Chapter 3 Problem Set A

1. Car Class

Write a class named Car that has the following fields:

- yearModel. The yearModel field is an int that holds the car's year model.
- make. The make field is a String object that holds the make of the car.
- speed. The speed field is an int that holds the car's current speed.

In addition, the class should have the following methods.

- Constructor. The constructor should accept the car's year model and make as arguments.

  These values should be assigned to the object's yearModel and make fields. The constructor should also assign 0 to the speed field.

- Accessor. The appropriate accessor methods get the values stored in an object's yearModel, make, and speed fields.
- accelerate. The accelerate method should add 5 to the speed field each time it is called.
- brake. The brake method should subtract 5 from the speed field each time it is called.

Demonstrate the class in a program that creates a Car object, and then calls the accelerate method five times. After each call to the accelerate method, get the current speed of the car and display it. Then, call the brake method five times. After each call to the brake method, get the current speed of the car and display it.

2. Personal Information Class

Design a class that holds the following personal data: name, address, age, and phone number. Write appropriate accessor and mutator methods. Demonstrate the class by writing a program that creates three instances of it. One instance should hold your information, and the other two should hold your friends' or family members' information.

3. Retail Item Class

Write a class named RetailItem that holds data about an item in a retail store. The class should have the following fields:

- description. The description field is a String object that holds a brief description of the item.
- unitsOnHand. The unitsOnHand field is an int variable that holds the number of units currently in inventory.
- price. The price field is a double that holds the item's retail price.

Write appropriate mutator methods that store values in these fields and accessor methods that return the values in these fields. Once you have written the class, write a separate program that creates three RetailItem objects and stores the following data in them.

	Description	**Units On Hand**	**Price**
Item #1	Jacket	12	59.95
Item #2	Designer Jeans	40	34.95
Item #3	Shirt	20	24.95

4. Pet Class

Design a class named Pet., which should have the following fields:

- name. The name field holds the name of a pet.
- type . The type field holds the type of animal that a pet is. Example values are “Dog", “Cat", and ”Bird".
- age. The age field holds the pet's age:.

The Pet class should also have the following methods:

- setName   The setName method stores a value in the name field.
- setType".   The setType method stores a value in the type field.
- setAge".   The setAge method stores a value in the age field.
- getName.   The getName method returns the value of the name field.
- getType.   The getType method n:rurns the value of the type field.
- getAge".   The getAge method returns the value of the age field.

Once you have designed the class, design a program that creates an object of the class and prompts the user to enter the name, type, and age of his or her pet. This data should be stored in the object. Use the object's accessor methods to retrieve the pet's name, type, and age and display this data on the screen.