Core Java Checklist

> Review one portions need to be completed

Week 1:

1. **What is Java?**
    - **History**
    - **Features**
2. **JDK, JRE and JVM**
3. **How to set path?**
4. **Hello Java Program**
    - ➢ Sample Hello world.
5. **Naming Convention**
6. **Compile & Runtime errors**
7. **Command line arguments**
    - ➢ Write a program to add two numbers. Accept numbers using command line arguments.
8. **Data types**
    - **Primitive**
    - **Non Primitive(Wrapper classes)**
9. **Variables & its types**
    - **Local**
    - **Instance**
    - **Static**

    ➢ Write a program to

    a) add two integer numbers

    b) add one float number and one integer number

    c) print your name and age

    d) calculate area of Triangle

    ➢ Write a program to swap two numbers with temporary variable. Print the numbers before and after swap.

- Write a program to swap two numbers without temporary variable. Print the numbers before and after swap.

10. Decision Making in Java

- **Simple if Statement**
- **If else Statement**
- **If else if ladder statement**
- **Switch statement**
  - Write a program to check the given number is positive.
  - Write a program to check whether the candidate is eligible for driving license.
  - Write a program to find the grade of a Student based on total marks

    Mark less than 40- Failed

    40 to 60–Grade D

    61 to 70-Grade C

    71 to 80-Grade B

    81 to 100-Grade A
  - Write a program to check whether the given character is Vowel/not (Use switch-case).

Week 2:

11. Looping Statements in Java

- **For**
- **While**
- **Do…while**
- **For-Each Loop**
  - Write a program to print Fibonacci series[0 1 1 2 3 5].
  - Write a program to find the reverse of number.

12. Branching Statements in Java

- **Break**
- **Continue**

13. Java Comments

14. Java Array

- **Single Dimensional**
- **Multi-Dimensional**
  - Write a program to sum up all the elements of an array.
  - Write a program to add two matrices.
  - Write a program to search array element with Linear Search.

15. Methods

- **Syntax of Methods**
- **Static**
- **Call static methods & variables**
- **Invoking Parameterized Methods & different Return types**
- **Method Overloading with static methods**

- Write a program to set your name and age by a static method and get them in an another static method.
- Write a program to find the factorial of a number(**static method**) Output – "Factorial of <given number> is <result>.
- Write a program to check whether the given number is Palindrome/Not by using **static methods**.
  - Method 1- to find the reverse(pass the number in argument)
  - Method 2-to check palindrome/not
- Write a program to check whether the candidate is eligible for Voting(Use **static method** and boolean return type).

- Write a program to check whether the customer have discount (get 20% discount if total amount is greater than 5000) or not and get the final amount in main method. (**static methods**)
  - Get prices of items using parameterized method
  - Method 1 - Calculate total amount
  - Method 2 - Check discount

Week 3:

16. **Instance methods**
    - **Object and Class**
    - **Constructor syntax**
    - **Default constructor**
    - **Parameterized constructor**
    - **Call instance methods & variables**

    Example for static & instance variable memory allocation using constructor

    - Write a program to find the grade of 2 students based on total marks(3 subjects)
        - Get the student‟s marks by constructor
        - Return total mark to in main method
        - Find the grade of each student.
    - Write a program to find the
        - Average of three integer numbers, three float numbers(should have same method name)
        - Area of figures(circle, rectangle, square) by using three methods(should have same method name)
17. **this keyword and its uses**
    - Program to find the reverse of a number
        - Two constructors, one for calculation reverse and other for display “Finding reverse…”
        - Argument variable and instance variable should be same Main method allows to invoke only one constructor
    - Program to find the factorial of a number
        - Two methods, one for find calculate and other for print result.
        - Pass the number as argument
        - Main method allows to invoke only one method

❖ **Eclipse installation**

18. **Package**
19. **String Class**

- **String class methods(length, equals, equalsIgnoreCase, contains, charAt, etc)**
- **StringBuffer class**
- **StringBuilder class**

➢ Program to check whether the given Strings are Palindrome or not.

- Java
- Malayalam

> Portions to be completed before review 2
>Notes Distribution

Week 4:

20. **Git Installation**
21. **Install Maven in Eclipse**
22. **Inheritance(IS-A)**

- **Types of Inheritance**

➢ Program to find the total salary by hand of an Employee

- Class 1
  Get basic pay, deduction and bonus from console.
- Class 2
  Calculate hra (5% of basic pay) and pf (20% of basic pay).
- Class 3
  Find the total salary (basicpay+hra-pf-deduction+bonus) and get the salary slip
- Salary slip should contains :- basic pay, deduction, hra, pf, bonus and total salary by hand.

23. **Access modifiers**
24. **Debugging**

25. **super keyword and it's uses**

➢ Program to check the addition result is divisible by 10

- Class 1
  Return addition result of two numbers
- Class 2

Check the addition result is divisible by 10(use **super** keyword)

26. this vs super

Week 5:

27. Aggregation(HAS-A)

- Write a program to get the details of a Student
  - Class 1 - Student
  - Get the student name and roll number
  - Class 2- Address
  - Get the address of student
  - Print Student name, roll number with address
  - There is no IS-A relationship

28. Polymorphism

- **Method Overloading**
  - **Method Overriding**
  - **Method Overloading vs Method Overriding**
    - Program to calculate discount If customer purchase clothes on Offseason, set discount 15% and on Onseason 40%
    - Should use two classes, Onseason and Offseason
    - Use two methods- discount(method name should be same)

29. Encapsulation

- Program to withdraw amount from an ATM
- Class 1- Bank One method to set pin from „User" class and validate Pin in another method

[Valid pins – 1001, 1234, 1212]

- Pin number should declared as private
- Class 2 – User Get the pin from User

30. final keyword

- **variable**
- **method**

- **class**

31. **Abstraction**
    - **Abstract class**
        - Let's first create the superclass Employee and define a method called calculateSalary() as an abstract method. The Contractor class inherits all properties from its parent Employee but have to provide its own implementation of calculateSalary() method and multiply the value of payment per hour with given working hours. The FullTimeEmployee also has its own implementation of calculateSalary()method. In this case we just multiply by constant 8 hours.

    - **Interface**
        - Write a program, where RBI will be an interface, have a method recurringDeposit which can accept the amount and duration. This must be implemented in class HDFC.
            - When a customer deposit amount in HDFC, they must be able to know how much amount they will get after depositing for n period of time.
            - Interest rate is defined in RBI interface.

32. **Abstract vs Interface**

Week 6:

33. **Exception Handling**
    - **Checked Exception**
    - **Unchecked Exception**
    - **Exception handling**
    - **Try, catch, finally**
    - **Throw &throws**
    - **Custom exception**

34. **Collections**
    - **Generic vs Non generic**
    - **List and ArrayList [add(), contains(), remove(), size() etc ..]**
      Two ways to retrieve data (Iterator & advanced for loop)

- Set and HashSet[methods]
  - Write a Java program to create a new array list, add some colors (string) and print out the collection.
  - Write a Java program to retrieve an element (at a specified index) from a given array list.
  - Write a Java program to iterate through all elements in a array list.
  - Write a Java program to remove the third element from a array list.
  - Write a Java program to search an element in a array list.

35. **Maven**
    - **Data Driven Framework**
    - **Download and add Apache POI**
    - **Apache POI – Excel read**

36. **Jenkins**
    - **Jenkins Implementation**

Final Review…