4

5

6

7

8

9

10

Sum of array elements is:55

# Java program to check prime number

This program will prompt user to enter a number and then **it will check and display whether the input number is prime or not**.

import java.util.Scanner;

class PrimeCheck

{

public static void main(String args[])

{

int temp;

boolean isPrime=true;

Scanner scan= new Scanner(System.in);

System.out.println("Enter a number for check:");

//capture the input in an integer

int num=scan.nextInt();

for(int i=2;i<=num/2;i++)

{

temp=num%i;

if(temp==0)

{

isPrime=false;

break;

}

}

//If isPrime is true then the number is prime else not

if(isPrime)

System.out.println(num + " is Prime Number");

else

System.out.println(num + " is not Prime Number");

}

}

Output:

Enter a number for check:

19

19 is Prime Number

Output 2:

Enter a number for check:

6

6 is not Prime Number

# Java Program to check Even or Odd number

import java.util.Scanner;

class CheckEvenOdd

{

public static void main(String args[])

{

int num;

System.out.println("Enter an Integer number:");

//The input provided by user is stored in num

Scanner input = new Scanner(System.in);

num = input.nextInt();

/\* If number is divisible by 2 then it's an even number

\* else odd number\*/

if ( num % 2 == 0 )

System.out.println("Entered number is even");

else

System.out.println("Entered number is odd");

}

}

Output 1:

Enter an Integer number:

78

Entered number is even

Output 2:

Enter an Integer number:

77

Entered number is odd

# java program to find factorial of a given number using recursion

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

Here we will **write programs to find out the factorial of a number using recursion**.

**Program 1:**  
Program will prompt user for the input number. Once user provide the input, the program will calculate the factorial for the provided input number.

/\*\*

\* @author: BeginnersBook.com

\* @description: User would enter the 10 elements

\* and the program will store them into an array and

\* will display the sum of them.

\*/

import java.util.Scanner;

class FactorialDemo{

public static void main(String args[]){

//Scanner object for capturing the user input

Scanner scanner = new Scanner(System.in);

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

//Stored the entered value in variable

int num = scanner.nextInt();

//Called the user defined function fact

int factorial = fact(num);

System.out.println("Factorial of entered number is: "+factorial);

}

static int fact(int n)

{

int output;

if(n==1){

return 1;

}

//Recursion: Function calling itself!!

output = fact(n-1)\* n;

return output;

}

}

Output:

Enter the number:

5

Factorial of entered number is: 120

# java program to check palindrome string using Stack, Queue, for or while loop

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

In this tutorial we will see **programs to check whether the given String is Palindrome or not**. Following are the ways to do it.  
1) Using Stack  
2) Using Queue  
3) Using for/while loop

**Program 1: Palindrome check Using Stack**

import java.util.Stack;

import java.util.Scanner;

class PalindromeTest {

public static void main(String[] args) {

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

Scanner in=new Scanner(System.in);

String inputString = in.nextLine();

Stack stack = new Stack();

for (int i = 0; i < inputString.length(); i++) {

stack.push(inputString.charAt(i));

}

String reverseString = "";

while (!stack.isEmpty()) {

reverseString = reverseString+stack.pop();

}

if (inputString.equals(reverseString))

System.out.println("The input String is a palindrome.");

else

System.out.println("The input String is not a palindrome.");

}

}

Output 1:

Enter any string:abccba

The input String is a palindrome.

Output 2:

Enter any string:abcdef

The input String is not a palindrome.

**Program 2: Palindrome check Using Queue**

import java.util.Queue;

import java.util.Scanner;

import java.util.LinkedList;

class PalindromeTest {

public static void main(String[] args) {

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

Scanner in=new Scanner(System.in);

String inputString = in.nextLine();

Queue queue = new LinkedList();

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

queue.add(inputString.charAt(i));

}

String reverseString = "";

while (!queue.isEmpty()) {

reverseString = reverseString+queue.remove();

}

if (inputString.equals(reverseString))

System.out.println("The input String is a palindrome.");

else

System.out.println("The input String is not a palindrome.");

}

}

Output 1:

Enter any string:xyzzyx

xyzzyx

The input String is a palindrome.

Output 2:

Enter any string:xyz

The input String is not a palindrome.

**Program 3: Using for loop/While loop and String function charAt**

import java.util.Scanner;

class PalindromeTest {

public static void main(String args[])

{

String reverseString="";

Scanner scanner = new Scanner(System.in);

System.out.println("Enter a string to check if it is a palindrome:");

String inputString = scanner.nextLine();

int length = inputString.length();

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

reverseString = reverseString + inputString.charAt(i);

if (inputString.equals(reverseString))

System.out.println("Input string is a palindrome.");

else

System.out.println("Input string is not a palindrome.");

}

}

Output 1:

Enter a string to check if it is a palindrome:

aabbaa

Input string is a palindrome.

Output 2:

Enter a string to check if it is a palindrome:

aaabbb

Input string is not a palindrome.

If you wanna use **While Loop** in above program then replace the for loop with this code:

int i = length-1;

while ( i >= 0){

reverseString = reverseString + inputString.charAt(i);

i--;

}

# How To Convert Char To String and a String to char in Java

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

In this tutorial, we will see programs for char to String and String to char conversion.

## Program to convert char to String

We have following two ways for char to String conversion.  
Method 1: Using toString() method  
Method 2: Usng [**valueOf()**](http://beginnersbook.com/2013/12/java-string-copyvalueof-method-example/) method

class CharToStringDemo

{

public static void main(String args[])

{

// Method 1: Using toString() method

char ch = 'a';

String str = Character.toString(ch);

System.out.println("String is: "+str);

// Method 2: Using valueOf() method

String str2 = String.valueOf(ch);

System.out.println("String is: "+str2);

}

}

**Output:**

String is: a

String is: a

## Converting String to Char

We can convert a String to char using [**charAt() method**](http://beginnersbook.com/2013/12/java-string-charat-method-example/) of [**String class**](http://beginnersbook.com/2013/12/java-strings/).

class StringToCharDemo

{

public static void main(String args[])

{

// Using charAt() method

String str = "Hello";

for(int i=0; i<str.length();i++){

char ch = str.charAt(i);

System.out.println("Character at "+i+" Position: "+ch);

}

}

}

**Output:**

Character at 0 Position: H

Character at 1 Position: e

Character at 2 Position: l

Character at 3 Position: l

Character at 4 Position: o

# How to convert a char array to a string in Java?

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

There are two ways to convert a char array (char[]) to String in Java:  
1) Creating String object by passing array name to the constructor  
2) Using [**valueOf() method**](http://beginnersbook.com/2013/12/java-string-copyvalueof-method-example/) of [**String class**](http://beginnersbook.com/2013/12/java-strings/).

**Example:**  
This example demonstrates both the above mentioned ways of converting a char array to String. Here we have a char array ch and we have created two strings str and str1 using the char array.

class CharArrayToString

{

public static void main(String args[])

{

// Method 1: Using String object

char[] ch = {'g', 'o', 'o', 'd', ' ', 'm', 'o', 'r', 'n', 'i', 'n', 'g'};

String str = new String(ch);

System.out.println(str);

// Method 2: Using valueOf method

String str2 = String.valueOf(ch);

System.out.println(str2);

}

}

**Output:**

good morning

good morning

# Java program to generate random number – Example

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

#### Example Program to generate random numbers

In the below program, we are using the nextInt() method of Random class to serve our purpose.

/\* Program: Random number generator

\* Written by: Chaitanya from beginnersbook.com

\* Input: None

\* Output:Random number between o and 200\*/

import java.util.\*;

class GenerateRandomNumber {

public static void main(String[] args) {

int counter;

Random rnum = new Random();

/\* Below code would generate 5 random numbers

\* between 0 and 200.

\*/

System.out.println("Random Numbers:");

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

for (counter = 1; counter <= 5; counter++) {

System.out.println(rnum.nextInt(200));

}

}

}

Output:

Random Numbers:

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

135

173

5

17

15

# Java program to perform binary search – Example

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

#### Example Program to perform binary search on a list of integer numbers

This program uses [**binary search algorithm**](http://en.wikipedia.org/wiki/Binary_search_algorithm) to search an element in given list of elements.

/\* Program: Binary Search Example

\* Written by: Chaitanya from beginnersbook.com

\* Input: Number of elements, element's values, value to be searched

\* Output:Position of the number input by user among other numbers\*/

import java.util.Scanner;

class BinarySearchExample

{

public static void main(String args[])

{

int counter, num, item, array[], first, last, middle;

//To capture user input

Scanner input = new Scanner(System.in);

System.out.println("Enter number of elements:");

num = input.nextInt();

//Creating array to store the all the numbers

array = new int[num];

System.out.println("Enter " + num + " integers");

//Loop to store each numbers in array

for (counter = 0; counter < num; counter++)

array[counter] = input.nextInt();

System.out.println("Enter the search value:");

item = input.nextInt();

first = 0;

last = num - 1;

middle = (first + last)/2;

while( first <= last )

{

if ( array[middle] < item )

first = middle + 1;

else if ( array[middle] == item )

{

System.out.println(item + " found at location " + (middle + 1) + ".");

break;

}

else

{

last = middle - 1;

}

middle = (first + last)/2;

}

if ( first > last )

System.out.println(item + " is not found.\n");

}

}

Output 1:

Enter number of elements:

7

Enter 7 integers

4

5

66

77

8

99

0

Enter the search value:

77

77 found at location 4.

# Java program for linear search – Example

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

#### Example Program:

This program uses [**linear search algorithm**](http://en.wikipedia.org/wiki/Linear_search) to find out a number among all other numbers entered by user.

/\* Program: Linear Search Example

\* Written by: Chaitanya from beginnersbook.com

\* Input: Number of elements, element's values, value to be searched

\* Output:Position of the number input by user among other numbers\*/

import java.util.Scanner;

class LinearSearchExample

{

public static void main(String args[])

{

int counter, num, item, array[];

//To capture user input

Scanner input = new Scanner(System.in);

System.out.println("Enter number of elements:");

num = input.nextInt();

//Creating array to store the all the numbers

array = new int[num];

System.out.println("Enter " + num + " integers");

//Loop to store each numbers in array

for (counter = 0; counter < num; counter++)

array[counter] = input.nextInt();

System.out.println("Enter the search value:");

item = input.nextInt();

for (counter = 0; counter < num; counter++)

{

if (array[counter] == item)

{

System.out.println(item+" is present at location "+(counter+1));

/\*Item is found so to stop the search and to come out of the

\* loop use break statement.\*/

break;

}

}

if (counter == num)

System.out.println(item + " doesn't exist in array.");

}

}

Output 1:

Enter number of elements:

6

Enter 6 integers

22

33

45

1

3

99

Enter the search value:

45

45 is present at location 3

Output 2:

Enter number of elements:

4

Enter 4 integers

11

22

4

5

Enter the search value:

99

99 doesn't exist in array.

# Java program to print Floyd’s triangle – Example

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

#### Example Program:

This program will prompt user for number of rows and based on the input, it would print the [**Floyd’s triangle**](http://en.wikipedia.org/wiki/Floyd's_triangle) having the same number of rows.

/\* Program: It Prints Floyd's triangle based on user inputs

\* Written by: Chaitanya from beginnersbook.com

\* Input: Number of rows

\* output: floyd's triangle\*/

import java.util.Scanner;

class FloydTriangleExample

{

public static void main(String args[])

{

int rows, number = 1, counter, j;

//To get the user's input

Scanner input = new Scanner(System.in);

System.out.println("Enter the number of rows for floyd's triangle:");

//Copying user input into an integer variable named rows

rows = input.nextInt();

System.out.println("Floyd's triangle");

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

for ( counter = 1 ; counter <= rows ; counter++ )

{

for ( j = 1 ; j <= counter ; j++ )

{

System.out.print(number+" ");

//Incrementing the number value

number++;

}

//For new line

System.out.println();

}

}

}

Output:

Enter the number of rows for floyd's triangle:

6

Floyd's triangle

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1

2 3

4 5 6

7 8 9 10

11 12 13 14 15

16 17 18 19 20 21

# Java Program to display first n or first 100 prime numbers

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

**Program to display first n prime numbers**

import java.util.Scanner;

class PrimeNumberDemo

{

public static void main(String args[])

{

int n;

int status = 1;

int num = 3;

//For capturing the value of n

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the value of n:");

//The entered value is stored in the var n

n = scanner.nextInt();

if (n >= 1)

{

System.out.println("First "+n+" prime numbers are:");

//2 is a known prime number

System.out.println(2);

}

for ( int i = 2 ; i <=n ; )

{

for ( int j = 2 ; j <= Math.sqrt(num) ; j++ )

{

if ( num%j == 0 )

{

status = 0;

break;

}

}

if ( status != 0 )

{

System.out.println(num);

i++;

}

status = 1;

num++;

}

}

}

Output:

Enter the value of n:

15

First 15 prime numbers are:

2

3

5

7

11

13

17

19

23

29

31

37

41

43

47

## Program to display the prime numbers from 1 to 100

It will display the prime numbers between 1 and 100.

class PrimeNumbers

{

public static void main (String[] args)

{

int i =0;

int num =0;

//Empty String

String primeNumbers = "";

for (i = 1; i <= 100; i++)

{

int counter=0;

for(num =i; num>=1; num--)

{

if(i%num==0)

{

counter = counter + 1;

}

}

if (counter ==2)

{

//Appended the Prime number to the String

primeNumbers = primeNumbers + i + " ";

}

}

System.out.println("Prime numbers from 1 to 100 are :");

System.out.println(primeNumbers);

}

}

Output:

Prime numbers from 1 to 100 are :

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

# java program to check palindrome string using recursion

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

**Program: Check whether String is palindrome using recursion**

package beginnersbook.com;

import java.util.Scanner;

class PalindromeCheck

{

//My Method to check

public static boolean isPal(String s)

{ // if length is 0 or 1 then String is palindrome

if(s.length() == 0 || s.length() == 1)

return true;

if(s.charAt(0) == s.charAt(s.length()-1))

/\* check for first and last char of String:

\* if they are same then do the same thing for a substring

\* with first and last char removed. and carry on this

\* until you string completes or condition fails

\* Function calling itself: Recursion

\*/

return isPal(s.substring(1, s.length()-1));

/\* If program control reaches to this statement it means

\* the String is not palindrome hence return false.

\*/

return false;

}

public static void main(String[]args)

{

//For capturing user input

Scanner scanner = new Scanner(System.in);

System.out.println("Enter the String for check:");

String string = scanner.nextLine();

/\* If function returns true then the string is

\* palindrome else not

\*/

if(isPal(string))

System.out.println(string + " is a palindrome");

else

System.out.println(string + " is not a palindrome");

}

}

Output:

Enter the String for check:

qqaabb

qqaabb is not a palindrome

Output 2:

Enter the String for check:

cocoococ

cocoococ is a palindrome

# Java Program to find duplicate Characters in a String

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

This program would find out the duplicate characters in a String and would display the count of them.

import java.util.HashMap;

import java.util.Map;

import java.util.Set;

public class Details {

public void countDupChars(String str){

//Create a HashMap

Map<Character, Integer> map = new HashMap<Character, Integer>();

//Convert the String to char array

char[] chars = str.toCharArray();

/\* logic: char are inserted as keys and their count

\* as values. If map contains the char already then

\* increase the value by 1

\*/

for(Character ch:chars){

if(map.containsKey(ch)){

map.put(ch, map.get(ch)+1);

} else {

map.put(ch, 1);

}

}

//Obtaining set of keys

Set<Character> keys = map.keySet();

/\* Display count of chars if it is

\* greater than 1. All duplicate chars would be

\* having value greater than 1.

\*/

for(Character ch:keys){

if(map.get(ch) > 1){

System.out.println("Char "+ch+" "+map.get(ch));

}

}

}

public static void main(String a[]){

Details obj = new Details();

System.out.println("String: BeginnersBook.com");

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

obj.countDupChars("BeginnersBook.com");

System.out.println("\nString: ChaitanyaSingh");

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

obj.countDupChars("ChaitanyaSingh");

System.out.println("\nString: #@$@!#$%!!%@");

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

obj.countDupChars("#@$@!#$%!!%@");

}

}

Output:

String: BeginnersBook.com

-------------------------

Char e 2

Char B 2

Char n 2

Char o 3

String: ChaitanyaSingh

-------------------------

Char a 3

Char n 2

Char h 2

Char i 2

String: #@$@!#$%!!%@

-------------------------

Char # 2

Char ! 3

Char @ 3

Char $ 2

Char % 2

# Java Program to get input from user

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

In this tutorial we are gonna see **how to accept input from user**. We are using Scanner class to get the input. In the below example we are getting input String, integer and a float number. For this we are using following methods:  
1) public String nextLine(): For getting input String  
2) public int nextInt(): For integer input  
3) public float nextFloat(): For float input

#### Example:

import java.util.Scanner;

class GetInputData

{

public static void main(String args[])

{

int num;

float fnum;

String str;

Scanner in = new Scanner(System.in);

//Get input String

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

str = in.nextLine();

System.out.println("Input String is: "+str);

//Get input Integer

System.out.println("Enter an integer: ");

num = in.nextInt();

System.out.println("Input Integer is: "+num);

//Get input float number

System.out.println("Enter a float number: ");

fnum = in.nextFloat();

System.out.println("Input Float number is: "+fnum);

}

}

**Output:**

Enter a string:

Chaitanya

Input String is: Chaitanya

Enter an integer:

27

Input Integer is: 27

Enter a float number:

12.56

Input Float number is: 12.56

# Java program to get IP address

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

In this example we are gonna see how to get IP address of a System. The steps are as follows:

1) Get the local host address by calling getLocalHost() method of InetAddress class.  
2) Get the IP address by calling getHostAddress() method.

import java.net.InetAddress;

class GetMyIPAddress

{

public static void main(String args[]) throws Exception

{

/\* public static InetAddress getLocalHost()

\* throws UnknownHostException: Returns the address

\* of the local host. This is achieved by retrieving

\* the name of the host from the system, then resolving

\* that name into an InetAddress. Note: The resolved

\* address may be cached for a short period of time.

\*/

InetAddress myIP=InetAddress.getLocalHost();

/\* public String getHostAddress(): Returns the IP

\* address string in textual presentation.

\*/

System.out.println("My IP Address is:");

System.out.println(myIP.getHostAddress());

}

}

**Output:**

My IP Address is:

115.242.7.243

# Java program to convert decimal to binary

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

There are three following ways to convert Decimal number to binary number:

1) Using [**toBinaryString() method**](http://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html#toBinaryString(int)) of Integer class.  
2) Do conversion by writing your own logic without using any predefined methods.  
3) Using Stack

### Method 1: Using toBinaryString() method

class DecimalBinaryExample{

public static void main(String a[]){

System.out.println("Binary representation of 124: ");

System.out.println(Integer.toBinaryString(124));

System.out.println("\nBinary representation of 45: ");

System.out.println(Integer.toBinaryString(45));

System.out.println("\nBinary representation of 999: ");

System.out.println(Integer.toBinaryString(999));

}

}

**Output:**

Binary representation of 124:

1111100

Binary representation of 45:

101101

Binary representation of 999:

1111100111

### Method 2: Without using predefined method

class DecimalBinaryExample{

public void convertBinary(int num){

int binary[] = new int[40];

int index = 0;

while(num > 0){

binary[index++] = num%2;

num = num/2;

}

for(int i = index-1;i >= 0;i--){

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

}

}

public static void main(String a[]){

DecimalBinaryExample obj = new DecimalBinaryExample();

System.out.println("Binary representation of 124: ");

obj.convertBinary(124);

System.out.println("\nBinary representation of 45: ");

obj.convertBinary(45);

System.out.println("\nBinary representation of 999: ");

obj.convertBinary(999);

}

}

**Output:**

Binary representation of 124:

1111100

Binary representation of 45:

101101

Binary representation of 999:

1111100111

### Method 3: Decimal to Binary using Stack

import java.util.\*;

class DecimalBinaryStack

{

public static void main(String[] args)

{

Scanner in = new Scanner(System.in);

// Create Stack object

Stack<Integer> stack = new Stack<Integer>();

// User input

System.out.println("Enter decimal number: ");

int num = in.nextInt();

while (num != 0)

{

int d = num % 2;

stack.push(d);

num /= 2;

}

System.out.print("\nBinary representation is:");

while (!(stack.isEmpty() ))

{

System.out.print(stack.pop());

}

System.out.println();

}

}

**Output:**

Enter decimal number:

999

Binary representation is:1111100111

# Java program for decimal to octal conversion

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

In this tutorial we will learn following two ways to **convert a decimal number to equivalent octal number**.

1) Using predefined method [**Integer.toOctalString(int num)**](http://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html#toOctalString(int))  
2) Writing our own logic for conversion

import java.util.Scanner;

class DecimalToOctalExample

{

public static void main(String args[])

{

Scanner input = new Scanner( System.in );

System.out.print("Enter a decimal number : ");

int num =input.nextInt();

/\* Method 1:

\* Using predefined method toOctalString(int)

\* Pass the decimal number to this method and

\* it would return the equivalent octal number

\*/

String octalString = Integer.toOctalString(num);

System.out.println("Method 1: Decimal to octal: " + octalString);

/\* Method 2:

\* Writing your own logic: Here we will write

\* our own logic for decimal to octal conversion

\*/

// For storing remainder

int rem;

// For storing result

String str="";

// Digits in Octal number system

char dig[]={'0','1','2','3','4','5','6','7'};

while(num>0)

{

rem=num%8;

str=dig[rem]+str;

num=num/8;

}

System.out.println("Method 2: Decimal to octal: "+str);

}

}

**Output:**

Enter a decimal number : 123

Method 1: Decimal to octal: 173

Method 2: Decimal to octal: 173

# Java program to convert decimal to hexadecimal

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

There are two following ways to convert Decimal number to hexadecimal number:

1) Using [**toHexString() method**](http://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html#toHexString(int)) of Integer class.  
2) Do conversion by writing your own logic without using any predefined methods.

### Method 1: Decimal to hexadecimal Using toHexString() method

import java.util.Scanner;

class DecimalToHexExample

{

public static void main(String args[])

{

Scanner input = new Scanner( System.in );

System.out.print("Enter a decimal number : ");

int num =input.nextInt();

// calling method toHexString()

String str = Integer.toHexString(num);

System.out.println("Method 1: Decimal to hexadecimal: "+str);

}

}

**Output:**

Enter a decimal number : 123

Method 1: Decimal to hexadecimal: 7b

### Method 2: Decimal to hexadecimal without using predefined method

import java.util.Scanner;

class DecimalToHexExample

{

public static void main(String args[])

{

Scanner input = new Scanner( System.in );

System.out.print("Enter a decimal number : ");

int num =input.nextInt();

// For storing remainder

int rem;

// For storing result

String str2="";

// Digits in hexadecimal number system

char hex[]={'0','1','2','3','4','5','6','7','8','9','A','B','C','D','E','F'};

while(num>0)

{

rem=num%16;

str2=hex[rem]+str2;

num=num/16;

}

System.out.println("Method 2: Decimal to hexadecimal: "+str2);

}

}

**Output:**

Enter a decimal number : 123

Method 2: Decimal to hexadecimal: 7B

# Java program for binary to decimal conversion

[**JAVA EXAMPLES**](http://beginnersbook.com/category/java-examples/)

There are two following ways to convert binary number to decimal number:

1) Using [**Integer.parseInt() method**](http://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html#parseInt(java.lang.String,%20int)) of Integer class.  
2) Do conversion by writing your own logic without using any predefined methods.

### Method 1: Binary to Decimal conversion using Integer.parseInt() method

import java.util.Scanner;

class BinaryToDecimal {

public static void main(String args[]){

Scanner input = new Scanner( System.in );

System.out.print("Enter a binary number: ");

String binaryString =input.nextLine();

System.out.println("Output: "+Integer.parseInt(binaryString,2));

}

}

**Output:**

Enter a binary number: 1101

Output: 13

### Method 2: Conversion without using parseInt

public class Details {

public int BinaryToDecimal(int binaryNumber){

int decimal = 0;

int p = 0;

while(true){

if(binaryNumber == 0){

break;

} else {

int temp = binaryNumber%10;

decimal += temp\*Math.pow(2, p);

binaryNumber = binaryNumber/10;

p++;

}

}

return decimal;

}

public static void main(String args[]){

Details obj = new Details();

System.out.println("110 --> "+obj.BinaryToDecimal(110));

System.out.println("1101 --> "+obj.BinaryToDecimal(1101));

System.out.println("100 --> "+obj.BinaryToDecimal(100));

System.out.println("110111 --> "+obj.BinaryToDecimal(110111));

}

}

**Output:**

110 --> 6

1101 --> 13

100 --> 4

110111 --> 55