# G.P.E.S , MANESAR

## Lesson Plan (OOPS Using JAVA)

Name of the Faculty: Virender Singh Rawat

Discipline: Computer Engg.

Semester: 4th

Subject: OOPS Using JAVA

Lesson Plan Duration: (From 15 Feb. 2024 to 23June 2023) Work Load (Lecture/Practical) per week (In hour): Lecture-02, Practical-04

|  |  |  |  |
| --- | --- | --- | --- |
| **WEEK** | **THEORY** | | **Practical** |
| **1st** | **LECTURE**  **DAY** | **TOPIC** | **TOPIC** |
| **1** | **UNIT 1 INTRODUCTION AND FEATURES**  Fundamentals of object oriented programming | Write a program in JAVA to print “Hello” using classes.  . |
| **2** | Procedure oriented programming V/S object oriented programming (OOP) |
| **3** | Object oriented programming concepts–  Classes ,object ,object reference |
| **2nd** | **1** | Abstraction, encapsulation |
| **2** | Inheritance, polymorphism |
| **3** | Introduction of eclipse(IDE)for developing programs in Java |
| **3rd** | **1** | **UNIT 2 LANGUAGE CONSTRUCTS**  Review of constructs of C used in JAVA: variables | Write a program to input using Scanner Class. |
| **2** | Types and type declarations |
| **3** | Data types |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **4th** | **1** | Increment operators | Write a program to print factorial of a Number. |
| **2** | Decrement operators |
| **3** | Relational and logical operators |
| **5th** | **1** | If then else clause; conditional expressions | Write a program to create a Class and make objects of that class. |
| **2** | Input using scanner class and output  statement |
| **3** | Loops, switch case, arrays, methods |
| **6th** | **1** | **UNIT 3 CLASSES AND OBJECTS**  Creation | Create a class with data members Feet, Inches and add them. |
| **2** | Accessing class members |
| **3** | Private Vs Public Vs Protected Vs Default |

|  |  |  |  |
| --- | --- | --- | --- |
| **7th** | **1** | Constructors | Create a class using constructors. |
| **2** | Object |
| **3** | Object Reference |
| **8th** | **1** | **UNIT 4 INHERITANCE**  Definition of inheritance | Create a class and show the use of Single inheritance. |
| **2** | Protected data |
| **3** | Public data, Constructor chaining |
| **9th** | **1** | Order of invocation |
| **2** | Types of inheritance |
| **3** | Single inheritance |
| **10th** | **1** | Multilevel inheritance, | Create a class and show the use of multiple inheritances. |
| **2** | Hierarchical inheritance |
| **3** | Hybrid inheritance |
| **11th** | **1** | **UNIT 5 POLYMORPHISM**  Method overloading |
| **2** | Constructor overloading |
| **3** | Method overriding |
| **12th** | **1** | Up-casting | Create a class and show the use of Multi-level inheritance. |
| **2** | Down-casting |
| **3** | **UNIT 6 ABSTRACT CLASS & INTERFACE**  Key points of Abstract class |
| **13th** | **1** | Interface | Create a class showing the use of Constructor Overloading. |
| **2** | Difference between an abstract class &  interface |
| **3** | Implementation of multiple inheritance through interface |

|  |  |  |  |
| --- | --- | --- | --- |
| **14th** | **1** | **UNIT 7 EXCEPTION HANDLING**  Definition of exception handling | Create a program showing the use of Interfaces. |
| **2** | Implementation of keywords like try |
| **3** | Catch, finally |
| **15th** | **1** | Throw &Throws | Create a program using Try and Catch Block. |
| **2** | Importance of exception handling in  Practical implementation of live projects |
| **3** | **REVISION** |
| **16th** | **1** | **TEST** | Revision |
| **2** | **REVISION** |
| **3** | **REVISION** |