

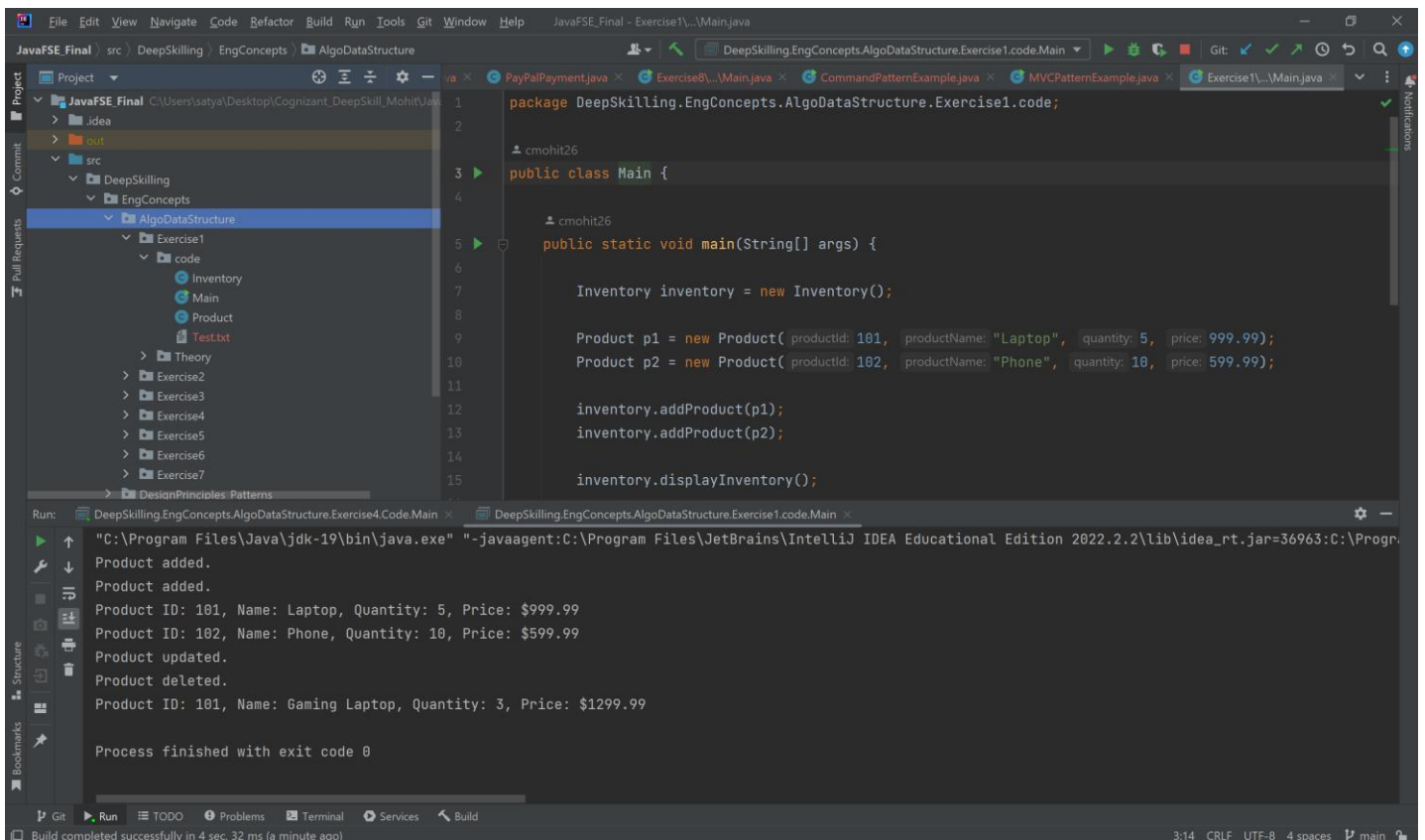
OUTPUTS [Week 1]

DesignPrinciplesAndPatterns

DeepSkilling -> EngConcepts -> AlgoDataStructure

Exercise 1 to 7

1. EXERCISE 1



The screenshot displays the IntelliJ IDEA IDE interface. The left sidebar shows the project structure with the following hierarchy:

- JavaFSE_Final
 - src
 - DeepSkilling
 - EngConcepts
 - AlgoDataStructure
 - Exercise1
 - code
 - Inventory
 - Main
 - Product
 - Test.txt

The main editor window shows the code for `Exercise1.code.Main`:

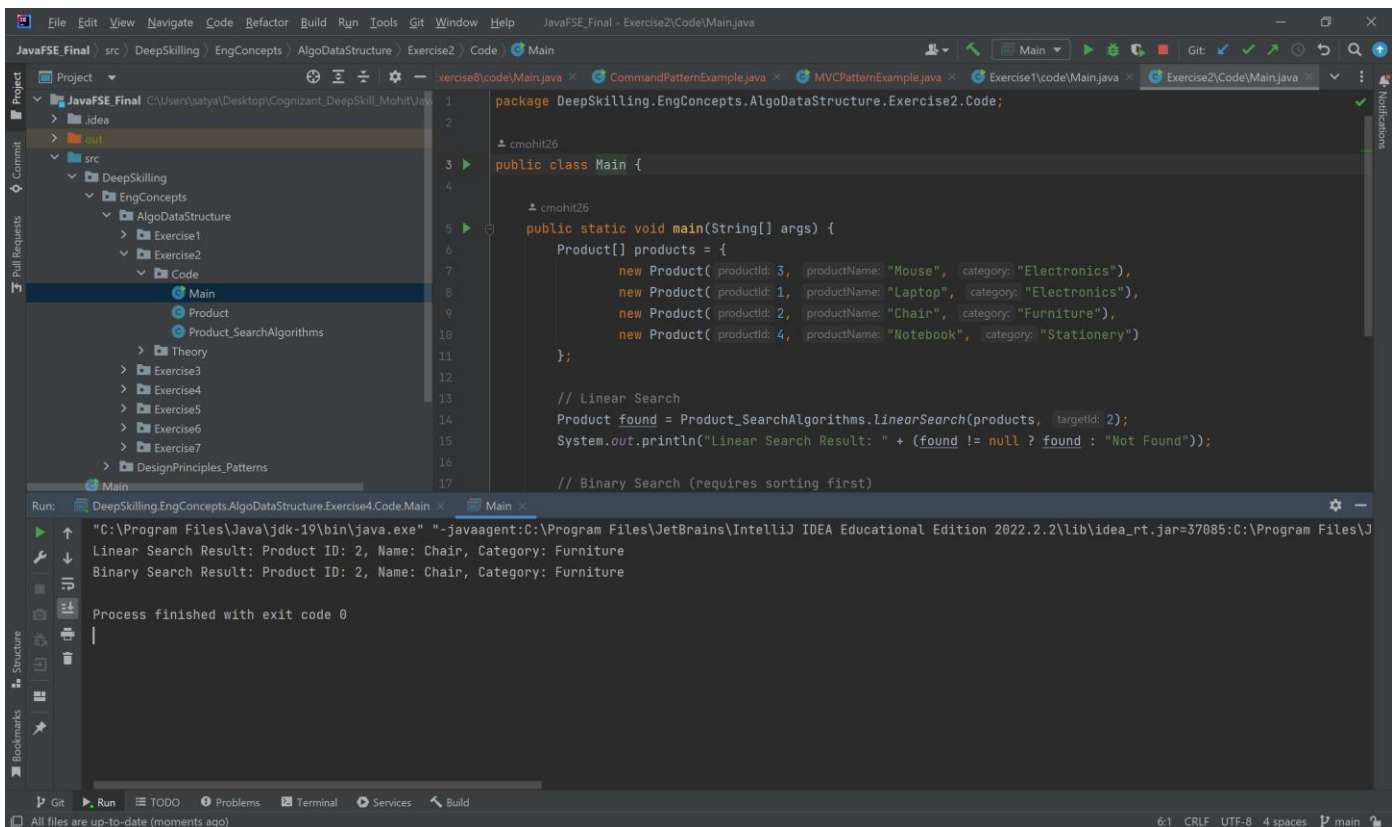
```
package DeepSkilling.EngConcepts.AlgoDataStructure.Exercise1.code;  
  
cmohit26  
public class Main {  
  
    cmohit26  
    public static void main(String[] args) {  
  
        Inventory inventory = new Inventory();  
  
        Product p1 = new Product( productId: 101, productName: "Laptop", quantity: 5, price: 999.99);  
        Product p2 = new Product( productId: 102, productName: "Phone", quantity: 10, price: 599.99);  
  
        inventory.addProduct(p1);  
        inventory.addProduct(p2);  
  
        inventory.displayInventory();  
    }  
}
```

The bottom panel shows the Run output:

```
Run: DeepSkilling.EngConcepts.AlgoDataStructure.Exercise4.Code.Main x DeepSkilling.EngConcepts.AlgoDataStructure.Exercise1.code.Main x  
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2022.2.2\lib\idea_rt.jar=36963:C:\Progr  
Product added.  
Product added.  
Product ID: 101, Name: Laptop, Quantity: 5, Price: $999.99  
Product ID: 102, Name: Phone, Quantity: 10, Price: $599.99  
Product updated.  
Product deleted.  
Product ID: 101, Name: Gaming Laptop, Quantity: 3, Price: $1299.99  
  
Process finished with exit code 0
```

The status bar at the bottom indicates: Build completed successfully in 4 sec, 32 ms (a minute ago). 3:14 CRLF UTF-8 4 spaces main

2. EXERCISE 2



```
package DeepSkillEngConcepts.AlgoDataStructure.Exercise2.Code;

import java.util.*;

public class Main {

    public static void main(String[] args) {
        Product[] products = {
            new Product(3, "Mouse", "Electronics"),
            new Product(1, "Laptop", "Electronics"),
            new Product(2, "Chair", "Furniture"),
            new Product(4, "Notebook", "Stationery")
        };

        // Linear Search
        Product found = ProductSearchAlgorithms.linearSearch(products, 2);
        System.out.println("Linear Search Result: " + (found != null ? found : "Not Found"));

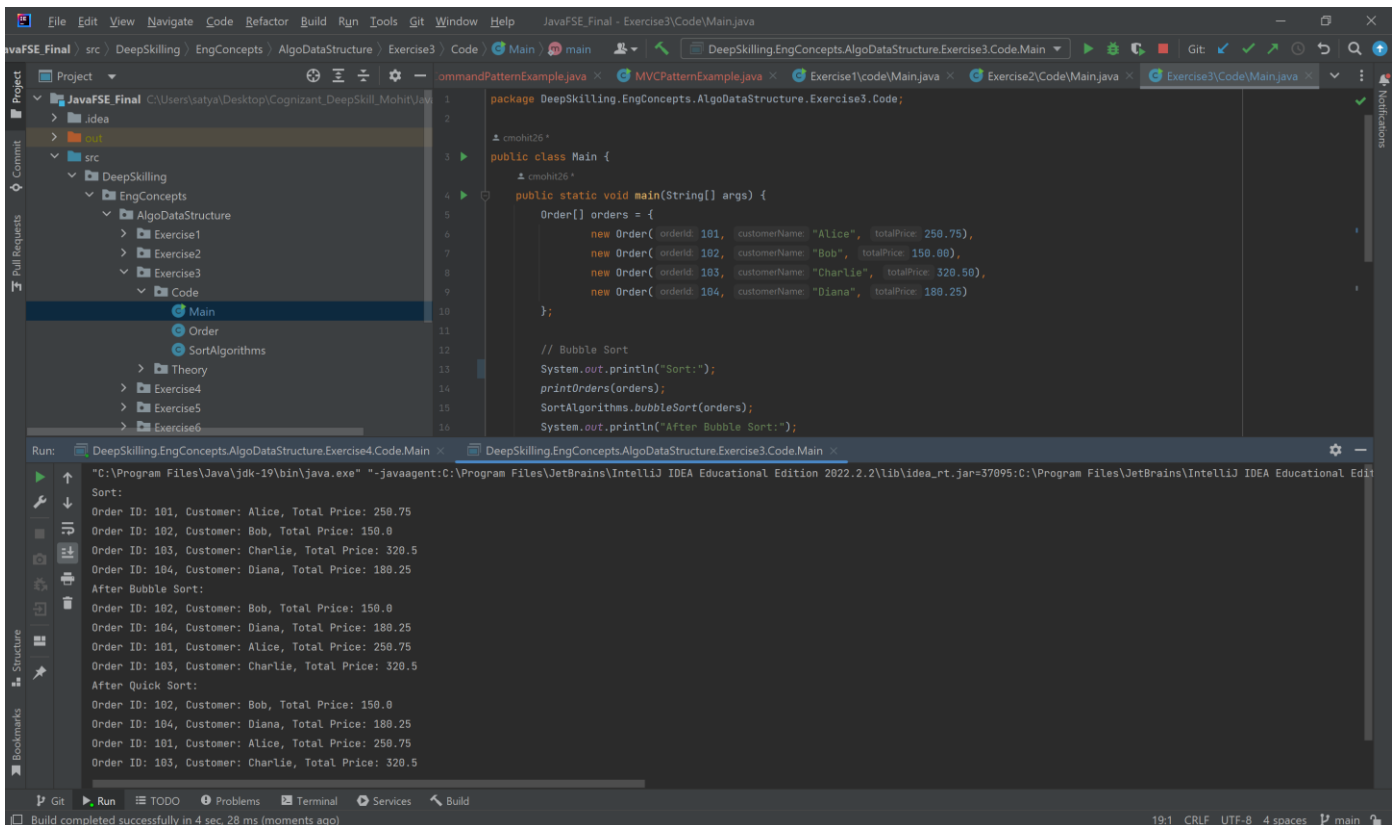
        // Binary Search (requires sorting first)
    }
}
```

Run: DeepSkillEngConcepts.AlgoDataStructure.Exercise4.Code.Main x Main x

```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2022.2.2\lib\idea_rt.jar=37085:C:\Program Files\J"
Linear Search Result: Product ID: 2, Name: Chair, Category: Furniture
Binary Search Result: Product ID: 2, Name: Chair, Category: Furniture

Process finished with exit code 0
```

3. EXERCISE 3



```
package DeepSkillEngConcepts.AlgoDataStructure.Exercise3.Code;

import java.util.*;

public class Main {

    public static void main(String[] args) {
        Order[] orders = {
            new Order(101, "Alice", 250.75),
            new Order(102, "Bob", 150.00),
            new Order(103, "Charlie", 320.50),
            new Order(104, "Diana", 180.25)
        };

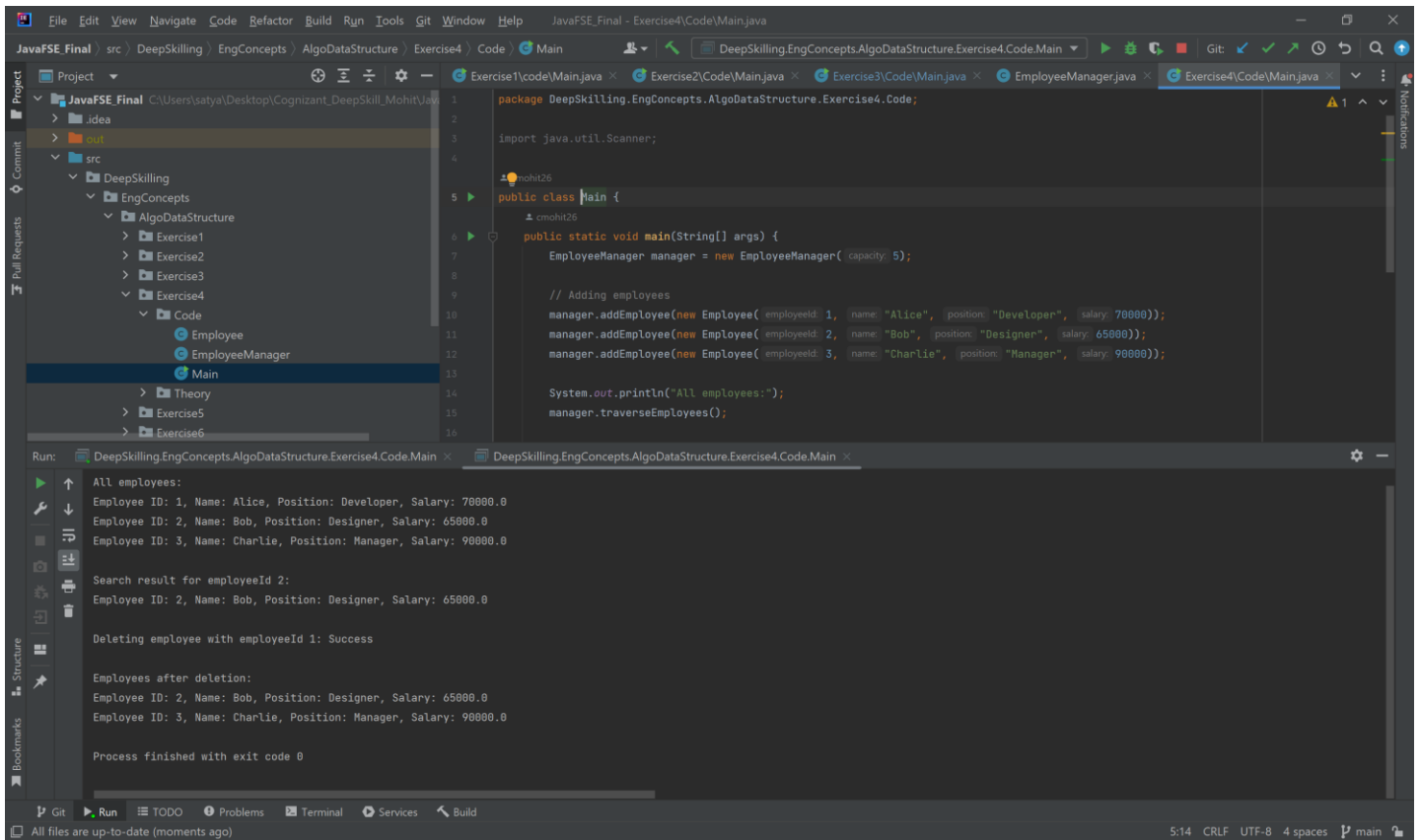
        // Bubble Sort
        System.out.println("Sort:");
        printOrders(orders);
        SortAlgorithms.bubbleSort(orders);
        System.out.println("After Bubble Sort:");

        // Quick Sort
        System.out.println("Sort:");
        printOrders(orders);
        SortAlgorithms.quickSort(orders);
        System.out.println("After Quick Sort:");
    }
}
```

Run: DeepSkillEngConcepts.AlgoDataStructure.Exercise4.Code.Main x DeepSkillEngConcepts.AlgoDataStructure.Exercise3.Code.Main x

```
"C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2022.2.2\lib\idea_rt.jar=37095:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edi"
Sort:
Order ID: 101, Customer: Alice, Total Price: 250.75
Order ID: 102, Customer: Bob, Total Price: 150.0
Order ID: 103, Customer: Charlie, Total Price: 320.5
Order ID: 104, Customer: Diana, Total Price: 180.25
After Bubble Sort:
Order ID: 102, Customer: Bob, Total Price: 150.0
Order ID: 104, Customer: Diana, Total Price: 180.25
Order ID: 101, Customer: Alice, Total Price: 250.75
Order ID: 103, Customer: Charlie, Total Price: 320.5
After Quick Sort:
Order ID: 102, Customer: Bob, Total Price: 150.0
Order ID: 104, Customer: Diana, Total Price: 180.25
Order ID: 101, Customer: Alice, Total Price: 250.75
Order ID: 103, Customer: Charlie, Total Price: 320.5
```

4. EXERCISE 4



```
package DeepSkillEngConcepts.AlgoDataStructure.Exercise4.Code;

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {

        EmployeeManager manager = new EmployeeManager( capacity: 5);

        // Adding employees
        manager.addEmployee(new Employee( employeeId: 1, name: "Alice", position: "Developer", salary: 70000));
        manager.addEmployee(new Employee( employeeId: 2, name: "Bob", position: "Designer", salary: 65000));
        manager.addEmployee(new Employee( employeeId: 3, name: "Charlie", position: "Manager", salary: 90000));

        System.out.println("All employees:");
        manager.traverseEmployees();

    }
}
```

Run: DeepSkillEngConcepts.AlgoDataStructure.Exercise4.Code.Main

All employees:
Employee ID: 1, Name: Alice, Position: Developer, Salary: 70000.0
Employee ID: 2, Name: Bob, Position: Designer, Salary: 65000.0
Employee ID: 3, Name: Charlie, Position: Manager, Salary: 90000.0

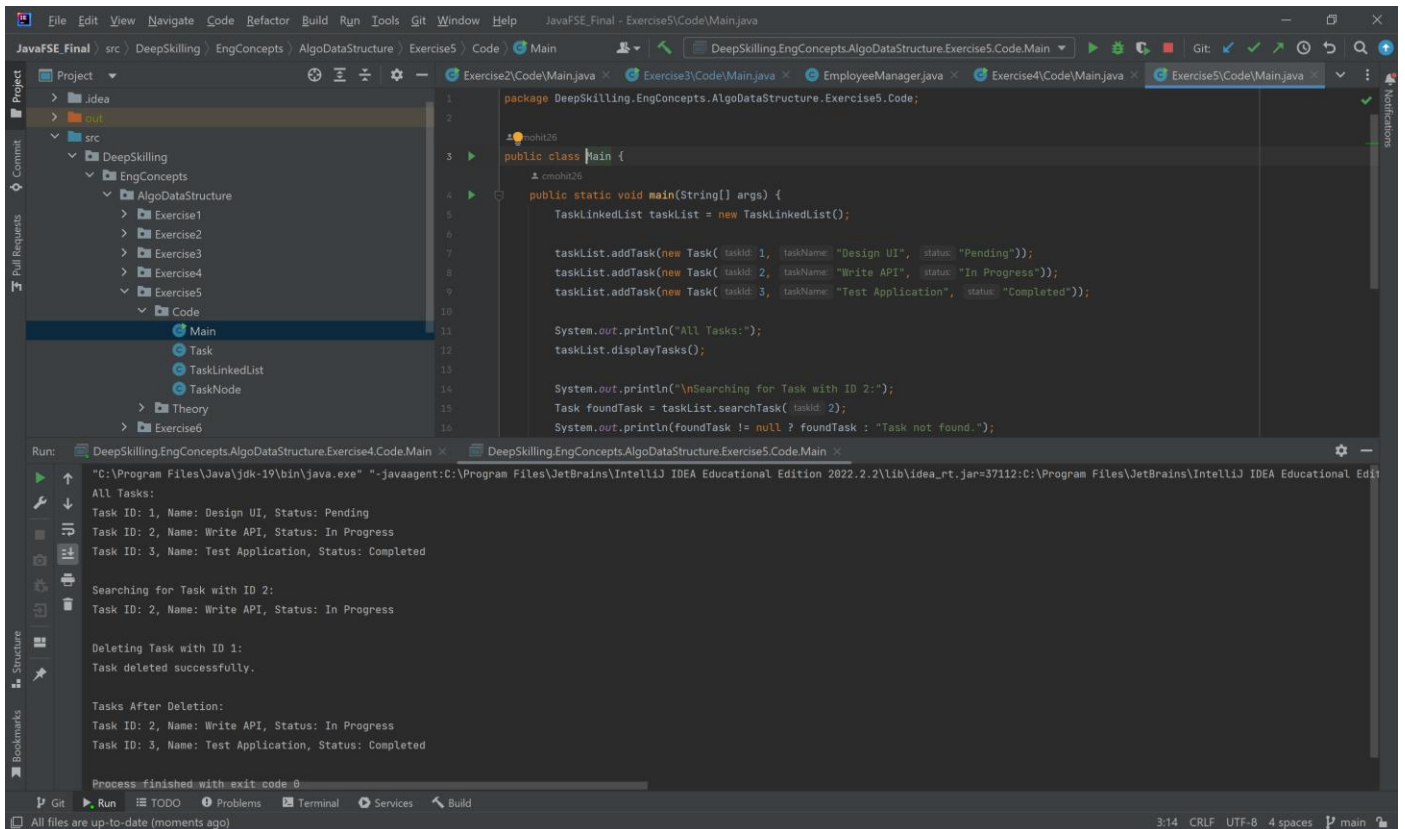
Search result for employeeId 2:
Employee ID: 2, Name: Bob, Position: Designer, Salary: 65000.0

Deleting employee with employeeId 1: Success

Employees after deletion:
Employee ID: 2, Name: Bob, Position: Designer, Salary: 65000.0
Employee ID: 3, Name: Charlie, Position: Manager, Salary: 90000.0

Process finished with exit code 0

5. EXERCISE 5



```
package DeepSkillEngConcepts.AlgoDataStructure.Exercise5.Code;

public class Main {

    public static void main(String[] args) {

        TaskLinkedList taskList = new TaskLinkedList();

        taskList.addTask(new Task( taskId: 1, taskName: "Design UI", status: "Pending"));
        taskList.addTask(new Task( taskId: 2, taskName: "Write API", status: "In Progress"));
        taskList.addTask(new Task( taskId: 3, taskName: "Test Application", status: "Completed"));

        System.out.println("All Tasks:");
        taskList.displayTasks();

        System.out.println("\nSearching for Task with ID 2:");
        Task foundTask = taskList.searchTask( taskId: 2);
        System.out.println(foundTask != null ? foundTask : "Task not found.");

    }
}
```

Run: DeepSkillEngConcepts.AlgoDataStructure.Exercise5.Code.Main

All Tasks:
Task ID: 1, Name: Design UI, Status: Pending
Task ID: 2, Name: Write API, Status: In Progress
Task ID: 3, Name: Test Application, Status: Completed

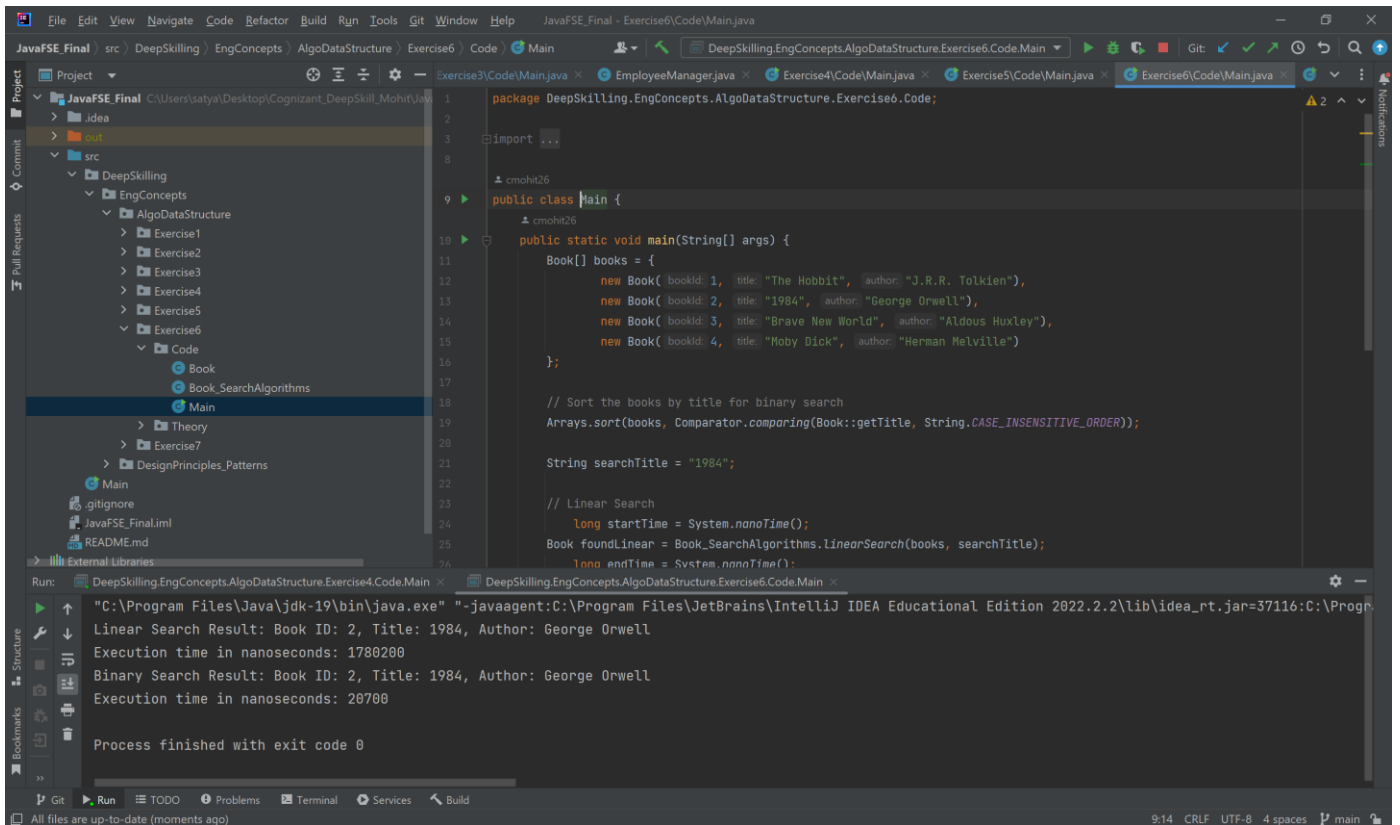
Searching for Task with ID 2:
Task ID: 2, Name: Write API, Status: In Progress

Deleting Task with ID 1:
Task deleted successfully.

Tasks After Deletion:
Task ID: 2, Name: Write API, Status: In Progress
Task ID: 3, Name: Test Application, Status: Completed

Process finished with exit code 0

6. EXERCISE 6



```
package DeepSkillEngConcepts.AlgoDataStructure.Exercise6.Code;

import ...

cmohit26
public class Main {

cmohit26
    public static void main(String[] args) {
        Book[] books = {
            new Book( bookid: 1, title: "The Hobbit", author: "J.R.R. Tolkien"),
            new Book( bookid: 2, title: "1984", author: "George Orwell"),
            new Book( bookid: 3, title: "Brave New World", author: "Aldous Huxley"),
            new Book( bookid: 4, title: "Moby Dick", author: "Herman Melville")
        };

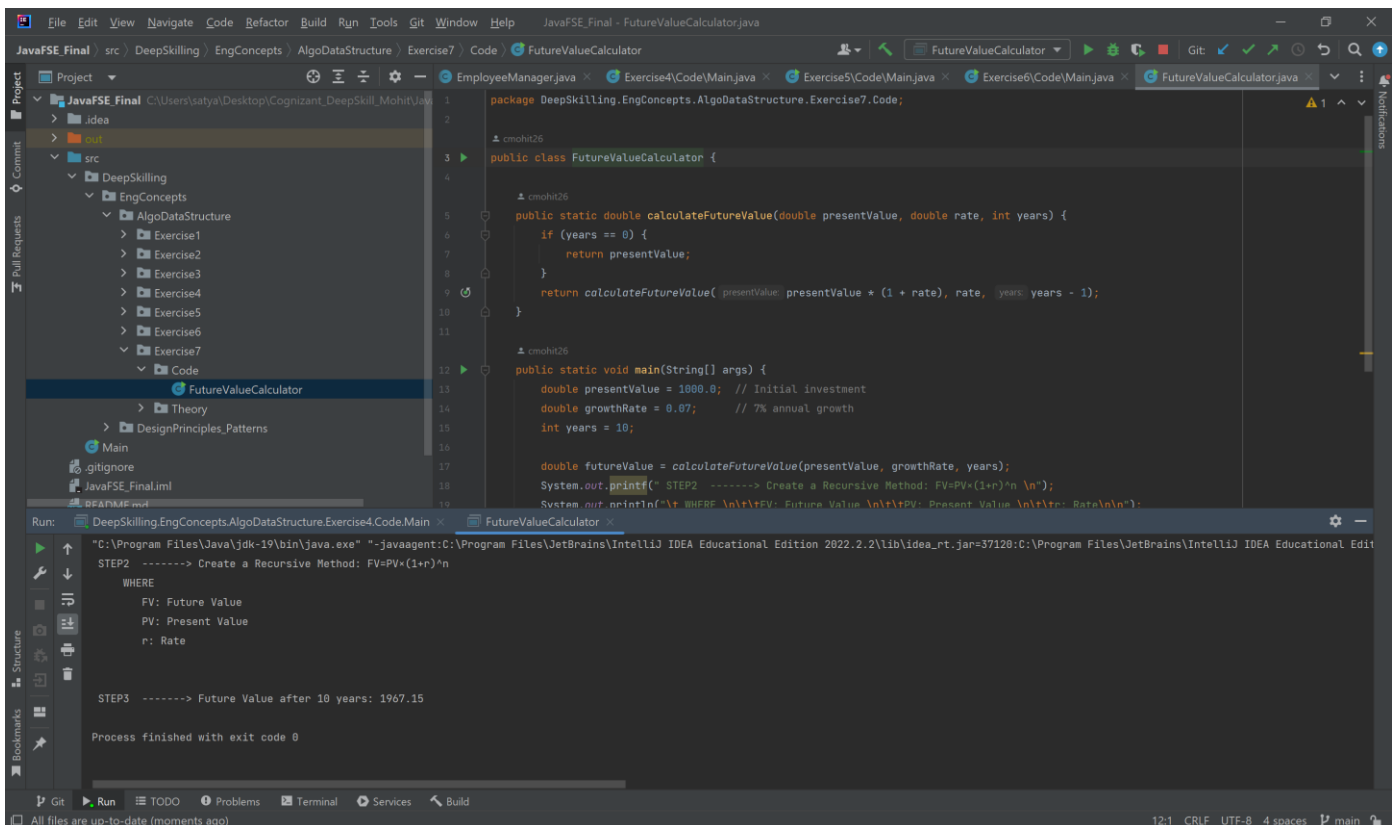
        // Sort the books by title for binary search
        Arrays.sort(books, Comparator.comparing(Book::getTitle, String.CASE_INSENSITIVE_ORDER));

        String searchText = "1984";

        // Linear Search
        long startTime = System.nanoTime();
        Book foundLinear = Book_SearchAlgorithms.linearSearch(books, searchText);
        long endTime = System.nanoTime();

        Run: DeepSkillEngConcepts.AlgoDataStructure.Exercise4.Code.Main x DeepSkillEngConcepts.AlgoDataStructure.Exercise6.Code.Main x
        "C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2022.2.2\lib\idea_rt.jar=37116:C:\Progr
        Linear Search Result: Book ID: 2, Title: 1984, Author: George Orwell
        Execution time in nanoseconds: 1780200
        Binary Search Result: Book ID: 2, Title: 1984, Author: George Orwell
        Execution time in nanoseconds: 20700
        Process finished with exit code 0
```

7. EXERCISE 7



```
package DeepSkillEngConcepts.AlgoDataStructure.Exercise7.Code;

cmohit26
public class FutureValueCalculator {

cmohit26
    public static double calculateFutureValue(double presentValue, double rate, int years) {
        if (years == 0) {
            return presentValue;
        }
        return calculateFutureValue(presentValue * (1 + rate), rate, years - 1);
    }

cmohit26
    public static void main(String[] args) {
        double presentValue = 1000.0; // Initial investment
        double growthRate = 0.07; // 7% annual growth
        int years = 10;

        double futureValue = calculateFutureValue(presentValue, growthRate, years);
        System.out.println(" STEP2 -----> Create a Recursive Method: FV=PV*(1+r)^n\n");
        System.out.println("\t BHFRE \n\t FV: Future Value \n\t PV: Present Value \n\t r: Rate\n\t n:");

        Run: DeepSkillEngConcepts.AlgoDataStructure.Exercise4.Code.Main x FutureValueCalculator x
        "C:\Program Files\Java\jdk-19\bin\java.exe" "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edition 2022.2.2\lib\idea_rt.jar=37120:C:\Program Files\JetBrains\IntelliJ IDEA Educational Edit
        STEP2 -----> Create a Recursive Method: FV=PV*(1+r)^n
        WHERE
        FV: Future Value
        PV: Present Value
        r: Rate

        STEP3 -----> Future Value after 10 years: 1967.15
        Process finished with exit code 0
```