BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, Pilani
Pilani Campus
AUGS/ AGSR Division

SECOND SEMESTER 2022-23
COURSE HANDOUT

Date: 16.01.2023

In addition to part I (General Handout for all courses appended to the Time table), this portion gives further specific details regarding the course.

Course No	**:**	**CS F213**
Course Title	**:**	**Object Oriented Programming**
Instructor-in-Charge	**:**	**Dr. Avinash Gautam (avinash@ pilani.bits-pilani.ac.in)**
Instructor(s)	**:**	**Dr. L. Rajya Lakshmi (rajya.lakshmi@pilani.bits-pilani.ac.in)**
Lab Instructors	**:**	**Mr. Ashish Verma (p20200020@pilani.bits-pilani.ac.in)**
		Ms. Shail Saharan (p20170404@pilani.bits-pilani.ac.in)
		Ms. Devika S (p20210024@pilani.bits-pilani.ac.in)

1. Course Description:

This course gives students an in-depth understanding of object-oriented programming, object-oriented analysis and design, and design patterns. Java programming language is chosen as a vehicle to teach the concept of object orientation. The course is taught with live demonstrations, running and debugging several examples on tools like Eclipse. The later part of the course focuses on the analysis and design of object-oriented software systems. After completing this course a student should be able to effectively realize and implement real-world problems using object-oriented principles and techniques.

2. Scope and Objective of the Course:

a) Gives an in-depth understanding of object-oriented programming using the java programming language, object-oriented analysis and design, and design patterns.
b) In the classroom, the course will be taught with live demonstrations, running and debugging several examples on tools like Eclipse.
c) The latter part of the course focuses on designing object-oriented software.

3. Text Books:

T1. Java: The Complete Reference, Herbert Schildt, McGraw Hill Education, Tenth Edition, 2017
T2. Object Oriented Design & Patterns, Cay Horstmann, John Wiley & Sons, 2004

4. Reference Books:

R1. Java™ Design Patterns – A Tutorial, James W. Cooper, Addison-Wesley, 2000

4. Course Plan

Module Number	Lecture session	Reference	Learning Outcome
1. Object-Oriented and Java Basics (**05 Lectures**)	L1.1. Object-Oriented Basics	T1, Ch. 2	- Object and Class Basics - Basic Pillars of Object-Oriented Programming o Abstraction o Encapsulation o Inheritance o Polymorphism
	L1.2. Java Programming Syntax	T1, Ch. 2-5	- Java Program Structure - Compiling and Executing a Simple

			Java Application - Types of Variables in Java - Primitive Types in Java - Type Promotion and Type Casting Rules - Operators - Control statements
	L1.3 – 1.4 Defining Classes and Object Creation	T1, Ch. 6	- Defining Classes and Access Modifiers - Creating Objects - Role of Constructors - Accessing Instance Fields and Methods - Local Variables vs. Instance Variables - Mutable and Immutable Objects - Command-Line Arguments (**lab**) - Reading Input from the console using Scanner class (**lab**) - UML representation of a Class
	L1.5 Use of static final keywords in Java, Method Overloading, Objects as Parameters	T1, Ch. 7	- Use of static and final keywords (**lab**) - Method Overloading - Constructor Overloading - Objects as Parameters to Methods
2. Arrays and String in Java (**02 Lectures**)	L2.1 Arrays in Java	T1, Ch.3, Ch. 19	- Implementing 1-D and 2-D Arrays - Role of Arrays class
	L2.2. Strings in Java	T1, Ch. 17	- String class and methods - StringBuffer and StringTokennizer
3. Polymorphism and Inheritance in Java (**07 Lectures**)	L3.1. Inheritance in Java	T1, Ch. 8	- Extending classes and role of super keyword - Method Overriding [Super Type vs Sub-Type Relationships] - UML representation of Inheritance relationship
	L3.2. – 3.7 Abstract Classes, Abstract Methods and Interfaces	T1, Ch. 8, Class notes	- Abstract methods and classes - Interfaces in Java [Class vs Interface] - Nested and Inner Classes - Anonymous class and objects - Lamda expressions - UML representation of Abstract classes and methods
4. Collections Framework of Java (**06 Lectures**)	L4.1 – 4.3 Collections in Java	T1, Ch. 19, Class notes	- Introduction to Collection Framework - Important Collection Interfaces and their methods - ArrayList and LinkedList classes - Iterators and ListIterators - Wrapper classes and Autoboxing - Sets and Hash-Maps in Java - Comparable and Comparator Interfaces - UML representation of Collection Classes
	L4.4 – 4.6 Generic Programming	T1, Ch. 14 , Class Notes	- Generic Form of a class - Generic Interfaces and Bounded Types

5. Exception Handling Mechanism (**02 Lectures**)	L5.1– 5.2 Exceptions in Java	T1, Ch. 10	- Exception basics and types - Catching Exceptions - Writing your own Exceptions
6. Input/ Output in Java (**01 Lectures**)	L6.1 File handling in Java	T1, Ch. 21 <<class notes>>	- Create a file in java - Read a file in java using BufferedInputStream - Read a file in using BufferedReader - Write to a file in using FileOutputStream - Write to file in using BufferedWriter - Append to a file in java using BufferedWriter, PrintWriter, FileWriter - Delete file in using delete() method. - Rename file in Java using renameTo() method
7. Multithreading (**05 Lectures**)	L7.1 – 7.5 Multithreaded Programming in Java	T1, Ch. 11	- Multithreading vs. Multitasking - Thread Class and methods - Creating your own Threads and Runnable Interface - Thread Synchronization - Inter Thread Communication
8. Object Model (**01 Lectures**)	L8.1 Java Object Model	T2, Ch. 7	- The Java Type System - Type Inquiry - Object Class - Shallow and Deep Copy
9. Object-Oriented Analysis and Design (**03 Lectures**)	L9.1 – 9.3 Object-Oriented Analysis and Design	T2, Ch. 2 and class notes	- Understanding Class Relationships, Multiplicities (Cardinality) - Drawing Class Diagram [Dependency Diagram, and Relationship diagrams] - State Diagrams
10. Software Design Patterns (**08 Lectures**)	L10.1 – 10.8 Object-Oriented Design Patterns	Class notes Derived from reference book R1 and other online resources	**(a) Creational Patterns [01 Lecture]** - Singleton - Builder - Abstract Factory and Factory Method **(b) Structural Patterns [03 Lectures]** - Adapter - Composite - Decorator - Bridge **(c) Behavioral Patterns [04 Lectures]** - Iterator - State - Strategy - Observer - Command - Chain of Responsibility

5. Lab Plan

Lab #	Topics to be Covered
1	Introduction to Eclipse IDE
2	Reading user input, Designing Simple Classes
3	Unit testing using JUnit Framework
4	Static variables, methods, and blocks, Object as Parameter, Wrapper Classes
5	Inheritance, Polymorphism, Abstract Classes
6	Arrays, Passing arrays, Multi-dimensional arrays, Strings, StringBuffer, StringTokenizer
7	Interfaces [Comparable, Comparator], Inner classes and static inner classes, Lamda expressions
8	Anonymous inner classes, Collections, Generics
9	Exception Handling and Text File I/O
10	Multi-threaded Programming in Java
11	Design Patterns (creational, structural, and behavioral patterns)

6. Evaluation Scheme:

Component	Duration	Weightage (%)	Date & Time	Nature of component (Close Book/ Open Book)
Quiz-1	30 Min	05%	25/02/2023	Closed Book
Quiz-2	30 Min	05%	15/04/2023	Closed Book
Quiz-3	30 Min	05%	29/04/2023	Closed Book
Computer Based Test (CBT)	90 Min	20%	16/04/2023	Open Book
Mid-Semester Exam	90 Min	30%	18/03/2023	Closed Book
Comprehensive Exam	180 Min	40%	20/05/2023	Partly Open

7. Important Course Policies

a) **Labs falling on holidays will not be canceled or rescheduled. They will be conducted on the very same date and time.**

b) **No makeup request will be entertained if the student has not attended at-least seven lab sessions out of the remaining eight labs.**

8. Chamber Consultation Hour : All the instructors can be contacted through email.

9. Notices : All notices will be posted on Nalanda.

10. Make-up Policy : Make-up will be granted only in case of hospitalization. The makeup request should reach the IC 24 hours before the exam.

The best of the two quizzes will be considered for the final grading. There is one buffer quiz, and no makeup will be granted for the quizzes.

Dr. Avinash Gautam
Instructor-in-charge
Course No. CS F213