```
# 2.1 Implement a class called BankAccount that represents a bank account. The class
should have private attributes for account number, account holder name, and account
balance. Include methods to deposit money, withdraw money, and display the account
balance. Ensure that the account balance cannot be accessed directly from outside the
class. Write a program to create an instance of the BankAccount class and test the
deposit and withdrawal functionality.

class BankAccount:

    def __init__(self, account_number, account_holder_name, initial_balance=0.0):

        self.__account_number = account_number

        self.__account_holder_name = account_holder_name

        self.__account_balance = initial_balance


    def deposit(self, amount):

        if amount > 0:

            self.__account_balance += amount

            print(f"Deposited ${amount:.2f} into account {self.__account_number}")

        else:

            print("Invalid deposit amount. Please deposit a positive amount.")


    def withdraw(self, amount):

        if amount > 0:

            if self.__account_balance >= amount:

                self.__account_balance -= amount

                print(f"Withdrew ${amount:.2f} from account {self.__account_number}")

            else:

                print("Insufficient balance. Cannot withdraw.")

        else:

            print("Invalid withdrawal amount. Please withdraw a positive amount.")


    def display_balance(self):
```

```
        print(f"Account {self.__account_number} balance:
${self.__account_balance:.2f}")



# Testing the BankAccount class

if _name_ == "__main__":

    # Create a BankAccount instance

    account1 = BankAccount("123456", "John Doe", 1000.0)


    # Deposit money

    account1.deposit(500.0)


    # Withdraw money

    account1.withdraw(200.0)


    # Display balance

    account1.display_balance()
```