CE219.01: OBJECT ORIENTED PROGRAMMING USING JAVA

Credit and Hours:

Teaching Scheme	Theory	Practical	Total	Credit
Hours/week	3	4	7	5
Marks	100	100	200	

A. Objective of the Course:

This subject introduces OOP using Java as the implementation language. It emphasizes proper formulation and abstraction of the problem domain in the programming process in order to build programs that are robust, secure, and portable.

The objective of course is,

- To familiar with the concepts of Object oriented Programming using Java Language.
- To familiar with various Real word Java Applications.
- To apply the concepts of Java theoretically and practically as well.

B. Outline of the Course:

Sr No.	Title of the unit	Minimum number of hours
1.	Fundamental of Programming in Java	01
2.	Class Fundamentals	01
3.	Array & String Handling	02
4.	Inheritance, Interfaces & Packages	02
5.	Exceptions Handling	02
6.	Multithreaded Programming	04
7.	File I/O and NIO	03
8.	Java Collection Frameworks and Generics	03
9	Java 8 Programming Features.	02
10	Introduction of OO concepts	03
11	Modeling Concepts	03
12	Class Modeling	04
13	Interaction Modeling, State Modeling	05
14	Analysis and Design, System Design, Class design	10

Total hours (Theory): 45

Total hours (Lab): 60

Total hours: 120

C. Detailed Syllabus:

1.	**Fundamental of Programming in Java**	**01 Hour**	**03 %**
1.1	History of Java		
1.2	Basic overview of java, Bytecode, JVM		
1.3	Buzz-words		
1.4	Application and applets		
1.5	Constants, Variables & Data Types, Comment		
1.6	Operators, Control Flow		
2.	**Class Fundamentals**	**01 Hour**	**03 %**
2.1	General form of class, Creating class Overloading methods		
2.2	Constructor ,Declaring Object, Returning objects, using objects as parameters,		
2.3	Assigning object reference variables		
2.4	Introducing Access control , Understanding static		
2.5	Introducing final, The finalize ( ) method		
2.6	The this keyword ,Garbage collection		
3.	**Array & String Handling**	**02 Hours**	**05 %**
3.1	Array basics, String Array, String class		
3.2	StringBuffer and StringBuilder class		
3.3	String Tokenizer Class and Object Class		
4.	**Inheritance, Interfaces & Packages**	**02 Hours**	**06 %**
4.1	Inheritance: Using super creating multilevel Hierarchy		
4.2	method overriding, Dynamic method dispatch		
4.3	abstract classes, Using final with Inheritance		
4.4	Using Package: Defining package		
4.5	Finding package and CLASSPATH		
4.6	Access protection, Importing package		
4.7	Interface: Defining Interface,		
4.8	Implementing Interface, Variables in Interface		
5.	**Exceptions Handling**	**02 Hours**	**04 %**

5.1	Exception types		
5.2	Try ...Catch...Finally, Throw, Throws		
5.3	creating your own exception subclasses		
6.	**Multithreaded Programming**	**04 Hours**	**10 %**
6.1	Life cycle of thread		
6.2	thread methods, thread priority		
6.3	thread exceptions		
6.4	Implementing Runnable interface		
6.5	Synchronization and Concurrency		
7.	**File NIO**	**03 Hours**	**06 %**
7.1	File and Directories		
7.2	Byte streams and character streams, Random Access Files		
7.3	NIO: Meta Data File Attributes Buffers, Channels, Recursive Operation.		
8.	**Collection Framework and Generics**	**03 Hours**	**06 %**
8.1	Collections of objects		
8.2	Collections:Sets, Sequence, Map		
8.3	Understanding Hashing		
8.4	Use of Array List & Vector		
8.5	Generics Class		
9	**Java 8 Programming Features.**	**02 Hours**	**04 %**
9.1	Lamda Expression, Default methods		
9.2	Processing data with streams		
10.	**Introduction of OO concepts**	**03 Hours**	**06%**
10.1	About Object Orientated Technology		
10.2	Development and OO Modeling History		
11.	**Modeling Concepts**	**03 hours**	**06%**
11.1	Modeling design Technique		
11.2	Three models, Class Model, State model and Interaction model		
12.	**Class Modeling**	**04 Hours**	**08%**
12.1	Object and class concepts		
12.2	link and association		
12.3	Generalization and Inheritance		
12.4	Advanced class modeling- aggregation, Abstract class ,metadata, constraints		
13.	**Interaction Modeling, State Modeling**	**05 Hours**	**11%**

13.1	Use case Models, sequence models, activity models,		
13.2	Event, state, Transition and conditions, state diagram, state diagram behaviour		
13.3	Concurrency, Relation of Class and State models.		
14.	**Analysis and Design, System Design, Class design**	**10 Hours**	**22%**
14.1	Development Life cycle, Development stages		
14.2	Domain Analysis-Domain class model, domain state model, domain interaction model		
14.3	breaking system into subsystems ,identifying concurrency, allocation of subsystems		
14.4	management of data storage, choosing a software control strategy		

D. Instructional Method and Pedagogy:

- At the start of course, the course delivery pattern, prerequisite of the subject will be discussed.
- Lectures will be conducted with the aid of multi-media projector, black board, OHP etc.
- Attendance is compulsory in lectures and laboratory which carries a 5% component of the overall evaluation.
- Faculty deals with concept test as it implies focus on one key concept of learning
- Minimum two internal exams will be conducted and average of two will be considered as a part of 15% overall evaluation.
- Assignments based on course content will be given to the students at the end of each unit/topic and will be evaluated at regular interval. It carries a weightage of 5%.
- Surprise tests/Quizzes will be conducted which carries 5% component of the overall evaluation. The course includes a laboratory, where students have an opportunity to build an appreciation for the concepts being taught in lectures.
- Minimum 10 experiments are suggested in the laboratory related to course content.

E. Student Learning Outcome:

By taking this course Programming in Java,

- **Students will be able to use different commands of JDK.**
- **Student will able implement GUI as well as multithreaded programming for real life projects.**

Employability

- **Students are able to design and develop projects in higher semesters using Object orienteddesign approach and java programming language.**

Skill development

F. Recommended Study Material:

❖ Text Books:

1. The Complete Reference Javatm2 ,HerbertSchildt ,TMH publication
2. Teach yourself Java - by Joseph O'neil, TMH publication
3. Oriented Modeling and Design with UML, MichaelBlaha, James Ram Baugh, Prentice Hall,and 2nd or 3rd Edition.
4. Java Generics and Collections by Maurice Naftalin, Philip Wadler Publisher: O'Reilly Media

❖ Reference Books:

1. Thinking in Java, Bruce Eckel, Prentice Hall
2. Java 1.1 Unleashed, Michael Morrison, Sams Publication

❖ Web Materials:

1. www.java.sun.com
2. www.javaarchives.com
3. www.freewarejava.com
4. www.codeguru.com

http://www.tutorialspoint.com/java/java_collections.htm