Write a Java program to create a class called "BankAccount" with attributes for account number, account holder's name, and balance. Include methods for depositing and withdrawing money, as well as checking the balance. Create a subclass called "SavingsAccount" that adds an interest rate attribute and a method to apply interest.

Sample Solution:

Java Code:

BankAccount.java

```
        // Define the BankAccount class
        public class BankAccount {

            // Attribute for account number
            private String accountNumber;

            // Attribute for account holder's name
            private String accountHolderName;

            // Attribute for account balance
            private double balance;

            // Constructor to initialize BankAccount object
            public BankAccount(String accountNumber, String accountHolderName, doub
                this.accountNumber = accountNumber;
                this.accountHolderName = accountHolderName;
                this.balance = initialBalance;
            }

            // Method to deposit money into the account
            public void deposit(double amount) {
                if (amount > 0) {
                    balance += amount;
                    System.out.println("Deposited: " + amount + ". New balance: " +
                } else {
                    System.out.println("Deposit amount must be positive.");
                }
            }

            // Method to withdraw money from the account
            public void withdraw(double amount) {
                if (amount > 0 && amount <= balance) {
                    balance -= amount;
                    System.out.println("Withdrew: " + amount + ". New balance: " +
                } else {
                    System.out.println("Insufficient balance or invalid amount.");
                }
            }

            // Method to check the account balance
            public double checkBalance() {
                return balance;
            }

            // Getter method for account number
            public String getAccountNumber() {
                return accountNumber;
            }

            // Getter method for account holder's name
            public String getAccountHolderName() {
                return accountHolderName;
            }
        }
```

The above Java code defines a BankAccount class with three attributes: accountNumber, accountHolderName, and balance. It includes a constructor to initialize these attributes when an object of this class is created. The class also provides methods to:

- Deposit money (deposit): Adds a specified amount to the balance if the amount is positive.
- Withdraw money (withdraw): Subtracts a specified amount from the balance if the amount is positive and does not exceed the current balance.
- Check balance (checkBalance): Returns the current balance.
- Get account number (getAccountNumber): Returns the account number.
- Get the account holder's name (getAccountHolderName): Returns the account holder's name.

SavingsAccount.java

```
// Define the SavingsAccount subclass that extends BankAccount
class SavingsAccount extends BankAccount {

    // Attribute for interest rate
    private double interestRate;

    // Constructor to initialize SavingsAccount object
    public SavingsAccount(String accountNumber, String accountHolderName, 
        super(accountNumber, accountHolderName, initialBalance); // Call th
        this.interestRate = interestRate;
    }

    // Method to apply interest to the balance
    public void applyInterest() {
        double interest = checkBalance() * interestRate / 100; // Calculate
        deposit(interest); // Add interest to the balance
        System.out.println("Interest applied: " + interest + ". New balance
    }

    // Getter method for interest rate
    public double getInterestRate() {
        return interestRate;
    }

    // Setter method for interest rate
    public void setInterestRate(double interestRate) {
        if (interestRate > 0) {
            this.interestRate = interestRate;
        } else {
            System.out.println("Interest rate must be positive.");
        }
    }
}
```

The above Java code defines a SavingsAccount class that extends the BankAccount class, inheriting its attributes and methods. The SavingsAccount class adds a new attribute, interestRate, which represents the interest rate for the savings account. The class includes:

- Constructor: Initializes the SavingsAccount object with the account number, account holder's name, initial balance, and interest rate by calling the superclass (BankAccount) constructor for the common attributes.

- Method to apply interest (applyInterest): Calculates interest based on the current balance and interest rate, then deposits interest into the account.
- Getter method for interest rate (getInterestRate): Returns the current interest rate.
- Setter method for interest rate (setInterestRate): Updates the interest rate if the provided rate is positive, ensuring it is a valid interest rate.

Main.java

```
// Main class to test the BankAccount and SavingsAccount classes
public class Main {
    public static void main(String[] args) {
        // Create a BankAccount object
        BankAccount account = new BankAccount("123456789", "Henri Lionel",
        System.out.println("Current balance: " + account.checkBalance()); /
        account.deposit(4000.0); // Deposit money
        account.withdraw(3000.0); // Withdraw money
        System.out.println("Current balance: " + account.checkBalance()); /

        // Create a SavingsAccount object
        SavingsAccount savings = new SavingsAccount("888888888", "Amphitri
        savings.applyInterest(); // Apply interest
        System.out.println("Savings account balance: " + savings.checkBala
    }
}
```

The above Java code defines a Main class with a main method to test the BankAccount and SavingsAccount classes. The main method demonstrates the creation and usage of these classes:

- Create a BankAccount object: An instance of BankAccount is created with the account number "123456789", account holder's name "Henri Lionel", and an initial balance of 1000.0. The current balance is printed.
- Deposit money: 4000.0 is deposited into the BankAccount.
- Withdraw money: 3000.0 is withdrawn from the BankAccount.
- Check balance: The current balance is printed again.
- Create a SavingsAccount object: An instance of SavingsAccount is created with the account number "888888888", account holder's name "Amphitrite Jun", an initial balance of 2000.0, and

an interest rate of 5.0.

- Apply interest: Interest is applied to the SavingsAccount, and the new balance is printed.

This code tests the functionality of depositing, withdrawing, and applying interest in bank accounts.

Sample Output:

```
Current balance: 1000.0
Deposited: 4000.0. New balance: 5000.0
Withdrew: 3000.0. New balance: 2000.0
Current balance: 2000.0
Deposited: 100.0. New balance: 2100.0
Interest applied: 100.0. New balance: 2100.0
Savings account balance: 2100.0
```

Java Code Editor: