1. Suppose you have a class `Cat` defined as shown below. Fill in the code for the missing methods, following guidelines given through comments.

```java
public class Cat {
    // instance variables
    private String name;
    private int age;
    private int lives; // number of lives remaining

    // constructor: takes a String as parameter and creates a
    // new Cat object with given String as the name.
    // all cats start out at age 0 and with 9 lives.

    // birthday(): increases the cat’s age by 1

    // death(): For cats that have at least one life remaining,
    // decreases the number of lives by 1; no effect
    // on dead cats (i.e., cats with 0 lives).
}
```
2) Suppose you are writing a **driver class** that uses the **Cat** class. Answer the questions below by writing code fragments for this driver class.

a) Write some code to instantiate an object of the **Cat** class with name “Luca” and assign it to a variable named `myProfsCat`.

```java
// Example code
Cat myProfsCat = new Cat("Luca");
```

b) Suppose you have already declared and initialized three variables `cat1`, `cat2`, and `cat3` that refer to **Cat** objects. Write some code that uses the `birthday()` and `death()` methods to increase the ages of each of all the cats (by one year each), and to make `cat3` lose one life.

```java
// Example code
Cat cat1 = /* initial state */;
Cat cat2 = /* initial state */;
Cat cat3 = /* initial state */;

cat1.birthday();
cat2.birthday();
cat3.birthday();
cat3.death();
```

c) Suppose that the **Cat** class also has accessor and mutator methods for `lives` (you do NOT need to write the code for these methods). Write some code to print the sum total of the lives of `cat1`, `cat2`, and `cat3`.

```java
// Example code
int totalLives = cat1.getLives() + cat2.getLives() + cat3.getLives();
System.out.println("Total lives: "+ totalLives);
```
1. Suppose you have a class `Car` defined as shown below. Fill in the code for the missing methods, following guidelines given through comments.

```java
public class Car {
    // instance variables
    String manufacturer;
    String model;
    int year;
    double price;

    // constructor
    public Car(String x, String y, int z, double w) {
        manufacturer = x;
        model = y;
        year = z;
        price = w;
    }

    // constructor: another version of the constructor, without
    // parameters for year, or price; sets these to default
    // values (2013 and 0, respectively).

    // price mutator

    // price accessor
}
```
2) Suppose you are writing a **driver class** that uses the **Car** class. Answer the questions below by writing code fragments for this driver class.

a) Write some code to instantiate an object of the **Car** class (using the second constructor) and to assign this object to a variable named `obscureObjectOfDesire`. Use this information for the car: manufactured by Jaguar, model name “XF Supercharged.”

b) Suppose you have already declared and initialized three variables `car1`, `car2`, and `car3` that refer to **Car** objects. Write some code to find the average of the car prices and assign it into a variable named `priceAvg`.

e) Write some code to reduce the price of `car2` by $5000. (So, for example, if the current price is $30000, it should be reduced to $25000.) Use the accessor and mutator methods.
1. Suppose you have a class `Cat` defined as shown below. Fill in the code for the missing methods, following guidelines given through comments.

```java
public class Cat {
    // instance variables
    private String name;
    private int age;
    private int lives; // number of lives remaining

    // constructor
    public Cat(String x, int y) {
        name = x;
        age = y;
        lives = 9; // start with 9 lives
    }

    // age accessor

    // age mutator

    // toString()
}
```
2) Suppose you are writing a **driver class** that uses the **Cat** class. Answer the questions below by writing code fragments for this driver class.

a) Write some code to instantiate an object of the **Cat** class, named “Luca”, aged 5, and assign it to a variable named **myProfsCat**

b) Write some code to print the information about **myProfsCat**

c) Suppose you have already declared and initialized three variables **cat1**, **cat2**, and **cat3** that refer to **Cat** objects. Write some code to calculate and print the average of the ages of **cat1**, **cat2**, and **cat3**.
1. Suppose you have a class `Car` defined as shown below. Fill in the code for the missing methods, following guidelines given through comments.

```java
public class Car {
    // instance variables
    String manufacturer;
    String model;
    int year;
    double price;

    // constructor
    public Car(String x, String y, int z, double w) {
        manufacturer = x;
        model = y;
        year = z;
        price = w;
    }

    // price mutator

    // price accessor

    // toString()
}
```
2) Suppose you are writing a **driver class** that uses the **Car** class. Answer the questions below by writing code fragments for this driver class.

a) Write some code to instantiate an object of the **Car** class and to assign this object to a variable named `obscureObjectOfDesire`. Use this information for the car: manufactured by Jaguar, model name “XF Supercharged,” year 2013, and price $68,100.

b) Write some code to print the information about `obscureObjectOfDesire`

c) Suppose you have already declared and initialized three variables `car1`, `car2`, and `car3` that refer to **Car** objects. Write some code to calculate and print the average price of these cars.

d) Write some code to set the price of `car1` to $25,000.
1. Suppose you have a class `Cat` defined as shown below. Fill in the code for the missing methods, following guidelines given through comments.

```java
public class Cat {
    // instance variables
    private String name;
    private int age;
    private int lives; // number of lives remaining

    // constructor
    public Cat(String x) {
        name = x;
        age = 0;
        lives = 9; // start with 9 lives
    }

    // birthday(): increases the cat’s age by 1

    // kill(): For cats that have at least one life remaining,
    // decreases the number of lives by 1; no effect
    // on dead cats (i.e., cats with 0 lives).

    // alive() returns true if cat still has at least one life
}
```
2) Suppose you are writing a **driver class** that uses the **Car** class. Answer the questions below by writing code fragments for this driver class.

a) Write some code to instantiate an object of the **Cat** class with name “Luca” and assign it to a variable named `myProfsCat`

b) Suppose the class also has a `toString()` method implemented (you do NOT need to write the code for this method). Write some code to print the information about `myProfsCat`

c) Repeat (b), but only print the information if the cat is still alive (i.e., use the `alive()` method); do not print anything if the cat is dead.

c) Suppose you have three variables `cat1`, `cat2`, and `cat3` that refer to **Cat** objects. Write some code that uses the `birthday()` and `kill()` methods to increase the ages of each of the cats, and to make `cat3` lose one life.