ATM Implementation Code

```
import java.util.*;

public class ATMImplementation implements ATMinterface {
    ATM atm = new ATM();
    Map<String, Double> miniStatement = new LinkedHashMap<>();

    @Override
    public void viewBalance() {
        System.out.println("Available Balance: ■" + atm.getBalance());
    }

    @Override
    public void withdrawAmount(double withdrawAmount) {
        if (withdrawAmount <= 0) {
            System.out.println("Invalid amount! Please enter a positive value.");
        } else if (withdrawAmount > atm.getBalance()) {
            System.out.println("Insufficient balance! Transaction failed.");
        } else {
            atm.setBalance(atm.getBalance() - withdrawAmount);
            System.out.println("Successfully withdrawn ■" + withdrawAmount);
            miniStatement.put("Withdrawn", withdrawAmount);
            viewBalance();
        }
    }

    @Override
    public void depositAmount(double depositAmount) {
        if (depositAmount <= 0) {
            System.out.println("Invalid deposit amount! Please enter a positive value.");
        } else {
            atm.setBalance(atm.getBalance() + depositAmount);
            System.out.println("Successfully deposited ■" + depositAmount);
            miniStatement.put("Deposited", depositAmount);
            viewBalance();
        }
    }

    @Override
    public void viewMiniStatement() {
        System.out.println("\n===== MINI STATEMENT =====");
        if (miniStatement.isEmpty()) {
            System.out.println("No recent transactions.");
        } else {
            for (Map.Entry<String, Double> entry : miniStatement.entrySet()) {
                System.out.println(entry.getKey() + ": ■" + entry.getValue());
            }
        }
        System.out.println("Current Balance: ■" + atm.getBalance());
        System.out.println("==========================\n");
    }
}
```