1.Write a java program to create a class known as “BankAccount” with methods called deposit() and withdraw(). Create a subclass called SavingsAccount that overrides the withdraw() method to prevent withdrawals if the account balance falls below one hundred.

```
class BankAccount {
    private String accountNumber;
    private double balance;

    public BankAccount(String accountNumber, double balance) {
        this.accountNumber = accountNumber;
        this.balance = balance;
    }

    public void deposit(double amount) {
        balance += amount;
    }

    public void withdraw(double amount) {
        if (balance >= amount) {
            balance -= amount;
        } else {
            System.out.println("Insufficient balance");
        }
    }

    public double getBalance() {
        return balance;
    }
}
// SavingsAccount.java
// Child class SavingsAccount
class SavingsAccount extends BankAccount {
    public SavingsAccount(String accountNumber, double balance) {
        super(accountNumber, balance);
    }

    @Override
    public void withdraw(double amount) {
        if (getBalance() - amount < 100) {
            System.out.println("Minimum balance of $100 required!");
        } else {
            super.withdraw(amount);
        }
    }
```

```
}
 // Main.java
// Main class
public class Main {
    public static void main(String[] args) {
        System.out.println("Create a Bank Account object (A/c No. BA1234) with initial balance of
$500:");
                    //Create a BankAccount object (A/c No. "BA1234") with initial balance of $500
        BankAccount BA1234 = new BankAccount("BA1234", 500);

        // Deposit $1000 into account BA1234
                    System.out.println("Deposit $1000 into account BA1234:");
        BA1234.deposit(1000);
        System.out.println("New balance after depositing $1000: $" + BA1234.getBalance());

        // Withdraw $600 from account BA1234
                    System.out.println("Withdraw $600 from account BA1234:");
        BA1234.withdraw(600);
        System.out.println("New balance after withdrawing $600: $" + BA1234.getBalance());

        // Create a SavingsAccount object (A/c No. "SA1234") with initial balance of $450
                    System.out.println("\nCreate a SavingsAccount object (A/c No. SA1234) with
initial balance of $450:");
        SavingsAccount SA1234 = new SavingsAccount("SA1234",450);

        // Withdraw $300 from SA1234
        SA1234.withdraw(300);
        System.out.println("Balance after trying to withdraw $300: $" + SA1234.getBalance());



                    // Create a SavingsAccount object (A/c No. "SA1000") with initial balance of $300
                    System.out.println("\nCreate a SavingsAccount object (A/c No. SA1000) with
initial balance of $300:");
        SavingsAccount SA1000 = new SavingsAccount("SA1000",300);

        // Withdraw $250 from SA1000 (balance falls below $100)
                    System.out.println("Try to withdraw $250 from SA1000!");
        SA1000.withdraw(250);
        System.out.println("Balance after trying to withdraw $250: $" + SA1000.getBalance());

    }
}
```

Output:-

```
java -cp /tmp/3OQko3m0pj/Main
Create a Bank Account object (A/c No. BA1234) with initial balance of $500:
Deposit $1000 into account BA1234:
New balance after depositing $1000: $1500.0
Withdraw $600 from account BA1234:
New balance after withdrawing $600: $900.0

Create a SavingsAccount object (A/c No. SA1234) with initial balance of $450:
Balance after trying to withdraw $300: $150.0

Create a SavingsAccount object (A/c No. SA1000) with initial balance of $300:
Try to withdraw $250 from SA1000!
Minimum balance of $100 required!
Balance after trying to withdraw $250: $300.0

=== Code Execution Successful ===
```