Assignment-1

**1.Write a java program to find the area of rectangle?**

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

**public** **class** Rectangle {

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

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

**float** width=sc.nextFloat();

**float** height=sc.nextFloat();

**float** area=width\*height;

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

}

}

Output:

10.0

20.0

Area of rectangle is:200.0

**2.Write a java program to check the given number is Armstrong or not?**

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

**public** **class** Armstrong {

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

**int** s=0,r,x;

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

**int** n=sc.nextInt();

x=n;

**while**(n>0) {

r=n%10;

s=s+r\*r\*r;

n=n/10;

}

**if**(x==s) {

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

}

**else** {

System.***out***.println("Given number is not a Armstrong number");

}

}

}

Output:

153

Given number is Armstrong number

257

Given number is not a Armstrong number

**3.Write a java program to check given number is palindrome or not?**

**import** java.util.Scanner;

**public** **class** Palindrome {

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

**int** s=0,r,x;

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

**int** n=sc.nextInt();

x=n;

**while**(n>0) {

r=n%10;

s=(s\*10)+r;

n=n/10;

}

**if**(x==s) {

System.***out***.println("Given number is Palindrome");

}

**else** {

System.***out***.println("Given number is not a Palindrome");

}

}

}

Output:

676

Given number is Palindrome

487

Given number is not a Palindrome

**4.Write a java program to generate first N prime numbers?**

**import** java.util.Scanner;

**public** **class** Prime {

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

**int** s=1,number=3;

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

**int** n=sc.nextInt();

**if** (n>=1) {

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

System.***out***.println(2);

}

**for**(**int** i=2;i<=n;i++) {

**for**(**int** j=2;j<=Math.*sqrt*(number);j++) {

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

s=0;

**break**;

}

}

**if**(s!=0) {

System.***out***.println(number);

}

s=1;

number++;

}

}

}

Output:

10

First 10 prime numbers are:

2

3

5

7

11

13

17

19

23

29

**5. Write a java program to print even numbers in between given two numbers?**

**import** java.util.Scanner;

**public** **class** Even {

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

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

**int** n1=sc.nextInt();

**int** n2=sc.nextInt();

System.***out***.println("Even number from "+n1+" to "+n2+" are:");

**for**(**int** i=n1;i<=n2;i++) {

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

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

}

}

}

}

Output:

10

20

Even number from 10 to 20 are:

10

12

14

16

18

20

**1. What is Abstraction?**

**Abstraction** is a process of hiding the implementation details and showing only functionality to the user.

Another way, **Data Abstraction** may also be defined as the process of identifying only the required characteristics of an object ignoring the irrelevant details .The properties and behaviors of an object differentiate it from other objects of similar type and also help in classifying/grouping the objects.

**2. What is Encapsulation?**

**Encapsulation** is one of the four fundamental OOP concepts. The other three are inheritance, polymorphism, and abstraction.

Encapsulation in Java is a mechanism of wrapping the data (variables) and code acting on the data (methods) together as a single unit. In encapsulation, the variables of a class will be hidden from other classes, and can be accessed only through the methods of their current class. Therefore, it is also known as **data hiding**.

To achieve encapsulation in Java −

* Declare the variables of a class as private.
* Provide public setter and getter methods to modify and view the variables values.

**3. What is JDK?**

The **Java Development Kit (JDK)** is a software development environment that offers a collection of tools and libraries necessary for developing Java applications. You need the JDK to convert your source code into a format that the Java Runtime Environment (JRE) can execute. It includes the Java Runtime Environment (JRE), an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (javadoc) and other tools needed in Java development.

**4. What is JVM?**

* JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java bytecode can be executed.
* JVM, i.e., Java Virtual Machine
* JVM is the engine that drives the Java code
* Mostly in other Programming Languages, compiler produce code for a particular system but Java compiler produce Bytecode for a Java Virtual Machine.
* When we compile a Java program, then bytecode is generated. Bytecode is the source code that can be used to run on any platform
* Bytecode is an intermediary language between Java source and the host system.
* It is the medium which compiles Java code to bytecode which gets interpreted on a different machine and hence it makes it Platform/Operating system independent

**5. Define Inheritance?**

**Inheritance in Java** is a mechanism in which one object acquires all the properties and behaviors of a parent object. It is an important part of OOPs (Object Oriented programming system).

The idea behind inheritance in Java is that you can create new classes that are built upon existing classes. When you inherit from an existing class, you can reuse methods and fields of the parent class. Moreover, you can add new methods and fields in your current class also.

Inheritance represents the **IS-A relationship** which is also known as a parent-child relationship.

**6. How java achieved platform independence?**

 Java is a platform independent programming language, Because when you install jdk software on your system then automatically JVM are installed on your system. For every operating system separate JVM is available which is capable to read the **.class** file or **byte code**. When we compile your Java code then .class file is generated by javac compiler these codes are readable by JVM and every operating system have its own JVM so JVM is platform dependent but due to JVM java language is become platform independent.

**7. Write the syntax of main function?**

Java main method is the entry point of any java program. Its syntax is always public static void main(String[] args)

Syntax:

Main.java:

public class Main {

     public static void main (String[] args)

     {

         System.out.println("Hello World!!");

     }

}

**8. What is conditional operator?**

The conditional operator is a ternary operator (it has three operands) and is used to evaluate boolean expressions, much like an if statement except instead of executing a block of code if the test is true, a conditional operator will assign a value to a variable. A conditional operator starts with a boolean operation, followed by two possible values for the variable to the left of the assignment (=) operator. The first value (the one to the left of the colon) is assigned if the conditional (boolean) test is true, and the second value is assigned if the conditional test is false.

The operator is written as:

Variable x = (expression)? value if true: value if false

**9. How many datatypes in java?**

Data types are divided into two groups:

* **Primitive data types** - includes byte, short, int, long, float, double, boolean and char
* **Non-primitive data types** - such as String, Arrays and Classes

**10. What is constant? How it is declared?**

A constant holds a value that does not change. A constant declaration specifies the name, data type, and value of the constant and allocates storage for it. The declaration can also impose the NOT NULL constraint.

Declaration:

* To turn an ordinary variable into a constant, you have to use the keyword "final."
* As a rule, we write constants in capital letters to differentiate them from ordinary variables.
* If you try to change the constant in the program, javac (the Java Compiler) sends an error message. This happens because you can only assign a value to a constant once.

.