1. Fill in some code for a `Cat` class, 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, int z) {
        name = x;
        age = y;
        lives = z;
    }

    // getAge(): Returns the age of the cat
    // 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).
}
```

2) Write client code that uses the `Cat` class:

a) Instantiate a `Cat` object with name “Luca,” age 11, with 9 lives and assign it to a variable named `myCat`

b) Suppose you have two cat objects `cat1` and `cat2`. Print the average age of these cats.
1. Fill in some code for a `Cat` class, 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).

    // getLives(): Returns number of lives remaining for this cat
}
```

2) Write client code that uses the `Cat` class:

a) Instantiate a `Cat` object with name “Tuna” and assign it to a variable named `myCat`

b) Suppose you have two cat objects `cat1` and `cat2`. Print the total number of lives of these cats.