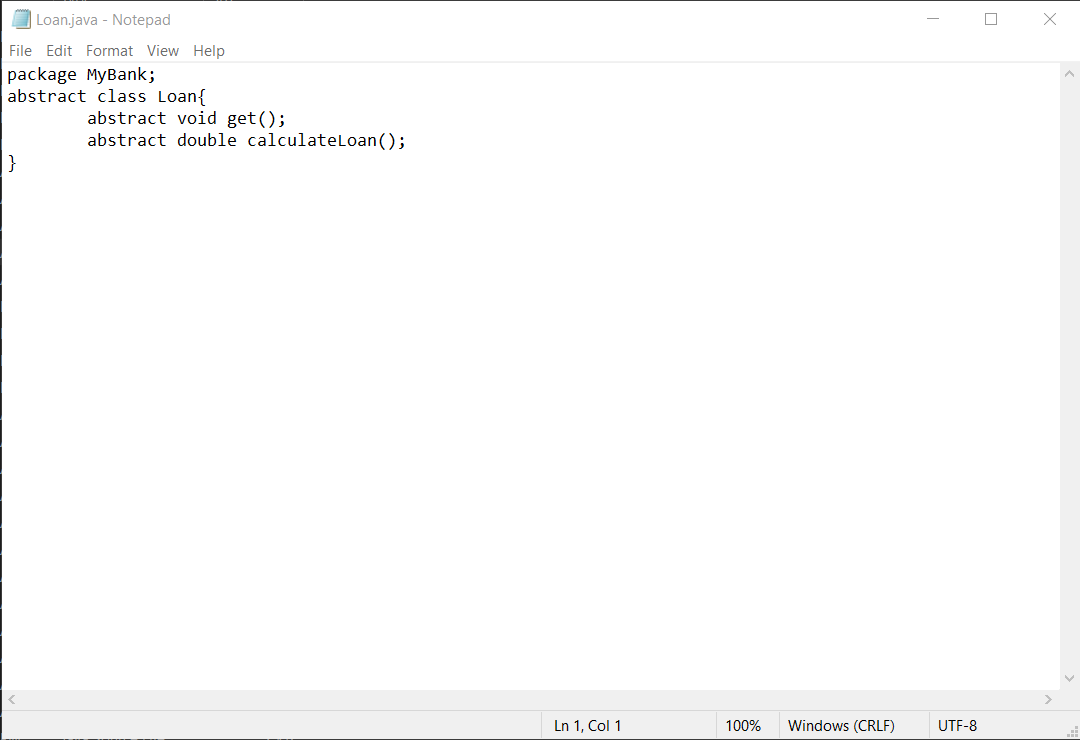
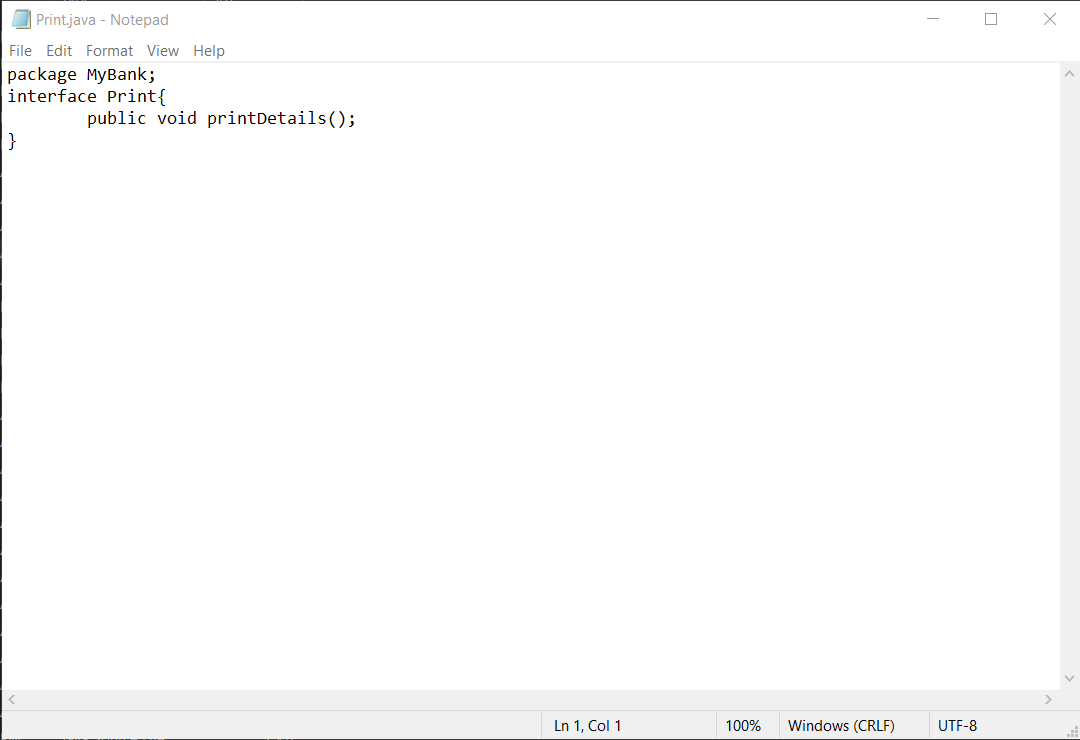
**1)**

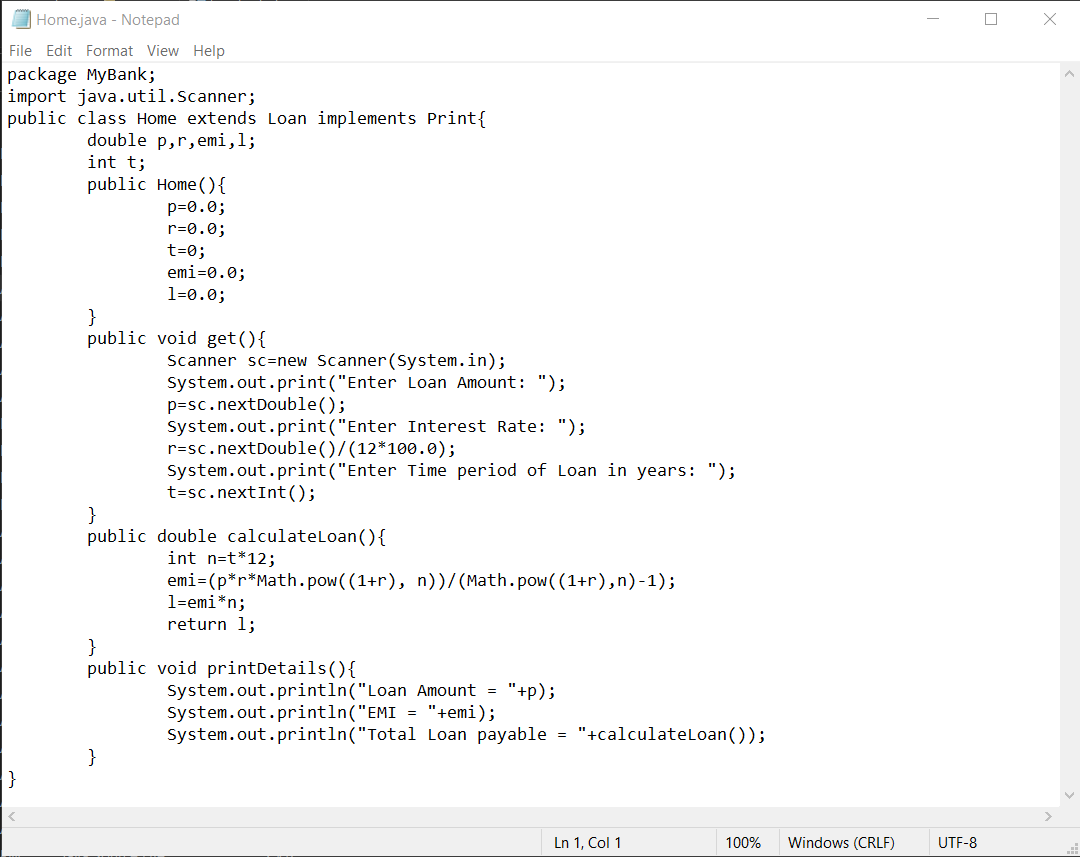
A nationalized bank is offering loans to buy home and vehicles. Devise a program that creates an abstract class named Loansand subclasses to display different types of loans and cost per month ( home, car, and so on..) Use constructors in each of the classes with appropriate arguments. Include get and set methods and an abstract method calculateLoan(). Prompt the user for the type to be displayed, and then create the appropriate object. Also create an interface with one method printDetails() that you use with subclasses.

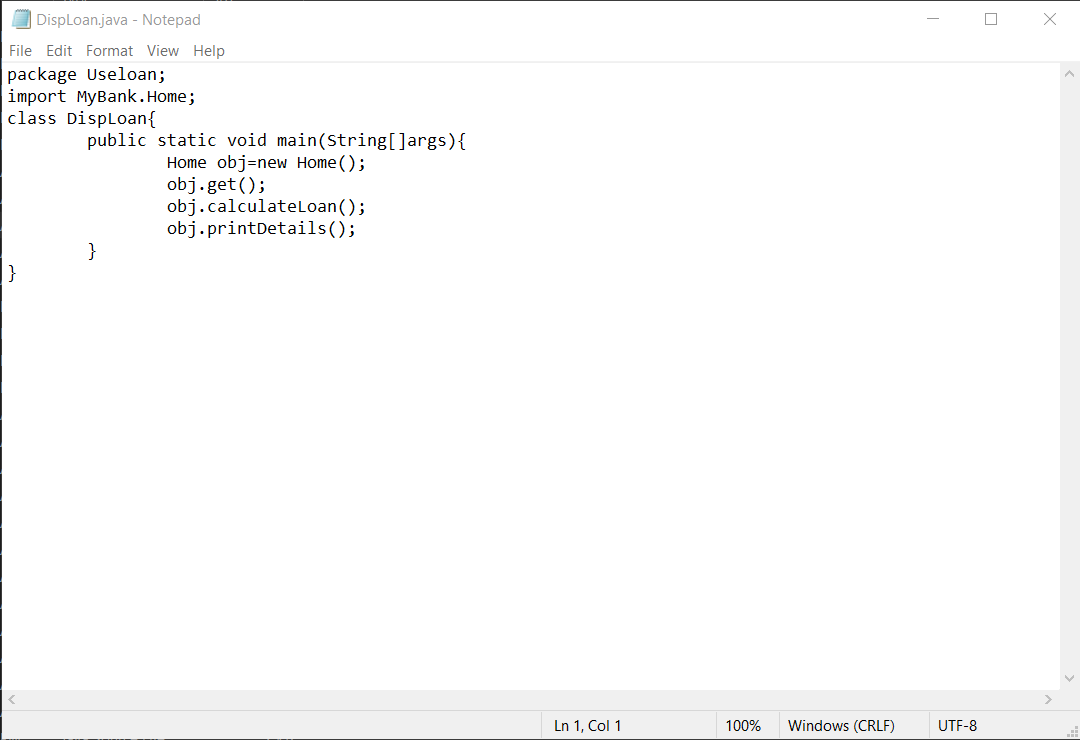
Save the Loan.java, Home.java, Car.java and Print.java in MyBank package and the testing program in UseLoan package.

**Code:**

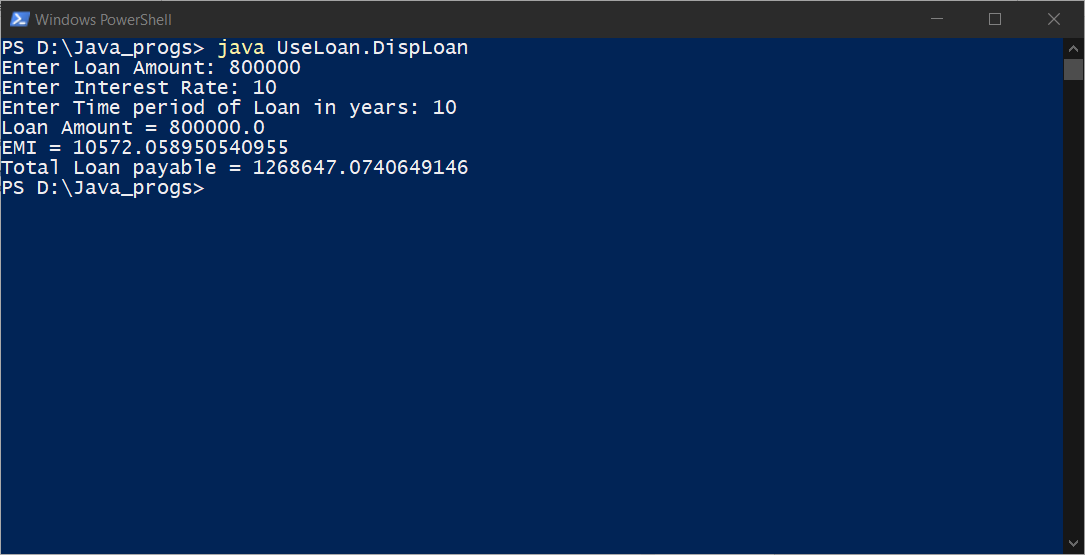
****

****

****

****

**OUTPUT:**

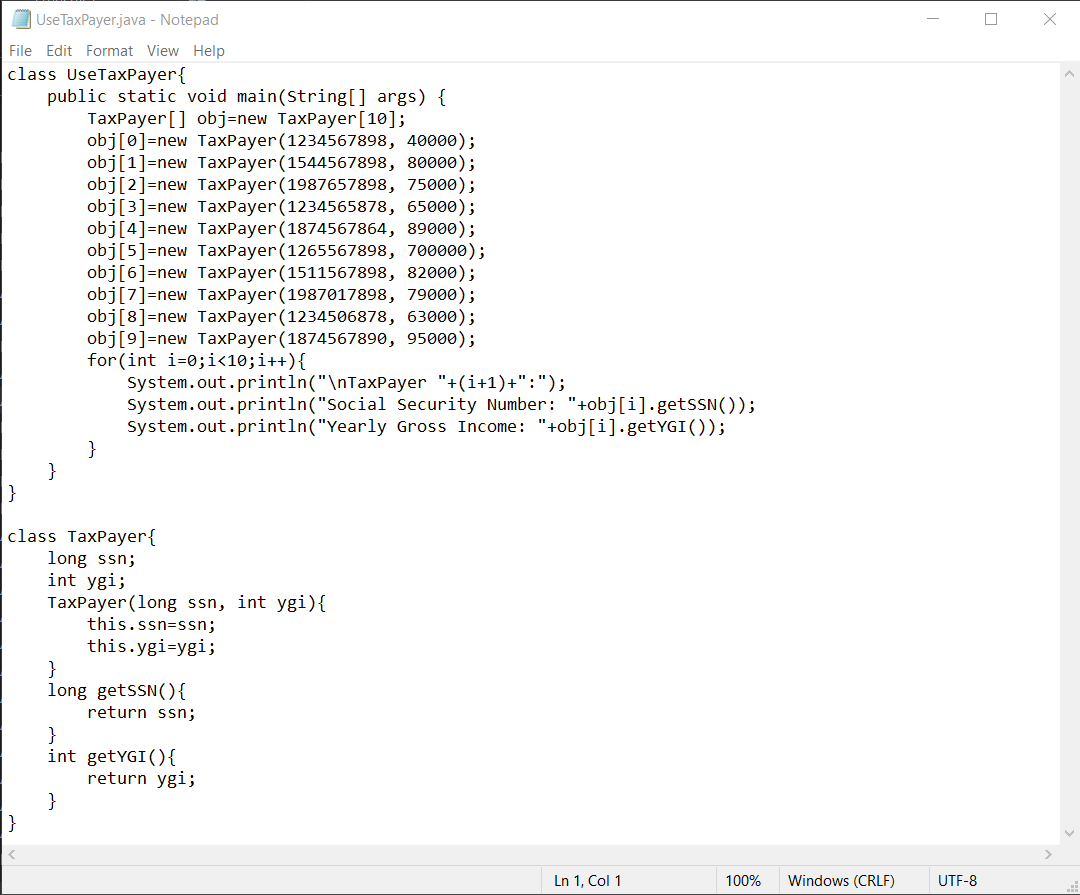
****

**2)**

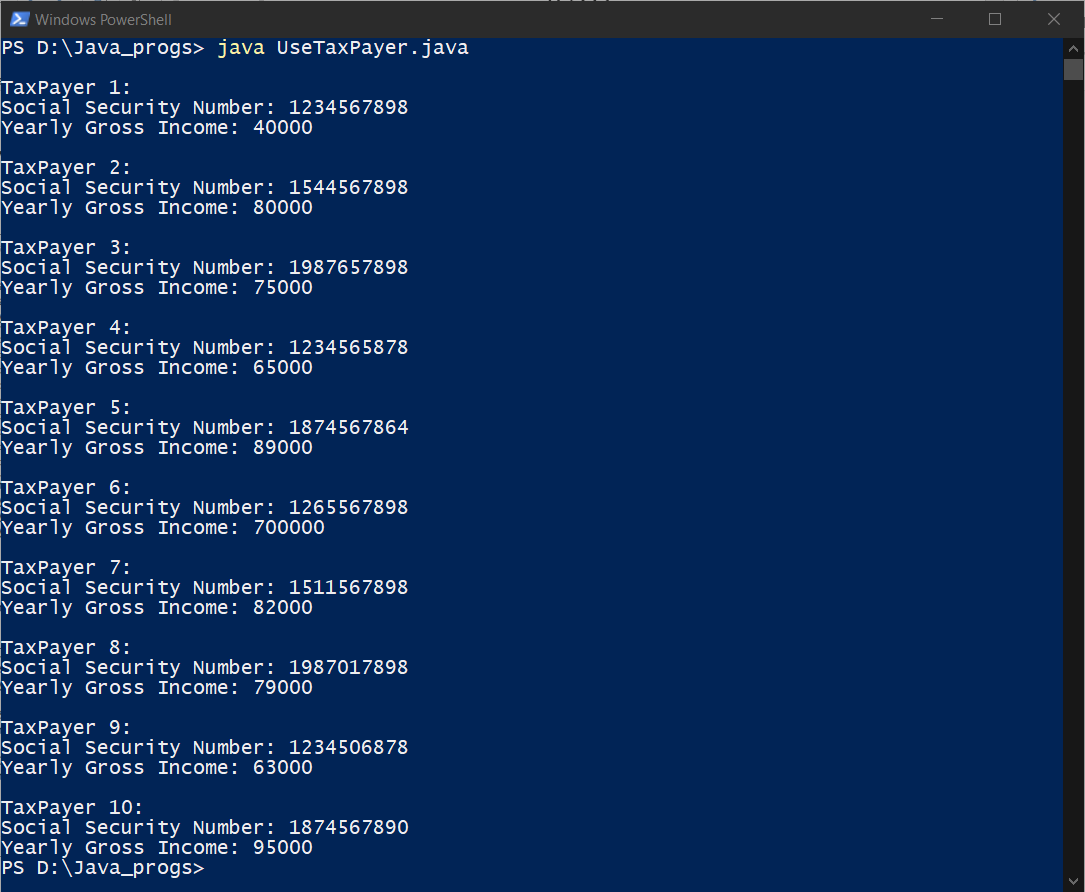
Create a class named TaxPayer. Data fields for TaxPayer include Social Security number (type long) and yearly gross income. Methods include a constructor that requires values for both data fields and two get methods that returns each of the data field values. The gross income should be in the range from $10000 to $100,000.

Write a JAVA program named UseTaxPayer that declares an array of 10 Taxpayer objects and display the 10 Taxpayer objects.

**Code:**

****

**OUTPUT:**

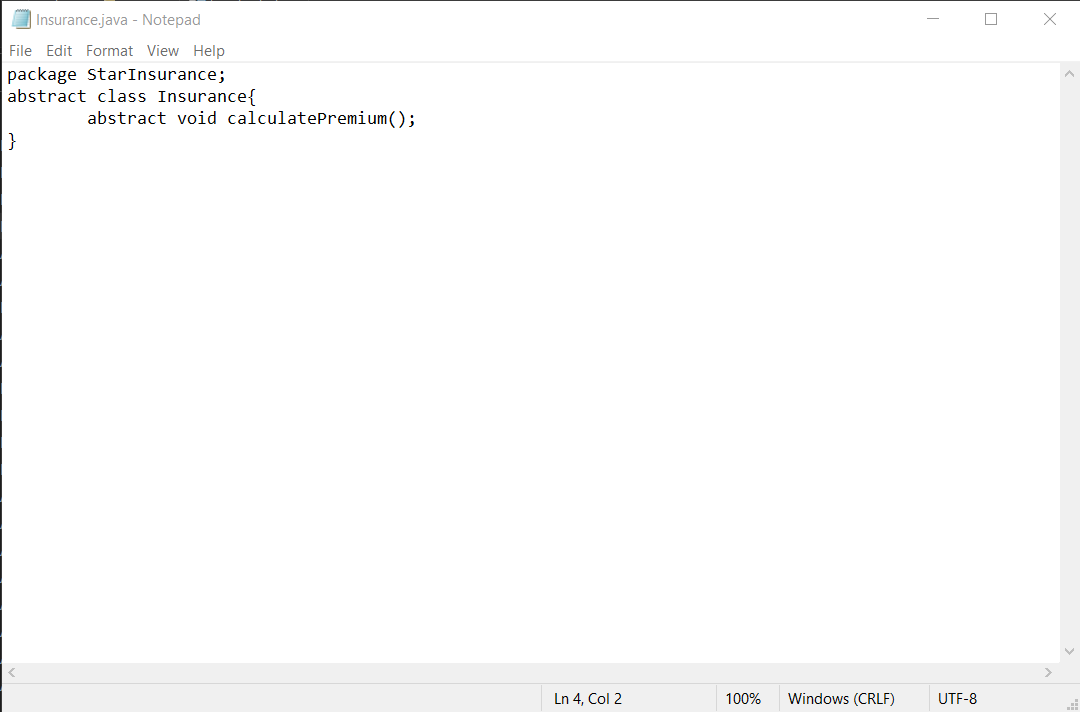


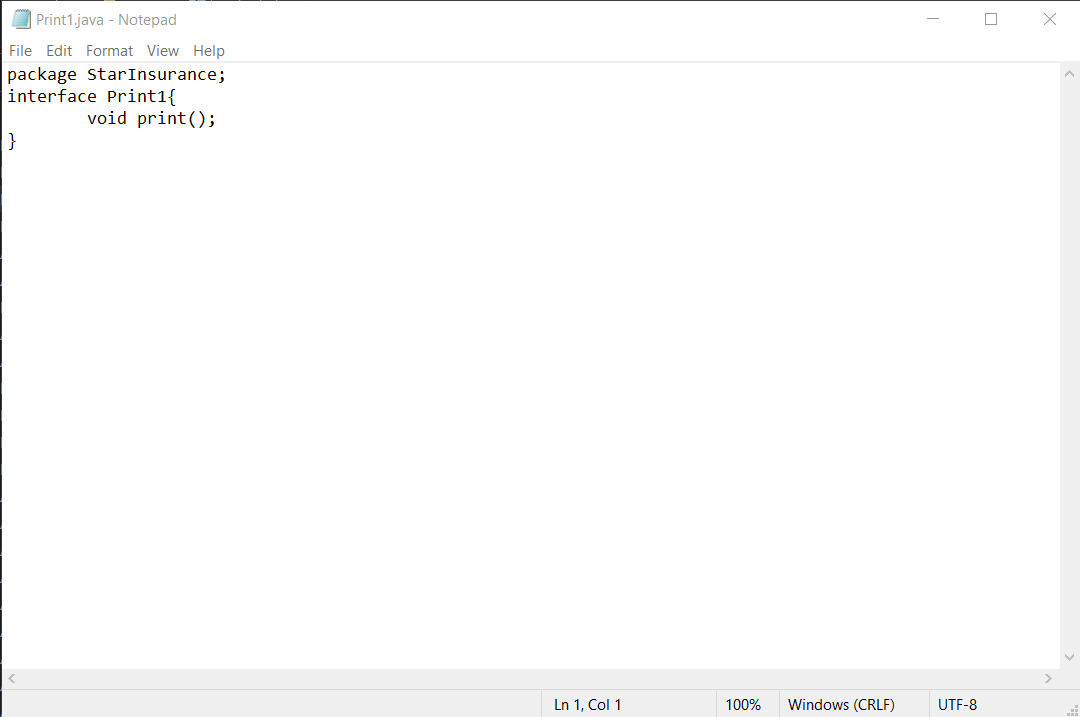
**3)**

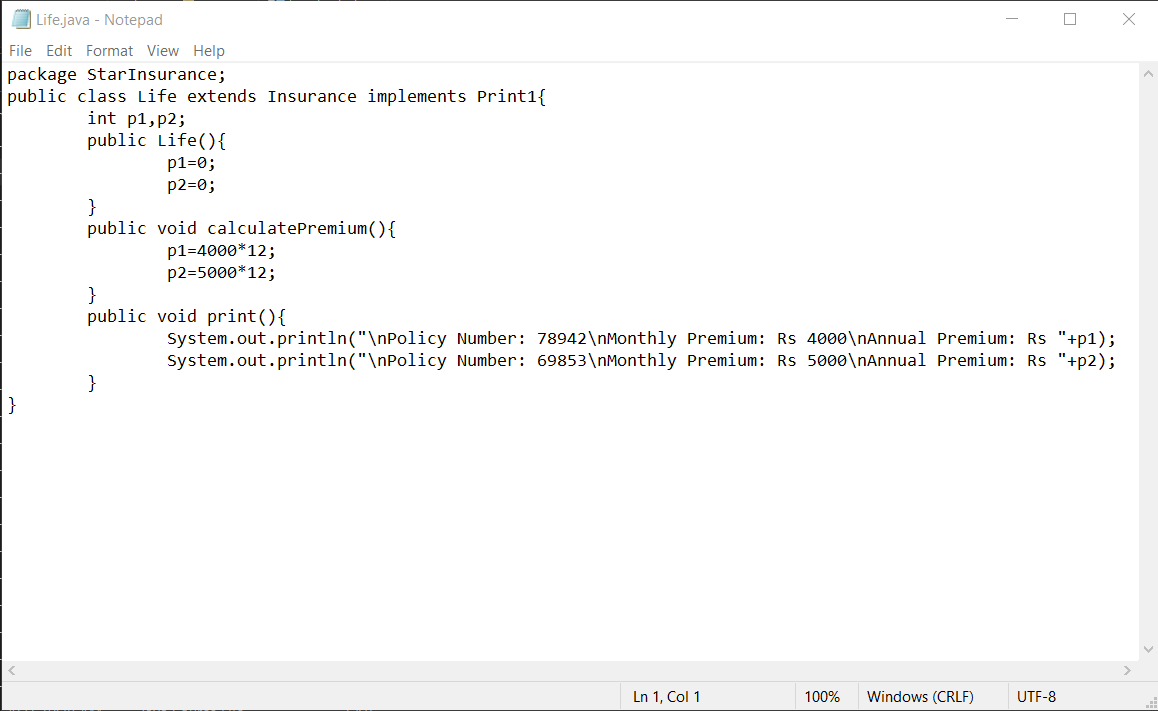
Star Insurance Company is offering different policies. You are asked to write a program that uses an abstract class namedInsurance and Health and Life subclasses to display different types of insurance policies and the cost per month. Use constructors in each class, with appropriate arguments. Include get and set methods, and an abstract method calculatePremium(). Prompt the user for the type to be displayed, and then create the appropriate object. Also create an interface for a print() method and use this interface with both subclasses.

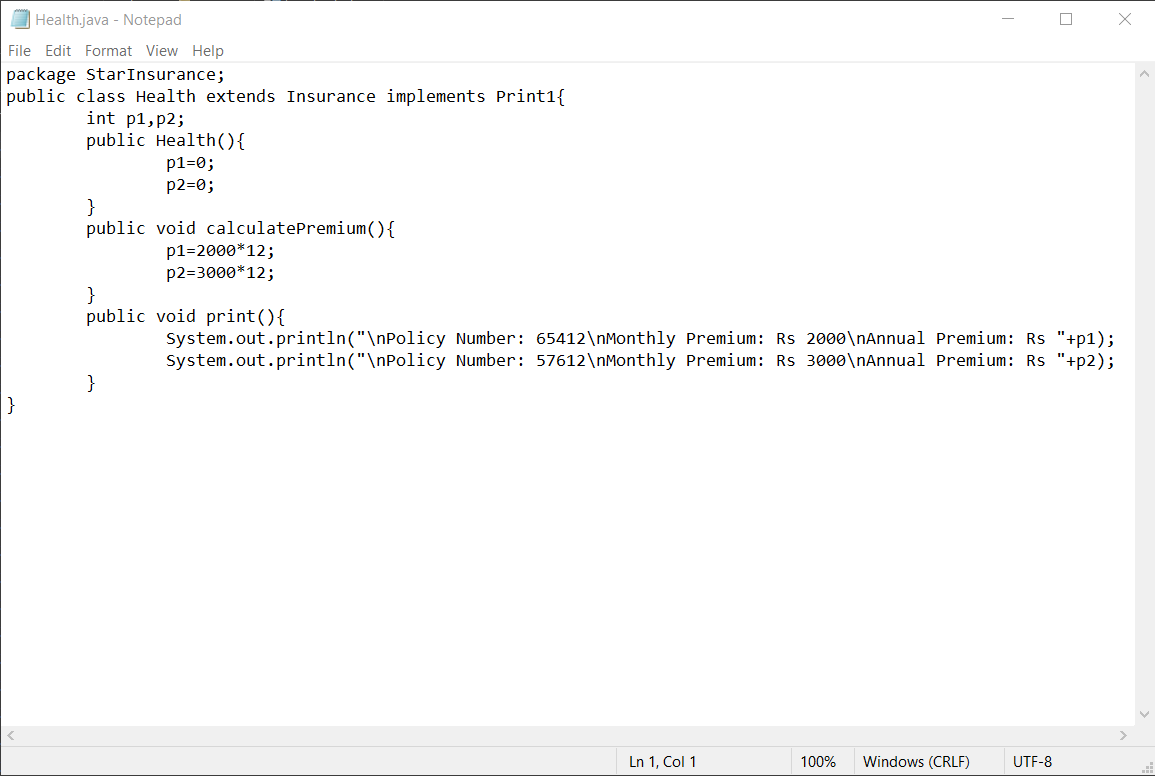
Save the Insurance.java, Health.java, Life.java and Print.java in StarInsurance package and the testing program in defaultpackage.

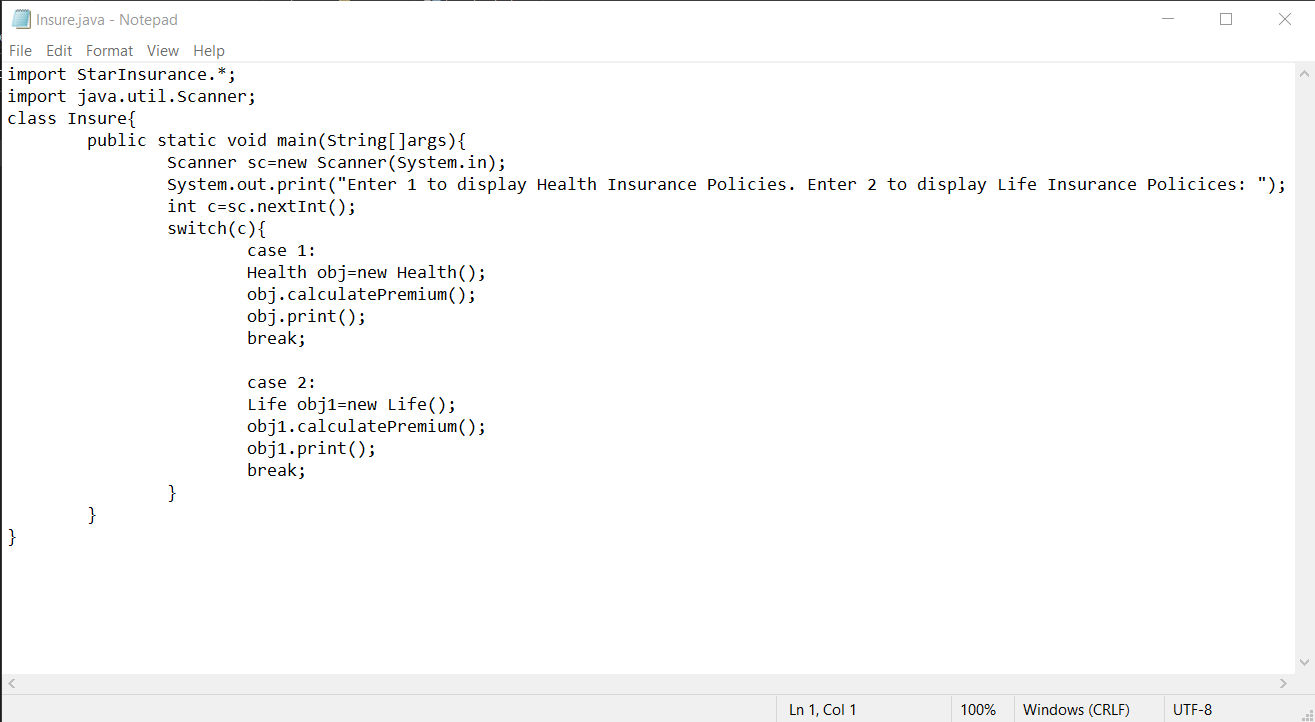
**Code:**

****

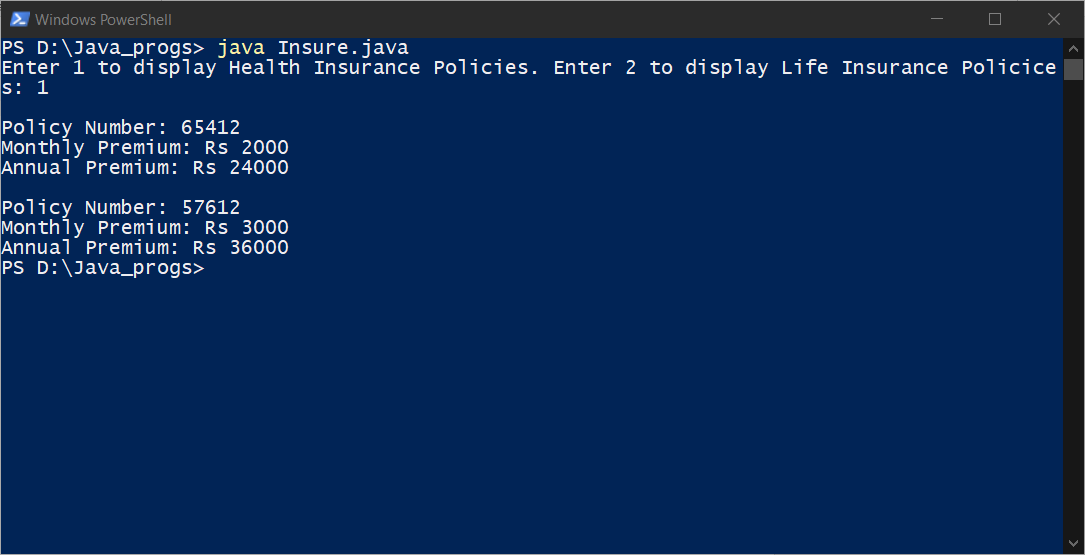
****

****

****

****

**OUTPUT:**

****