# Exercises: Classes

## Code Writing

**1**. Write a class called `Player` that keeps track of a generic player on some sports team. This class should store a player's first and last name as two separate values, the name of the team they play for, and their jersey number. Write the constructor method for this class, as well as the getter and setter methods for **only** the jersey number; your setter should ensure the jersey number is always greater than or equal to 0. Also write a method called `getPlayerInfo()` that has no parameters and returns a `String` value showing the player's name, team, and jersey number. For example, if the player's first and last name is `Bryce Harper`, his number `34`, and his team `Washington Nationals`, then `getPlayerInfo()` should return `Bryce Harper (Washington Nationals, #34)`.

**Solution:**

```
public class Player {

    private String firstName;
    private String lastName;
    private String team;
    private int jerseyNum;

    public Player(String fn, String ln, String t, int jn) {
        firstName = fn;
        lastName = ln;
        team = t;
        jerseyNum = jn;
    }

    public int getJerseyNum() {
        return jerseyNum;
    }

    public void setJerseyNum(int newNum) {
        if(newNum >= 0) {
            jerseyNum = newNum;
        }
    }

    public String getPlayerInfo() {
        return firstName + " " + lastName + " (" + team + ", #" + jerseyNum +
   ")");
    }

}
```

**2**. Write a new class called Email. Each object of this type will represent a single email. Emails are defined by who sent the email, who is receiving the email (just String names; note that this can be more than one person), the subject of the email, and the body of the email. Write the getter/setter methods for the subject (don't worry about the other attributes). The additional method for this class should print out the entire email nicely formatted (i.e., sender, receiver(s), subject, body).

**Solution:**

```
public class Email {

    private String sender;
    private String[] receivers;
    private String subject;
    private String body;

    public Email(String sender, String[] receivers, String subject, int body)
   {
        this.sender = sender;
        this.receivers = receivers;
        this.subject = subject;
        this.body = body;
    }

    public void printEmail() {
        System.out.println("Sent by: " + sender);
        System.out.println("To: ");

        for(int i = 0; i < receivers.length; i++) {
            if(receivers[i] != null) {
                System.out.println("\\t" + receivers[i]);
            }
        }

        System.out.println("Subject: " + subject);
        System.out.println(body);

    }

}
```

## Code Reading

Below is the `Ticket` class, which is used in the `TheCommons` programs found on the subsequent pages. What is displayed to the console after running the `TheCommons` program?

### `Ticket` Class

```
1  public class Ticket {
2      private int seat;
3      private String row;
4      private String graffiti;
5
6      public Ticket(int s, String r) {
7          seat = s;
8          row = r;
9          graffiti = "";
10     }
11
12     public int getSeat() {
13         return seat;
14     }
15
16     public String getRow() {
17         return row;
18     }
19
20     public void addGraffiti(String g) {
21         graffiti = graffiti + g;
22     }
23
24     public String getGraffiti() {
25         return graffiti;
26     }
27 }
```

## Group 1

```
1  public class TheCommons01 {
2      public static void main(String[] args) {
3          Ticket tix1, tix2, tix3, tix4;
4          Ticket larry, curly, moe, shemp, floor;
5
6          tix1 = new Ticket(10, "A");
7          tix2 = new Ticket(15, "C");
8          tix3 = new Ticket(25, "B");
9          tix4 = new Ticket(16, "C");
10
11         larry = tix1;
12         larry.addGraffiti("L,");
13         curly = tix2;
14         curly.addGraffiti("C,");
15         moe   = tix3;
16         moe.addGraffiti("M,");
17         shemp = tix4;
18         shemp.addGraffiti("S,");
19
20         floor = moe;
21         moe = shemp;
22         moe.addGraffiti("M,");
23         shemp = floor;
24         shemp.addGraffiti("S,");
25
26         larry = curly;
27         larry.addGraffiti("L,");
28         curly = larry;
29         curly.addGraffiti("C,");
30
31         curly = tix1;
32         curly.addGraffiti("C,");
33
34         System.out.println("Larry's " + getTicketString(larry));
35         System.out.println("Curly's " + getTicketString(curly));
36         System.out.println("Moe's   " + getTicketString(moe));
37         System.out.println("Shemp's " + getTicketString(shemp));
38     }
39     public static String getTicketString(Ticket t) {
40         String s = "Ticket) " + t.getRow() + "-"
41                    + t.getSeat() + " " + t.getGraffiti();
42         return s;
43     }
44 }
```

**Solution:**

```
Larry's Ticket) C-15 C,L,C,
Curly's Ticket) A-10 L,C,
Moe's   Ticket) C-16 S,M,
Shemp's Ticket) B-25 M,S,
```

## Group 2

```
1  public class TheCommons02 {
2      public static void main(String[] args) {
3          Ticket tix1, tix2, tix3, tix4;
4          Ticket larry, curly, moe, shemp, floor;
5
6          tix1 = new Ticket(10, "A");
7          tix2 = new Ticket(15, "C");
8          tix3 = new Ticket(25, "B");
9          tix4 = new Ticket(16, "C");
10
11         larry = tix1;
12         larry.addGraffiti("L,");
13         curly = tix2;
14         curly.addGraffiti("C,");
15         moe   = tix3;
16         moe.addGraffiti("M,");
17         shemp = tix4;
18         shemp.addGraffiti("S,");
19
20         larry = moe;
21         larry.addGraffiti("L,");
22         moe = larry;
23         moe.addGraffiti("C,");
24
25         floor = curly;
26         curly = shemp;
27         curly.addGraffiti("C,");
28         shemp = floor;
29         shemp.addGraffiti("S,");
30
31         moe = tix1;
32         moe.addGraffiti("M,");
33
34         System.out.println("Larry's " + getTicketString(larry));
35         System.out.println("Curly's " + getTicketString(curly));
36         System.out.println("Moe's   " + getTicketString(moe));
37         System.out.println("Shemp's " + getTicketString(shemp));
38     }
39     public static String getTicketString(Ticket t) {
40         String s = "Ticket) " + t.getRow() + "-"
41                    + t.getSeat() + " " + t.getGraffiti();
42         return s;
43     }
44 }
```

**Solution:**

```
Larry's Ticket) B-25 M,L,C,
Curly's Ticket) C-16 S,C,
Moe's   Ticket) A-10 L,M,
Shemp's Ticket) C-15 C,S,
```

## Group 3

```
1  public class TheCommons03 {
2      public static void main(String[] args) {
3          Ticket tix1, tix2, tix3, tix4;
4          Ticket larry, curly, moe, shemp, floor;
5
6          tix1 = new Ticket(10, "A");
7          tix2 = new Ticket(15, "C");
8          tix3 = new Ticket(25, "B");
9          tix4 = new Ticket(16, "C");
10
11         larry = tix1;
12         larry.addGraffiti("L,");
13         curly = tix2;
14         curly.addGraffiti("C,");
15         moe   = tix3;
16         moe.addGraffiti("M,");
17         shemp = tix4;
18         shemp.addGraffiti("S,");
19
20         larry = shemp;
21         larry.addGraffiti("L,");
22         shemp = larry;
23         shemp.addGraffiti("S,");
24         floor = larry;
25         larry = curly;
26         larry.addGraffiti("L,");
27         curly = shemp;
28         curly.addGraffiti("C,");
29         shemp = moe;
30         shemp.addGraffiti("S,");
31         moe = floor;
32         moe.addGraffiti("M,");
33
34         System.out.println("Larry's " + getTicketString(larry));
35         System.out.println("Curly's " + getTicketString(curly));
36         System.out.println("Moe's   " + getTicketString(moe));
37         System.out.println("Shemp's " + getTicketString(shemp));
38     }
39     public static String getTicketString(Ticket t) {
40         String s = "Ticket) " + t.getRow() + "-"
41                    + t.getSeat() + " " + t.getGraffiti();
42         return s;
43     }
44 }
```

**Solution:**

```
Larry's Ticket) C-15 C,L,
Curly's Ticket) C-16 S,L,S,C,M,
Moe's   Ticket) C-16 S,L,S,C,M,
Shemp's Ticket) B-25 M,S,
```