Practice with executing sequences of statements

1. Below, to the right, is a definition of class Student, and below are two initializing declarations and an assignment.

   Student p1 = new Student(356, 1987);
   Student p2 = new Student(123, 1988);
   p1 = p2;

(a) Below, draw the variables that are declared in the statements given above.

(b) Using the variables you drew in part (a), execute the sequence of three statements, drawing any objects created during execution and assigning to variables as required by execution.

/\** An instance represents a Student */
\public class Student {
   \// the student's ID
   \private int id;
   \// the student's year of birth
   \private int birthYear;
   /\** Constructor: a student with id d and birth year y */
   \public Student(int d, int y) {
      id = d; birthYear = y;
   }
}

2. Below, to the right, is a definition of class Employee, and below are two initializing declarations and an assignment.

   Employee e1 = new Employee(123, 50000);
   Employee e2 = new Employee(456, 45000);
   e1 = e2;

(a) Below, draw the variables that are declared in the statements given above.

(b) Using the variables you drew in part (a), execute the sequence of three statements, drawing any objects created during execution and assigning to variables as required by execution.

/\** An instance represents an Employee */
\public class Employee {
   \// the employee's ssn
   \private int ssn;
   \// the employee's annual salary
   \private int salary;
   /\** Constructor: an employee with ssn n and salary s */
   \public Employee(int n, int s) {
      ssn = n; salary = s;
   }
}
3. Below, to the right, is a definition of class Prof, and below are two initializing declarations and an assignment.

```java
Prof px = new Prof(101, 45);
Prof py = new Prof(110, 52);
px = py;
```

(a) Below, draw the variables that are declared in the statements given above.

(b) Using the variables you drew in part (a), execute the sequence of three statements, drawing any objects created during execution and assigning to variables as required by execution.

4. Draw the static components of Celeb somewhere below (do not draw file drawers)—you do not have to draw the method body.

Next, evaluate first the leftmost new-expression below and then the rightmost one, assuming that these are the first new-expressions evaluated. Under each new-expression, draw the object that results from evaluation of that new-expression; above each new-expression, write its value. Include the partition for class Object, and put in two methods that you know are declared in Object.

Be sure to fill in the values of fields correctly, based on the specifications of the methods in the classes.

```java
/** An instance maintains info about a celebrity */
public class Celeb {
    private String name; // celebrity’s name
    /* celebrity's no. in the list of celebrities */
    private int celebNum;
    // total number of celebrities
    private static int numCelebs = 0;
    /** Constructor: a celebrity with name n. */
    public Celeb(String n) {
        name = n;
        celebNum = numCelebs;
        numCelebs = numCelebs + 1;
    }
}

/** An instance maintains info about an athlete */
public class Ath extends Celeb {
    private String sport; // The athlete's sport
    private int wins; // # of wins
    private int losses; // # of losses
    private double salary; // The athlete's salary
    /** Constructor: a new athlete with name n, playing sport sport. 
     * The athlete has no wins, losses, or salary. */
    public Ath(String n, String sport) {
        super(n);
        this.sport = sport;
    }
}