1. Java program to find the area of rectangle.

**package** com.rsc.babystepprogram;

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

**class** CalculateArea {

**int** length, breadth;

**public** CalculateArea(**int** length, **int** breadth) {

**this**.length = length;

**this**.breadth = breadth;

}

**public** **int** area() {

**return** **this**.length \* **this**.breadth;

}

}

**public** **class** AreaOfRectangle {

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

Scanner scan = **new** Scanner(System.***in***);

System.***out***.print("Enter the length of rectangle: ");

**int** length = scan.nextInt();

System.***out***.print("Enter the breadth of rectangle: ");

**int** breadth = scan.nextInt();

**if**( length <= 0 || breadth <= 0)

System.***out***.println("Dimensions should be positive.");

**else** {

CalculateArea rect = **new** CalculateArea(length, breadth);

System.***out***.println("Area of rectangle with length:"+length+" and breadth:"+breadth+" is "+rect.area());

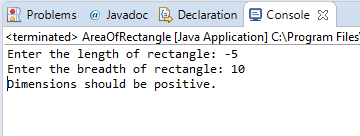
}

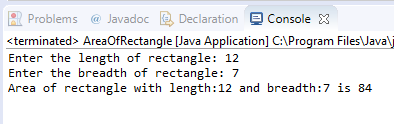
scan.close();

}

}

Output:





1. Java program to check the given no is Armstrong or not.

**package** com.rsc.babystepprogram;

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

**public** **class** ArmstrongNumber {

**public** **boolean** checkArmstrong(**int** number) {

**int** noOfDigits = 0, numberCopy = number, sum = 0;

**while**(numberCopy > 0) {

noOfDigits += 1;

numberCopy /= 10;

}

numberCopy = number;

**while**(number > 0) {

**int** remainder = number % 10;

sum += Math.*pow*(remainder, noOfDigits);

number /= 10;

}

**if**(sum == numberCopy) **return** **true**;

**else** **return** **false**;

}

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

Scanner scan = **new** Scanner(System.***in***);

System.***out***.print("Enter a number to check whether it is armstrong or not: ");

**try** {

**int** number = scan.nextInt();

**if**(number < 0)

System.***out***.println("Number must be positive.");

**else** {

ArmstrongNumber arm = **new** ArmstrongNumber();

**if**(arm.checkArmstrong(number))

System.***out***.println(number + " is an Armstrong number.");

**else**

System.***out***.println(number + " is not anArmstrong number.");

}

}

**catch**(InputMismatchException er) {

System.***out***.print("Input mismatch exception: "+er);

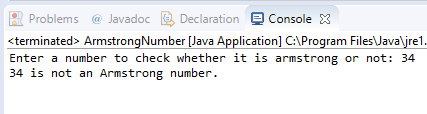
}

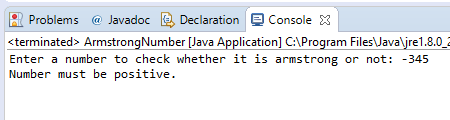
scan.close();

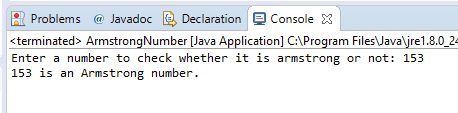
}

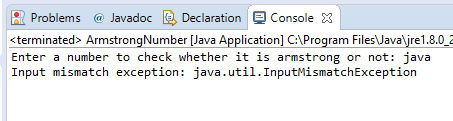
}

Output:









1. Java program to check the given no is palindrome or not

**package** com.rsc.babystepprogram;

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

**public** **class** CheckPalindrome {

**public** **boolean** isPalindrome(**int** number) {

**int** numberCopy = number, sum = 0;

**while**(number > 0) {

**int** remainder = number % 10;

sum = sum\*10 + remainder;

number /= 10;

}

**return** sum == numberCopy;

}

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

Scanner scan = **new** Scanner(System.***in***);

**try** {

System.***out***.print("Enter a number to check whether it is palindrome or not: ");

**int** number = scan.nextInt();

**if**(number < 0)

System.***out***.println("Number must be positive");

**else** {

CheckPalindrome pal = **new** CheckPalindrome();

**if**(pal.isPalindrome(number))

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

**else**

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

}

}

**catch**(InputMismatchException er) {

System.***out***.println("Input mismatch exception: "+ er);

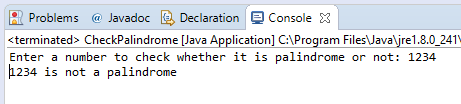
}

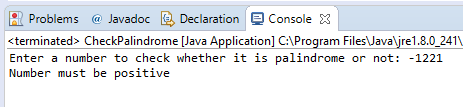
scan.close();

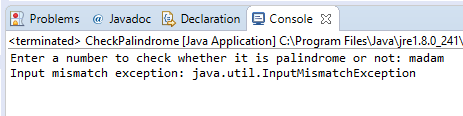
}

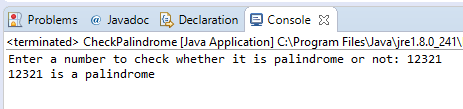
}

Output:









1. Java program to generate first N prime numbers

**package** com.rsc.babystepprogram;

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

**public** **class** PrimesUptoN {

**public** **int**[] primes(**int** number) {

**int** prime[] = **new** **int**[number], count = 0, num = 2;

**for**(count=0;count<number;) {

**boolean** flag = **false**;

**for**(**int** j=2; j<=num/2; j++) {

**if**(num%j==0) {

flag=**true**;

**break**;

}

}

**if**(!flag) {

prime[count] = num;

count += 1;

}

num += 1;

}

**return** prime;

}

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

Scanner scan = **new** Scanner(System.***in***);

**try** {

System.***out***.print("Enter the number upto which primes to be generated: ");

**int** N = scan.nextInt();

**if**(N <= 0)

System.***out***.println("Number must be greater than zero.");

**else** {

PrimesUptoN prime = **new** PrimesUptoN();

**int** primes[] = prime.primes(N);

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

**for**(**int** i:primes) System.***out***.print(i+ " ");

}

}

**catch**(Exception e) {

System.***out***.println("Input mismatch exception: "+e);

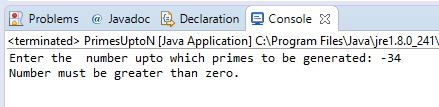
}

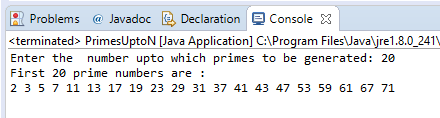
scan.close();

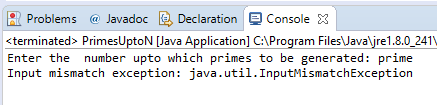
}

}

Output:







1. Java program to print even numbers in between given two numbers.

**package** com.rsc.babystepprogram;

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

**public** **class** EvenNumbersInRange {

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

Scanner scan = **new** Scanner(System.***in***);

**try** {

System.***out***.print("Enter the first number: ");

**int** start = scan.nextInt();

System.***out***.print("Enter the last number: ");

**int** end = scan.nextInt();

**if**(start <= end ) {

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

**for**(**int** i = start+2; i < end; i += 2)

System.***out***.print(i + " ");

**else**

**for**(**int** i = start+1; i < end; i += 2)

System.***out***.print(i + " ");

}

**else**

System.***out***.println("First number must be less than the last number");

}

**catch**(Exception e) {

System.***out***.println("Input mismatch exception: "+e);

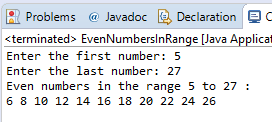
}

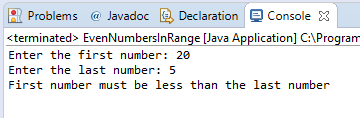
scan.close();

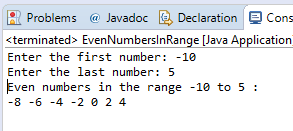
}

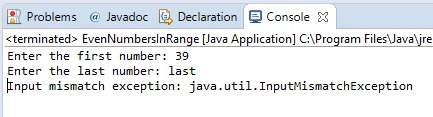
}

Output :









1. What is abstraction?
2. **Abstraction** is a process in which the internal implementation is hidden from the user and only the necessary details are shown. In java, abstraction is achieved with the help of
3. **Abstract classes.**
4. **Interfaces.**
5. What is Encapsulation?
6. The data members and member functions are together wrapped into a single unit called a class. This process of encapsulating the code and data is called **Encapsulation.** These members can only be accessed inside the class.
7. What is JDK?
8. JDK is a software development kit which contains all the necessary tools that are required for the complete development of java applications and programs.

It consists of development tools like java compiler and interpreter along with JRE .

1. What is JVM?
2. JVM is a virtual machine which provides the run time environment for the java applications where the compiled java code i.e., java byte code is executed. JVM is platform dependant because each operating system has its own configuration.
3. Define inheritance?
4. **Inheritance** is a concept of Object Oriented Programming, in which a child object or a class will acquire all the attributes and behaviors of its parent object or a class. The child class is will contains some advanced features compared to its parent class. **Extends** keyword is used to inherit a parent class.

**Eg. class ChildClassName extends ParentClassName {}**

1. How java achieved platform independence?
2. The java code once compiled produces a file with .class extension which contains the byte code. This byte code is executed in JVM. Irrespective of the platform dependence of JVM, the byte code can run in any machine, whatever the configuration of OS. Thus java achieved the platform independence.
3. Write the syntax of main function?
4. The syntax of java main function looks like :

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

1. What is conditional operator?
2. In java, ternary operator is called conditional operator. This operator contains three arguments, hence the name ternary.

Eg. result = (expression) ? ( statement-1 : statement-2 );

Here, the expression is resolved to check for conditionality. If true, then the result of statement-1 will be assigned to the variable, else statement-2.

It is the simplified version of if-else statement.

1. How many data types are in java?
2. There are two types of classification in datatypes.
   1. Primitive

Primitive consists of 8 data types:

Byte, short, int, long, char, boolean, float and double.

* 1. Non primitive contains the user defined data types like string, class, object, array, list, hash set etc.

1. What is constant. How it is declared?
2. A constant is a variable whose value once assigned cannot be changed throughout the program. The java **final** keyword is used to define a variable as a constant.

Eg. final float pi = 3.1415f;