DAY 1

// Java is considered as a Pure OOP language.

/\*

int x = 15; //Primitive data type, not an object

Integer x = new Integer(15); //Wrapper class object

\*/

/\*

JVM, JRE, JDK

JVM: Java Virtual Machine

JRE: Java Runtime Environment

JDK: Java Development Kit

Task of JVM:

1. Load the byte code

2. Verifies the code

3. Provide JRE for execution

4. Execution of code

Java is a portable language or Java is platform independent.

[Write Once, Run Anywhere(WORA)]

JVM is platform dependent, as it interacts with the OS.

C, C++, Python, Java

Speed of execution: C, C++, Java, Python

prog.c

#include <stdio.h>

int main()

{

printf("Hello");

return 0;

}

prog.cpp

#include <stdio.h>

int main()

{

printf("Hello");

return 0;

}

This will also excute in C++, as C++ is a superset of C.

\*/

/\*

Class & object

Class: Collection of similar objects

Object: Instance of a class

10 Audi cars at one place. Can we call it a class? Ans: NO

Class: Its a blueprint, set of features/specifications. It does not exist physically.

Object: Where all features are implemented. It exists physically.

Class contains: data members, methods, constructors, blocks(static & non-static block)

\*/

/\*

class Test{

int x; //Data member

//method

void show()

{

System.out.println("Data: " + x);

}

public static void main(String[] args) {

Test ob = new Test(); //Object Creation

ob.x = 25;

ob.show();

}

}

\*/

// Create multiple objects of a class

/\*

class Test {

int a;

void show() {

System.out.println("Data: " + a);

}

public static void main(String[] args) {

Test ob1 = new Test();

Test ob2 = new Test();

ob1.a = 35;

ob2.a = 45;

ob1.show();

ob2.show();

}

}

\*/

/\*

Datatypes:-

Primitive: int, float, double, byte, short, long, char, boolean

Non-primitive: Array, String, Class etc.

Wrapper Classes: Integer, Float, Double, Byte, Short, Long, Character, Boolean

Autoboxing & Auto- Unboxing:

int x = 20

Integer i = x; //Integer i = new Integer(x); <- Autoboxing

Integer y = new Integer(35);

int a = y; //Auto-unboxing

\*/

/\*

Static Variable: static int x; //x is static variable

Non-static: int y; // y is a non-static variable

\*/

//OOP Methodology

/\*

class Test {

static int x; //Static variable

int y; //Non-static variable

void show() {

System.out.println("Data: " + x + " " + y);

}

}

// Driver class

class TestDrive {

public static void main(String[] args) {

Test ob1 = new Test(); // Composition ("Has A" - relationship)

Test ob2 = new Test();

ob1.x = 15;

ob1.y = 25;

ob2.x = 30;

ob2.y = 50;

ob1.show();

ob2.show();

}

}

\*/

import java.util.Scanner;

class Student {

String name;

int regno;

int sem;

void input() {

Scanner sc = new Scanner(System.in);

System.out.print("Enter name, reg no & semester in order: ");

name = sc.nextLine();

regno = sc.nextInt();

sem = sc.nextInt();

}

void show() {

System.out.println("Details:");

System.out.println("Name:" + name);

System.out.println("Reg. No:" + regno);

System.out.println("Semester:" + sem);

}

}

class Database {

public static void main(String[] args) {

Student s1 = new Student();

Student s2 = new Student();

s1.input();

s2.input();

s1.show();

s2.show();

}

}