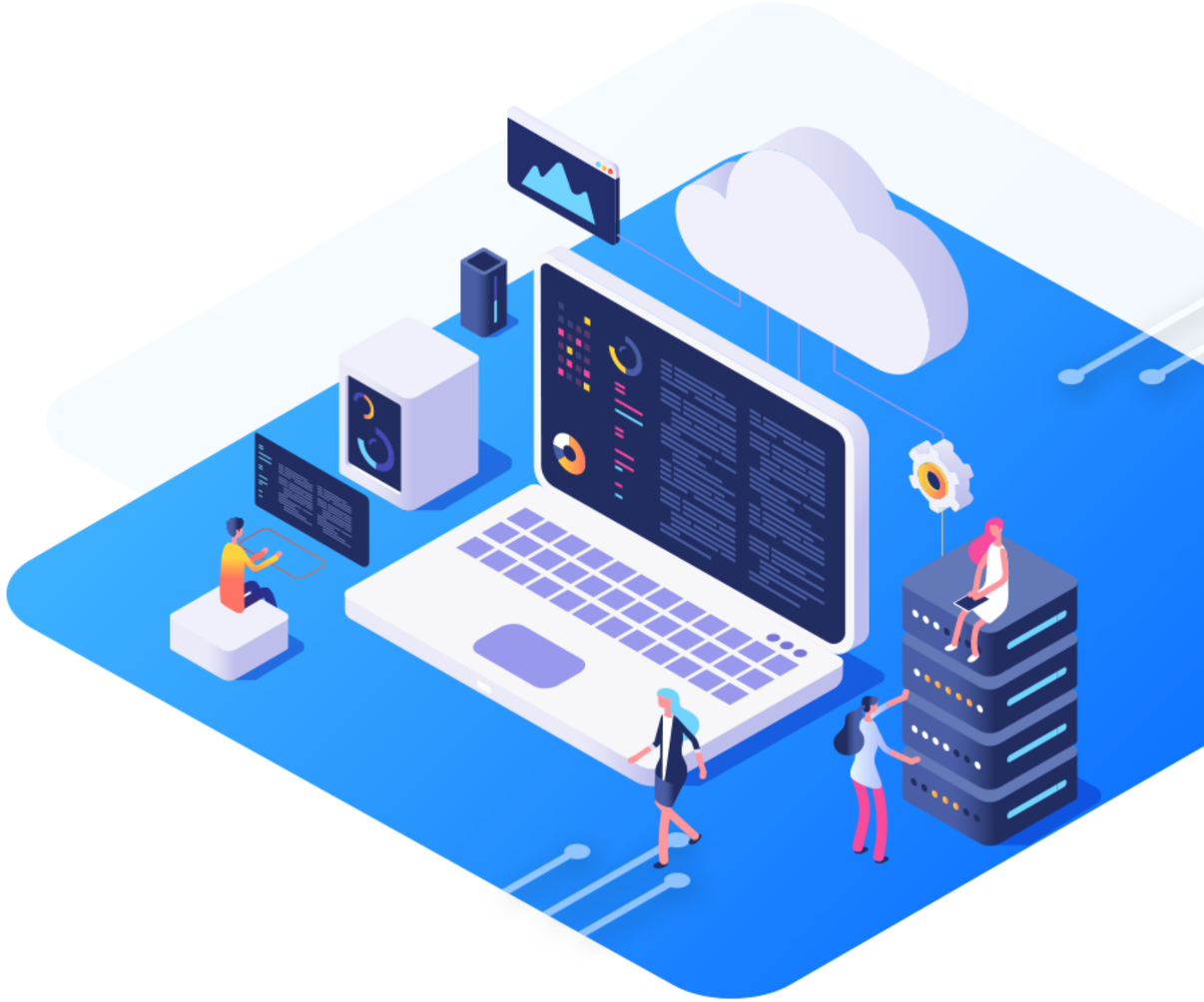
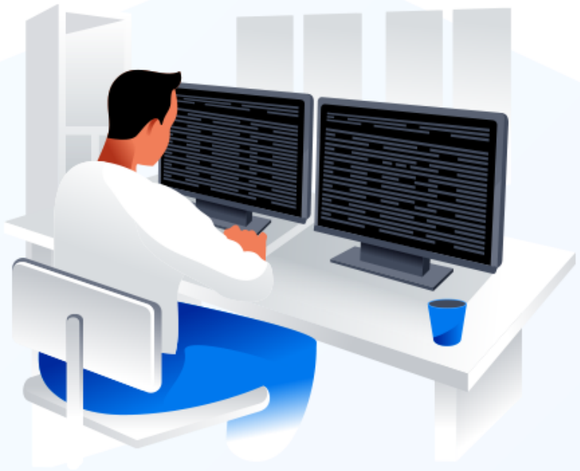
Data Structures and Algorithms

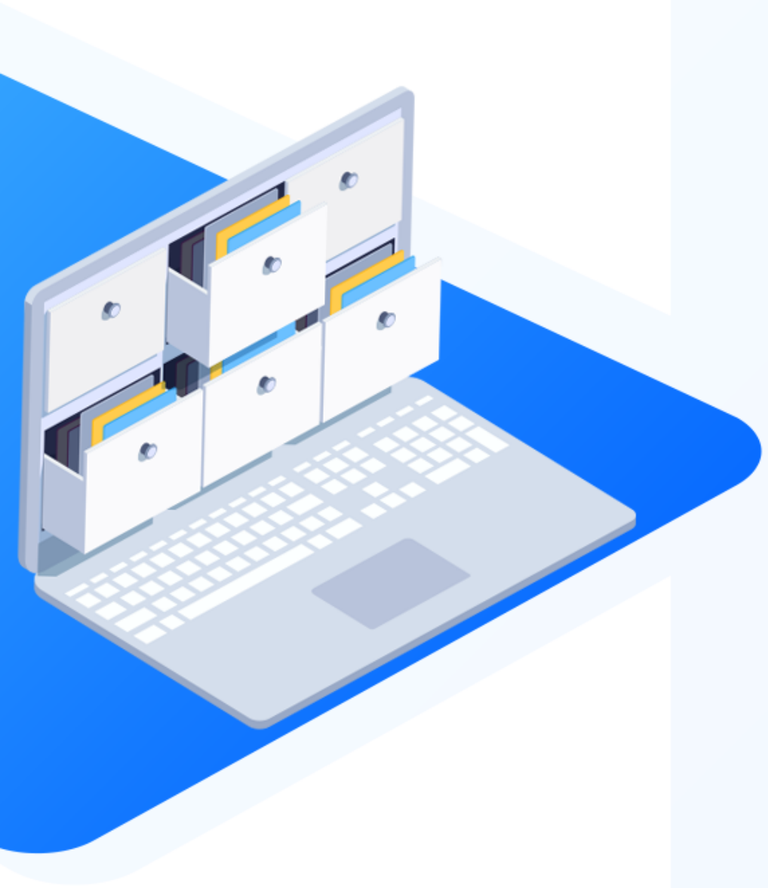


Phase-End Project



Express Banking System

Objective

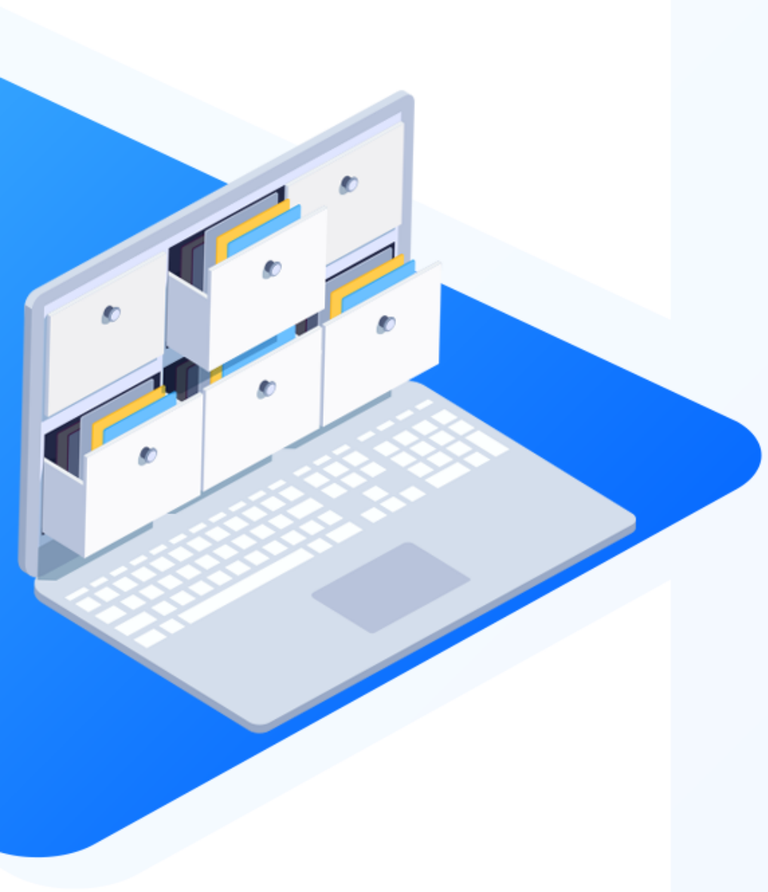


The primary aim of this Express Banking System project is to develop a secure and user-friendly application that streamlines and optimizes banking operations.

This system is designed to empower banking professionals to effectively manage bank accounts, facilitate money transfers, and simplify the account creation process. Moreover, customers will be able to create accounts, check their balances, and perform transactions. This project integrates the use of linear and non-linear data structures, and basic algorithms for functionality, providing a comprehensive understanding of JavaScript and data structures.

PHASE-END PROJECT

Problem Statement and Motivation

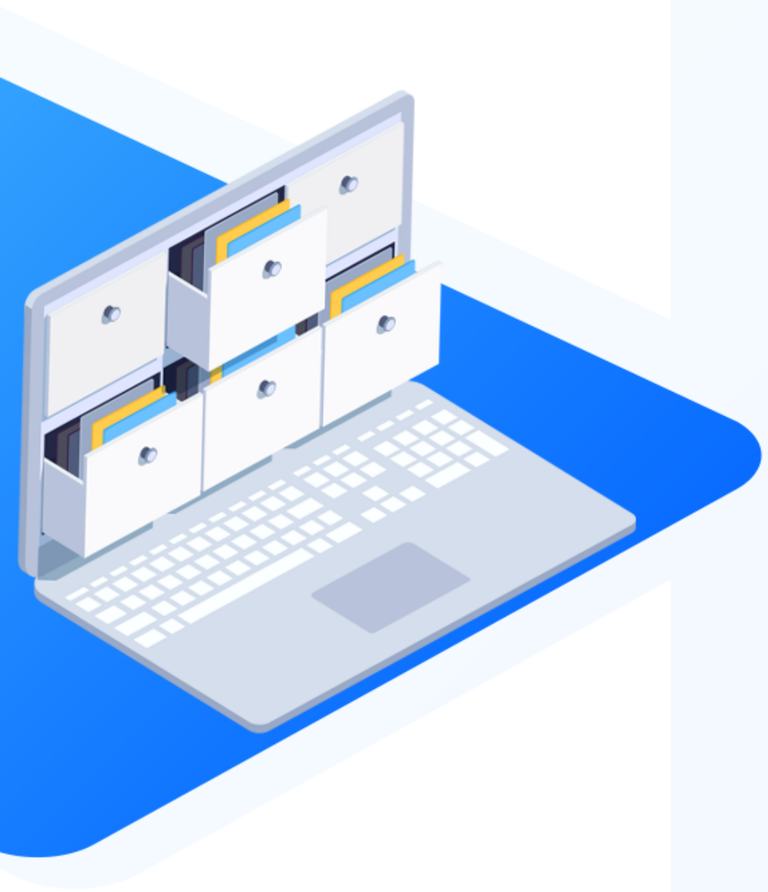


Real-time scenario:

In today’s digital age, managing banking operations manually can be time-consuming and prone to errors. These operations involve managing customer accounts, handling transactions, and overseeing account transfers. Traditional methods can be inefficient, lack security, and are not scalable as the number of customers increases.

The development of a computerized Express Banking System would address the shortcomings of the manual system and offer several advantages: improved efficiency, enhanced accuracy, and better customer service.

Industry Relevance



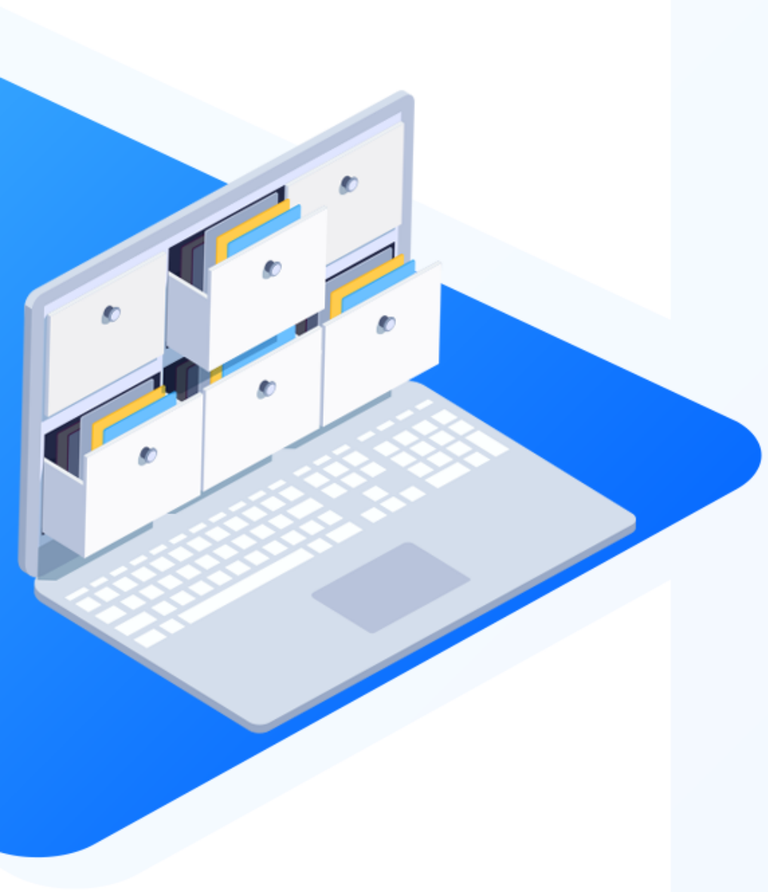
The following tools are widely used in the industry:

1. **JavaScript:** It is a high-level, interpreted programming language that is widely used in the industry to make web pages interactive.

2. **Data Structures:** It refers to specific arrangements designed for organizing, processing, retrieving, and storing data.

3. **Algorithms:** It involves clear and unambiguous methods for performing computations, which could encompass tasks such as calculations, data processing, automated reasoning, and more.

Tasks



The following tasks for developing the Express Banking System using JavaScript:

1. Start the project by setting up tools and libraries, including a Node.js environment

2. Construct the Node, LinkedList, and TreeNode classes for managing bank accounts

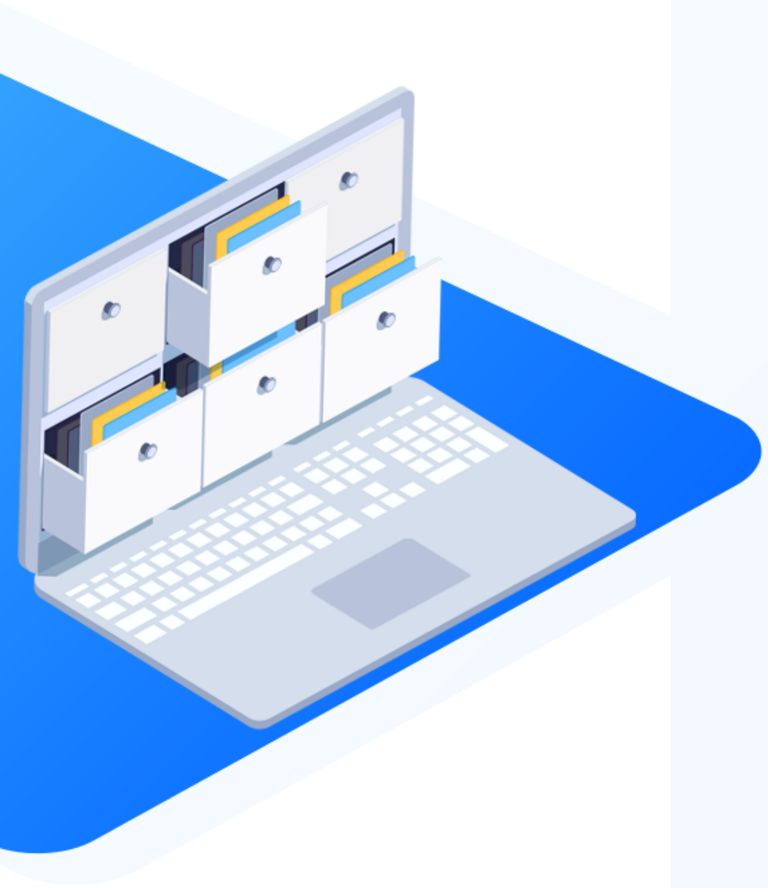
3. Create a Bank Class to manage account operations using previous data structures

4. Devise a Bank class method to add new accounts, using account number and balance, to the account structures

5. Establish transfer and balance-check methods in the Bank class, utilizing account numbers and transfer amounts

PHASE-END PROJECT

Project References



- **Task 1:** Course 2 Lesson 4
- **Tasks 2:** Course 3 Lesson 2, and Lesson 3
- **Task 3:** Course 3 Lesson 1
- **Tasks 4 and 5:** Course 2 Lesson 5

PHASE-END PROJECT

Thank You