

OOP Project

Banking System Management

~Tanvi Gupta (230911122), Roll No.-20

~Akshat Yuwan (230911192), Roll No.-26

~Simran Singhania (230911126), Roll No.-21

This project is a basic banking application designed to model core operations of a banking system, allowing users to interact with different types of accounts in a simplified, simulated environment. The focus is on building a program that can perform essential banking transactions, such as creating accounts, depositing and withdrawing money, transferring funds, and applying interest. It's ideal for demonstrating foundational concepts in programming, especially within Object-Oriented Programming (OOP).

Project Goals

Account Management: Provide a structure for creating and managing different types of bank accounts.

Transaction Handling: Allow standard banking transactions like deposits, withdrawals, and fund transfers.

Special Account Features: Include unique functionalities for different account types, such as interest application for savings accounts and overdraft support for checking accounts.

Error Handling: Gracefully manage errors (like insufficient funds) to ensure a stable and user-friendly system.

Functional Requirements:

Account Creation:

The application can create different types of accounts: savings accounts and checking accounts.

Each account has basic information, including an account number, owner's name, and balance.

Balance Inquiry:

Users can check the current balance of any account.

For simplicity, balance inquiries display the balance directly, allowing users to see their financial status quickly.

Deposit:

Users can deposit money into any account, which increases the account's balance.

Deposits can only be positive amounts to ensure data integrity.

Withdrawal:

The application allows users to withdraw funds from an account, as long as there are sufficient funds.

Different accounts handle withdrawals differently:

Savings Account: Only allows withdrawals up to the available balance.

Checking Account: Allows withdrawals up to the balance plus an overdraft limit.

Interest Application:

Savings accounts have an interest rate applied to their balance periodically, which simulates interest accumulation over time.

Interest application increases the account balance based on the specified rate.

Transfer Between Accounts:

The application enables users to transfer funds between two accounts, simulating common banking transfers.

Transfers adhere to each account's withdrawal rules, ensuring that funds cannot be transferred out if they exceed the account's limits.

Key Concepts Illustrated in the Project:

Types of Bank Accounts:

Savings Account: Primarily for saving funds and earning interest. It does not allow overdraft.

Checking Account: Used for frequent transactions, including payments. It allows for overdrafts within a set limit.

Transaction Logic:

The project defines a logical flow for handling each type of transaction, ensuring that deposits, withdrawals, and transfers respect each account's rules and limitations.

Error Handling with Custom Exceptions:

Insufficient funds raise custom errors, guiding users to correct actions without crashing the application. This ensures a better user experience and application reliability.

Interest Calculation:

For savings accounts, interest accrues at a defined rate. This calculation reflects the accumulation of wealth over time, as would happen in a real savings account.