// Class: a collection of related variables and functions

/\*

public class Person {

public int age;

public String name;

public String address;

public Person( int newAge, String newName, String newAddress ) {

this.age = newAge;

this.name = newName;

this.address = newAddress;

}

}

public class Test {

public static void main( String[] args ) {

// To declare a variable, the syntax is:

// [data type] [variable name];

// int x;

Person person = new Person( 10, "Kevin", "1234 Sesame Street" );

}

}

\*/

// TODO: Make a program to keep track of customers and purchases for a retail store.

// Make a class that represents products that a Customer can purchase. Keep track of the product's name,

// and price. Keep track of the customer's name, email address, and address.

// Make 3 customers (Customer instances) and make 3 products.

public class Product {

public double price;

public String name;

public Product ( double newPrice, String newName ) {

this.price = newPrice;

this.name = newName;

}

}

public class Customer {

public String name;

public String emailAddress;

public String address;

public Customer ( String newName, String newEmailAddress, String newAddress ) {

this.name = newName;

this.emailAddress = newEmailAddress;

this.address = newAddress;

}

}

public class Test {

public static void main( String[] args ) {

Product product = new Product ("NaturalKiss", 25.50);

Customer customer = new Customer ("Joe", "joe@gmail.com", "1234 Oh");

System.out.println( "The customer " + customer.name + " is purchasing one " + product.name + "." );

}

}

// Abstraction: using "private" on any properties or methods of a class that you don't need outside the class

// Access Modifier: a keyword (public or private) that you use on class properties and methods to control whether or not

// they're accessible outside of this class.

public class Person {

public String name;

private int age;

private String address;

public double salary;

public Person( String newName, int newAge, String newAddress, double newSalary ) {

this.name = newName;

this.age = newAge;

this.address = newAddress;

this.salary = newSalary;

}

}

public class Test {

public static void Main( String[] args ){

Person person = new Person( "Joe", 10, "1234 Sesame Street" );

System.out.println( person.name + " makes " + person.salary + " a year." );

}

}

public class Product {

private double price;

public String name;

private double taxRate;

public Product( double newPrice, String newName ) {

this.price = newPrice;

this.name = newName;

this.taxRate = 0.7;

}

public double totalPrice( ) {

return product.price + product.price \* product.taxRate;

}

}

public class Test {

public static void Main( String[] args ){

Product product = new Product ("NaturalKiss", 25.50);

// TODO: Output the total price (after tax) of the product

System.out.println( "Total price: " + product.totalPrice( ) );

}

}

public class Dog extends Animal {

public String breed;

public String name;

public Dog( String newBreed, String newName, double newWeight, String newColor ){

this.breed = newBreed;

this.name = newName;

this.weight = newWeight;

this.color = newColor;

}

}

public class Snake extends Animal {

public double bodyLength;

public boolean isVenomous;

// TODO: Make a constructor

}

public class Test {

public static void main( String[] args ){

Dog dog = new Dog( "Poodle", "Coco", 50, "White" );

Snake snake = new Snake( );

}

}