HCL Training
Automation Testing: Selenium with Java

(A) Assignments

Assignment 1: Submission date: 19/12/2025

1	A class sixth student required to solve basic mathematics problems. For this he/ she needs to perform operations such as addition, subtraction, multiplication, division, remainder, square, cube, and absolute. Write a program using methods to perform these basic operations.
2	Write a menu driven program in Java to perform insertion, deletion, linear search, binary search, to find maximum value, to count even/ odd and to perform insertion sort operation in one dimensional array.
3	Write a program to perform addition, subtraction, multiplication, transpose, to check square matrix, to check matrix is diagonal, and to check matrix is identity.
4	Differentiate String, StringBuffer and StringBuilder. Write a program to perform String related operations using length(), isEmpty(), chatAt(), toString(), equals(), compareTo(), contains(), indexOf(), lastIndexOf(), startsWith(), endsWith(), matches(), substring(), toLowerCase(), trim(), replace(), split(), join(), and valueOf().
5	Design a console-based Bank Management Program in Java that allows a user to create a bank account and perform basic banking operations such as deposit, withdrawal, balance enquiry, and account details display. The program should handle exceptions as required.

Assignment 2: Submission date: 22/12/2025

1	Design a Java program where a bank account class implements multiple interfaces to perform banking and customer-related operations. The program should handle exceptions as required.
2	Write a simple Java Program which demonstrate the use of REGEX in Java and provides Input Validation Using Regular Expressions. The program should have following menu 1. Validate Mobile Number 2. Validate Email ID 3. Validate Username 4. Validate Password 5. Exit. After validation a simple welcome message displayed and if validation failed then invalid input message should displayed. The program should handle exceptions as required.
3	Design a Java program to manage Student Records using different ArrayList, Vector, List collections. The program should Add student records, Display all students, Remove a student by roll number, Search a student by roll number. The program should handle exceptions, use interface and REGEX as required.
4	Design a Java program to manage Employee Records using Hashmap, Hashtable, and TreeMap. The program should Add employee records, Display all employees, Search employee by employee ID, Remove employee by employee ID, Demonstrate null key and null value support. The program should handle exceptions, use interface and REGEX as required.
5	Design a console-based University Student Management System using Java Collection Framework that manages students, courses, and marks using ArrayList, Vector, Stack, list, set & map, Hashmap, Hashtable, TreeMap. Add student records, Display all students, Remove student by ID, Search student by ID, Sort students by marks, Convert HashMap to TreeMap, Count students course-wise, Store unique course names, Prevent duplicate entries and Display all courses. The program should handle exceptions, use interface and REGEX as required.

(B) Mini Projects

Mini Project 1: Submission date: 24/12/25

Sr. No	Project
1.	Design a console-based Bank Management System in Java that allows a user to create a bank account and perform basic banking operations such as deposit, withdrawal, balance enquiry, and account details display. The data should be stored in file. Use the concept of class, object, constructor, hierarchical inheritance, encapsulation (access specifiers), polymorphism (method overloading, method overriding, run time polymorphism) and abstraction (abstract class, abstract method).

Mini Project2: Submission date: 30/12/2025

Sr. No	Project
1.	Employee Management System using Java Collections and File Handling. Uses Collections Framework for data handling Uses File Handling to store and retrieve employee data Demonstrates real-world CRUD (insert, update, delete and search) operations. Add login system (username/password) 1. Add Employee 2. Display All Employees 3. Search Employee by ID 4. Update Employee Salary 5. Delete Employee 6. Display Sorted Employees 7. Display Departments 8. Exit Validation Rules ⇨ Employee ID must be unique ⇨ Salary must be positive ⇨ Department cannot be empty File must update after every change Exception handling with custom messages