1. **JAVA Course Outline**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | | **Topic** | **Objective** | **Date** | **H** | **Q** | **P** |
|  | 01 | **What is OOP?**   * Object * Class * Inheritance * Polymorphism * Encapsulation | **Understand:**   * Object & Class * OOP Concept & Features | 15  MAR | ☑ |  | ☑ |
|  | 02 | **Java Fundamental**   * Characteristics * JDK/JRE * J2SE/J2EE/J2ME * Java API Document * JVM & Java Memory * Java Running Process * Installing Eclipse * First Java Program * Variable declaration * Java Naming Rule * Comment | **Understand:**   * Java Technology Features * Java Virtual Machines (JVM) * How JVM manage the memory * Configure JDK and IDE * Practice first Java program * Data Types * How to declare a variable? | 20  MAR |  |  | ☑ |
|  | 03 | **Java Syntax and Data Type**   * Primitive Data Type/Wrapper Class * Auto Boxing * Promotion and Casting * Operators * System Class input / output * Scanner Class * BufferedReader / InputStreamReader | **Understand:**   * The difference between Primitive Data Type and Type Wrapper Class * Promotion and Casting * Operators * Standard I/O | 21  MAR |  |  | ☑ |
|  | 04 | **Control Flow Statement**   * If-then/if-then-else Statement * Switch-case Statement * For loop * Do-while loop * Break, continue & return keyword   **Array**   * Concept of array * One-Dimensional array * Many-Dimensional array | **Understand:**   * How to use Control Flow Statement * How to use break ,continue and return Keyword * What & how to use Array * How to access many-dimensional array | 22  MAR | ☑ |  | ☑ |
|  | N/A | **Assignment 1**   * Hotel Management | **Practice with:**   * Standard I/O * Many-dimensional array * Control flow statements | 23  MAR  30  MAR |  |  | ☑ |
|  | 05 | **Java Method**   * Structure * Return type * Access modifier * Parameter & Varargs   **Class & Object**   * Class Structure * Constructor * this & this() * Object creation | **Understand:**   * Structure of method * Return type of method * Access modifier * Parameter * Variable Argument * Difference between this and this() * How constructor work? | 27  MAR |  | ☑ | ☑ |
|  | 06 | **Nested Class**   * Static & Final Keyword * Static inner class * Local inner class * Anonymous inner class | **Understand:**   * What & How to static & final * What’s Nested class? * How many types of Nested class * How to use each type of Nested class | 29 MAR |  |  | ☑ |
|  | 07 | **Inheritance & Polymorphism**   * Inheritance (is-a relationship) * Composite (has-a relationship) * this & super Keyword * Type of Inheritance * Overriding Method * Polymorphism Implementation * Hidden keyword   **Abstract Class & Interface**   * Abstract Class * Interface | **Understand:**   * What’re inheritance and Polymorphism * What is-a relationship is. What has-a relationship is * how many type of inheritance in java? * What overriding method is * How to use this & super Keyword * What the hidden keywords in java * What abstract class and interface are * Difference between abstract class and interface * When to use abstract class. When to use interface | 03  APR |  | ☑ |  |
|  | 08 | **Collections**   * List: Vector, ArrayList, LinkedList * Set: HashSet, TreeSet * Map: HashMap, TreeMap, HashTable | **Understand:**   * Understand the differences of those collections * Know when to use each collection | 05 APR |  | ☑ | ☑ |
|  | 09 | **Exception**   * Exception Hierarchy * Checked Exceptions * Unchecked Exception * User-Defined Exception * Throwing Exception * Try-catch statement * Try-finally statement * Try-catch-finally statement * Try-with-resource statement * Multi-catch clause | **Understand:**   * Exception Hierarchy * Checked Exceptions * Unchecked Exception * User-Defined Exception * Throwing Exception * Try-catch statement * Try-finally statement * Try-catch-finally statement * Try-with-resource statement Multi-catch clause | 10 APR |  | ☑ |  |
|  | 10 | **Thread**   * Program, process, thread concept * Two ways of implementation thread (Thread/Runnable) * Commonly used methods in Thread class (sleep, join, wait, notify, etc.) * Synchronization | **Understand:**   * Program, process, thread concept * How to implement thread (2 ways)? * What Synchronization is * How to use synchronization * Show some examples of using commonly used methods | 11 APR | ☑ |  | ☑ |
|  | 11 | **java.lang Package**   * Class, Math, Runtime, Process, String, StringBuilder, StringBuffered, SecurityManager   **java.util Package**   * Random, Currency, Stack, Properties, StringTokenizer, Timer, TimerTask | **Understand:**   * Work with math class * Work with string in Java field * Study about Java security * Access local calendar via * Currency class * Working with properties class * Understand about both importance package in Java | 18 APR |  |  | ☑ |
|  | 12 | **File I/O**   * Path Class * File Class * Read/Write File * FileInputStream/FileOutputStream * FileWriter/FileReader * InputStreamReader/OutputStreamWriter * InputStream/OutputStream * Serialization * Random Access | **Understand:**   * Working with path class to access window path * How to write contents to file * How to read content from file * Working with Random Access File * Read/Write content in binary | 26  APR |  |  | ☑ |
|  | N/A | **Mini-Project 1**   * Stock Management | **Practice:**   * Working with file in project * CRUD with file * Find a good way to use file * Compare the method of using file * Team cooperation | 26  APR  -  04  MAY |  |  | ☑ |
|  | 13 | **Database Concept**   * File vs. Database * Database Modeling (Conceptual/ Logic/Physical Modeling) * JDBC Driver setting * Basic procedure of JDBC * DAO & DTO * Connection | **Understand**:   * Understand Database Concept * Know the differences between File and Database * Get to know Database Modeling | 02 MAY | ☑ | ☑ |  |
|  | N/A | **Mini-Project 2:**   * Stock Management | **Understand:**   * How to connect to database * Practice with DAO & DTO * Move from File to Database * Team cooperation | 16  MAY |  |  | ☑ |
|  | 14 | **Java 8**   * Lambda expression * Method references * Functional interface * Default method * New Date Time API * Base64 | **Understand:**   * Learn new technologies of Java * Working with Lambda * How to encrypt content with Base64 | 03  MAY |  |  | ☑ |
|  | 15 | **MVC Model**   * View * Model * Controller | **Understand:**   * Using MVC to separate the task | 08 MAY |  |  |  |
|  | 16 | **Servlet**   * Introduction to servlet * Environment Setup * Servlet Architecture * Servlet Lifecycle * Servlet-Http Status Code * First Application with Servlet * **Servlet Request & Response** * RequestDispatcher * SendRedirect Method | **Understand:**   * Working with Servlet * Using Servlet API * Checking the process of Servlet * How to send request to other request * How to redirect to other request * Servlet-http status code | 09  MAY |  |  |  |
|  | 17 | **Spring Framework Overview**   * Introduction to Spring Framework * Spring Modules * Spring Projects * Spring Web Architecture   **Getting Started with Spring Boot**   * Spring Boot Features * First Application with Spring Boot * Basic Spring Annotation(@Controller, @RequestMapping) | **Understanding**   * What is Spring Framework? * What are the benefits of Spring Framework? * What are the Spring Projects and Modules? * Features of Spring Boot * Create Spring Web Project * Some Spring Annotations | 16  MAY | ☑ |  |  |
|  | 18 | **Spring Boot with Template Engine(Thymeleaf)**   * Thymeleaf | **Understanding**   * Introduction to thymeleaf * Introduction * Text, Variable * Available Object * Standard Expression Syntax * Conditional * Iteration * Fragment | 17 MAY  22  MAY  23  MAY  24  MAY | ☑ |  |  |
|  | 19 | **Spring Bean & Spring Container**   * Bean Lifecycle * Bean Scope   **Dependency Injection with Spring**   * Spring IoC Container * Dependency Injection * Setter and Constructor Injection * Inject value from property file | **Understanding**   * What is Bean & Container in Spring? * What is Inversion of Control? * What is Dependency Injection? * Injecting the dependency | 29  MAY    30 MAY |  |  |  |
|  | 20 | **Spring Expression Language(SpEL)**   * SpEL   **POJO Programming Model**   * What is POJO programming model? * Advantages of POJO programming model | **Understanding**   * What is Spring Expression Language? * Problems of the old EJB programming model that triggered the birth of POJO movement * advantages of POJO programming model | 31  MAY |  |  |  |
|  | 21 | **Building Web Application using Spring Boot and Thymeleaf**   * Application Layer Architecture * CRUD Operation with Spring and Thymeleaf * Spring Form Validation * Spring File Upload | **Understanding**   * Understanding Application layer architecture * CRUD operation with Spring & Thymeleaf * Spring form validation * Uploading file using MultipartFile | 05 JUN  06  JUN  07 JUN | ☑ |  |  |
|  | 22 | **Data Access with Spring**   * Simple JDBC * Mybatis | **Understanding**   * Connecting to database using JDBC * Data access with Mybatis | 12  JUN |  |  |  |
|  | 23 | **Spring Security**   * Spring Security Features * In-Memory Authentication * User Detail Service Authentication (Database) | **Understanding**   * Securing web resources * Authenticate user using In-Memory Authentication * Authenticate user using Database * Customize login page * Customize authentication success handler * Authentication entry point | 13 JUN  14 JUN | ☑ |  |  |
|  | 24 | **RESTful Web Service with Spring**   * First REST Web Service * Exception Handling Mechanism * JSON Partial Response * Securing RESTful Web Service using **Basic Authentication** or **JSON Web Token(JWT)** | **Understanding**   * What is RESTful? * RESTful architecture * Request & Response data as JSON format * Exception handling with REST * Securing REST api * Basic Authentication * JSON Web Token Authentication * Developer Document(Swagger) | 19  JUN  20  JUN  21  JUN | ☑ |  |  |
|  | 25 | **RestTemplate**   * Request data from Web Service | **Understanding**   * Request data from web service * Hide API Key by using RestTemplate | 26  JUN | ☑ |  |  |
|  | 26 | **iReport Getting Start**   * Environment setup * iReport Layout understanding * Filling * View and print * Exporting   **iReport Formatting**   * Section * Group * Fonts * Unicode support * Style   **iReport Data Management**   * Parameter * Data Source * Fields * Expression * Variable | **Understand:**   * Interface design * Layout design * Print report as pdf or excel * Parse parameter from Java to iReport * Add Data Source * Study about iReport expression * Support with Khmer font * Styling the report * Add sub query to report | 27  JUN  28  JUN |  |  |  |