Credit Name: Chapter 8
Assignment Name:Bank

How has your program changed from planning to coding to now? Please explain?

At first, the program was planned to demonstrate how inheritance works by creating a base class, Account, and two subclasses, PersonalAcct and BusinessAcct. The idea was simple: Account would handle shared details like balance and basic methods like deposit(), while the subclasses would add specific behaviors, such as penalties for low balances in withdraw(). The plan also included using an abstract method, withdraw(), that each subclass would implement to reflect its unique rules.

When I started coding, I encountered some challenges that required adjustments. For example: The subclasses had to properly implement the withdraw() method from the abstract class, or else the program wouldn't compile.I realized that I needed to use super() in the constructors of PersonalAcct and BusinessAcct to call the Account constructor and initialize the balance field. Adding user input for transactions and initial balances made the program more interactive, but it also required handling errors, such as invalid amounts or insufficient funds.
 Scanner input issues caused skipped inputs, which needed careful management of the input buffer.

Now, the program is more polished. It works as planned but also includes features I didn't initially consider, like validation for deposits and withdrawals to handle mistakes gracefully. The withdraw() method in each subclass now applies penalties correctly, and user input makes the program dynamic and interactive. What started as a simple idea has become a complete, functional program that demonstrates inheritance, abstract methods, and user interaction effectively. It's a good balance between simplicity and functionality.