1. Fill in some code for an Employee class, following guidelines given through comments.

```java
public class Employee {
    // instance variables
    String name;
    String position;
    double hourly;  // hourly wages
    double hours;  // hours worked this week

    // constructor
    public Employee(String x, String y, double w, double h) {
        name = x;
        position = y;
        hourly = w;
        hours = h;
    }

    // wages(): Returns the wages of this employee for this week
    // calculated as 1.5 of hourly rate for hours over 40.
    public double wages() {
        double w;
        if (hours <= 40) w = hourly * hours;
        else w = 40 * hourly + (hours - 40) * 1.5 * hourly;
        return w;
    }
}
```

2) Write client code that uses the Employee class:

a) Instantiate an Employee object with name “Lucia Rodriguez” position “software engineer”, with hourly rate $42.50 and who worked 52.5 hours last week. Assign it to a variable named coderBoss.

```java
Employee coderBoss = new Employee("Lucia Rodriguez",
                                     "Software engineer",
                                     42.50,
                                     52.5);
```

b) Suppose you have three Employee objects e1, e2 and e3. Write some client code that uses the Employee class to calculate and print the average of their wages. (Note: it is NOT necessary to format as currency).

```java
System.out.println((e1.wages() + e2.wages() + e3.wages()) / 3);
```
1. Fill in some code for an Employee class, following guidelines given through comments.

```java
public class Employee {
    // instance variables
    String firstName;
    String lastName;
    int yearHired;

    // constructor: Construct object with x, y, and z as first
    //              name, last name, and year hired, respectively.
    public Employee(String x, String y, int z) {
        firstName = x;
        lastName = y;
        yearHired = z;
    }

    // toString(): Returns a String corresponding to object.
    public String toString() {
        return (firstName + " " + lastName + ", year hired" + year);
    }

    // getYearHired(): Returns the year this Employee was hired.
    public int getYearHired() {
        return yearHired;
    }
}
```

2) Write client code that uses the Employee class:

a) Instantiate an Employee object for someone named Miranda Hopkins, hired in 2012. Assign it to a variable named coder:

```java
Employee coder = new Employee("Miranda", "Hopkins", 2012);
```

b) Suppose you have two Employee objects e1, e2. Write some client code that uses the getYearHired() and toString() methods of the Employee class to print the information associated with the Employee who has been with the company the longest (i.e., was hired earlier).

```java
if (e1.getYearHired() < e2.getYearHired())
    System.out.println(e1);
else
    System.out.println(e2);
```
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
    public void birthday() {
        age++;
    }

    // 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).
    public void kill() {
        if (lives > 0) {
            lives--;
        }
    }

    // getLives(): Returns number of lives remaining for this cat
    public int getLives() {
        return lives;
    }
}
```

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`
      ```java
      Cat myCat = new Cat("Tuna");
      ```
   b) Suppose you have two cat objects `cat1` and `cat2`. Print the total number of lives of these cats.
      ```java
      System.out.println(cat1.getLives() + cat2.getLives());
      ```