

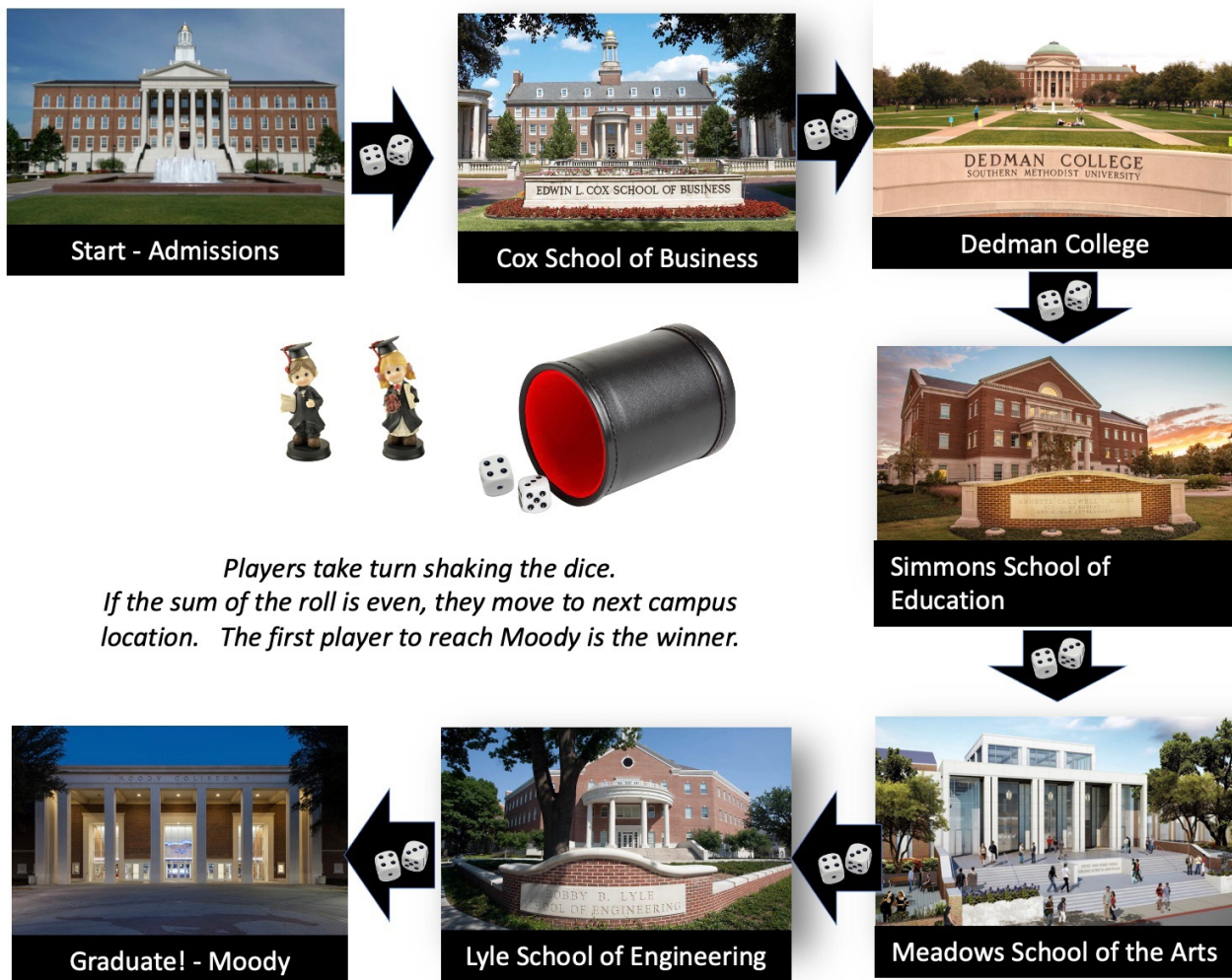
# CS 1341 - Lab 6 Assignment



## Graduate from SMU Game Object-Oriented Rewrite

### Lab (100 Points)

Create a modified version of the "Graduate from SMU" using the object-oriented design provided on the following pages. This lab is broken into five steps with point values assigned to each step. Complete and test each incremental solution before moving on to the next. Some code you'll need is provided, so read the instructions carefully.



**The Lab is due in Canvas Saturday April 9, 2022 at 6:00am.**

**Note:** Sharing these instructions with anyone other than an SMU CS Teaching Assistant or Senior Mentor, or ALEC tutor (or another tutor pre-approved by your instructor), posting on any website, or submission of any part of a solution that was created by anyone other than you constitutes an SMU honor code violation.

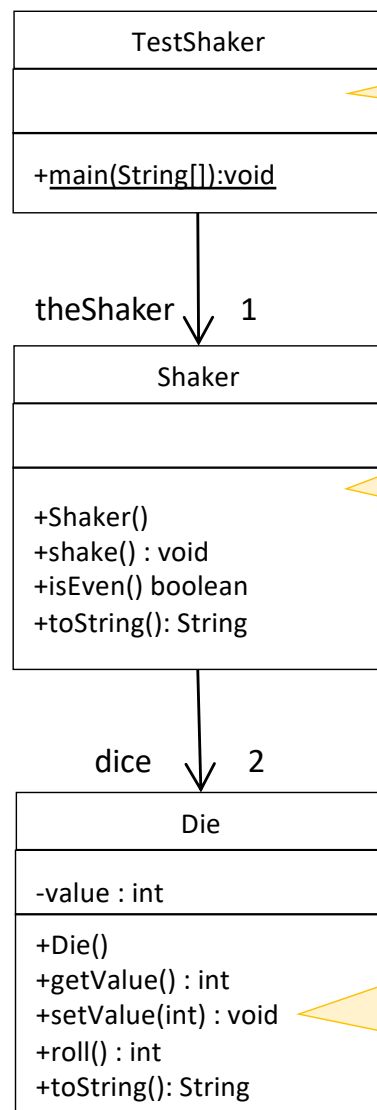
## Lab 6 - Part 1

[20 points]

Create the classes Shaker, Die, and TestShaker described below. Run TestShaker to get similar sample output (with different random values, of course):

### Sample output:

```
% java TestShaker
Shaker with Die 4 Die 3
Shaker with Die 4 Die 2
Sum is even!
Shaker with Die 3 Die 6
Shaker with Die 4 Die 6
Sum is even!
Shaker with Die 6 Die 3
Shaker with Die 3 Die 1
Sum is even!
Shaker with Die 5 Die 5
Sum is even!
Shaker with Die 4 Die 2
Sum is even!
Shaker with Die 4 Die 1
Shaker with Die 5 Die 4
```



#### In the main method:

```
Shaker theShaker = new Shaker();
for(int i = 1; i <= 10; i++)
{
    theShaker.shake();
    System.out.println(theShaker);
    if(theShaker.isEven() == true)
        System.out.println("Doubles!");
}
```

#### Reference instance variable

- `dice` is an array containing two `Die` objects

#### Constructor

- Create two `Die` objects and put in `dice` array

#### `shake()`

- Send roll to each `Die` in the array

#### `isEven()`

- Returns `true` if the sum of both `Die` objects is even, otherwise returns `false`

#### `toString()`

- Returns a `String` containing the values of both dice in the array
- Example: Shaker with Die 6 Die 4

#### Simple instance variable

- `value` contains an integer with the roll value

#### Constructor

- Initialize value to 1

#### `getValue()` and `setValue(int)`

- Getter and setter for value attribute

#### `roll()`

- Using `java.util.Random`, assign a random number in the range of 1..6 in the attribute value

#### `toString()`

- Returns a `String` containing label `Die` followed by the contents of the value attribute
- Example: Die 5

## Lab 6 - Part 2

[30 points]

Create the classes *Campus*, *Location*, and *TestCampus* described below. Run *TestCampus* to match the sample output.

### Sample output:

```
% java TestCampus
Starting location is Location #0: Admissions
Campus:
Location #0: Admissions
Location #1: Cox School of Business
Location #2: Dedman College
Location #3: Simmons School of Education
Location #4: Meadows School of the Arts
Location #5: Lyle School of Engineering
Location #6: Moody
```

#### In the main method:

```
Campus theCampus = new Campus();
System.out.println("Starting location is " + theCampus.getStartLocation());
System.out.println(theCampus);
```

#### Reference instance variable

- locations is an array to contain 7 instances of Location

#### Constructor

- Create the locations array
- Create 7 Location objects and put in each of the seven slots in the array. (Use name/numbers in sample output)

#### getLocation(int x)

- Using x as an index number in the locations array, return the Location object at that index #

#### getStartLocation()

- Returns the Location object at index # 0

#### toString()

- Returns a String containing the String representation of all seven locations in the array (hint: use a loop to build the String containing all Location objects in the array)

#### Simple instance variables

- name contains a String with the location name
- locationNumber has value 0,1,2,3,4,5 or 6

#### Constructor

- Two parameters contain the location name and number
- Set name and locationNumber to the passed in values

#### getValue() and setValue(int)

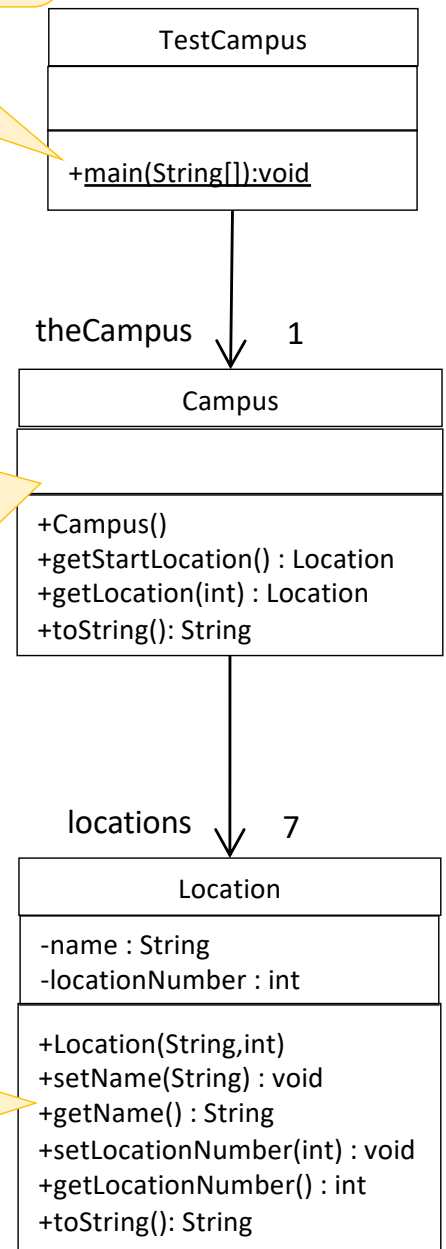
Getter and setter for name attribute

#### getLocationNumber() and setLocationNumber(int)

Getter and setter for locationNumber attribute

#### toString()

- Returns a String containing label Location # followed by locationNumber and name
- Example: Location #0: Admissions



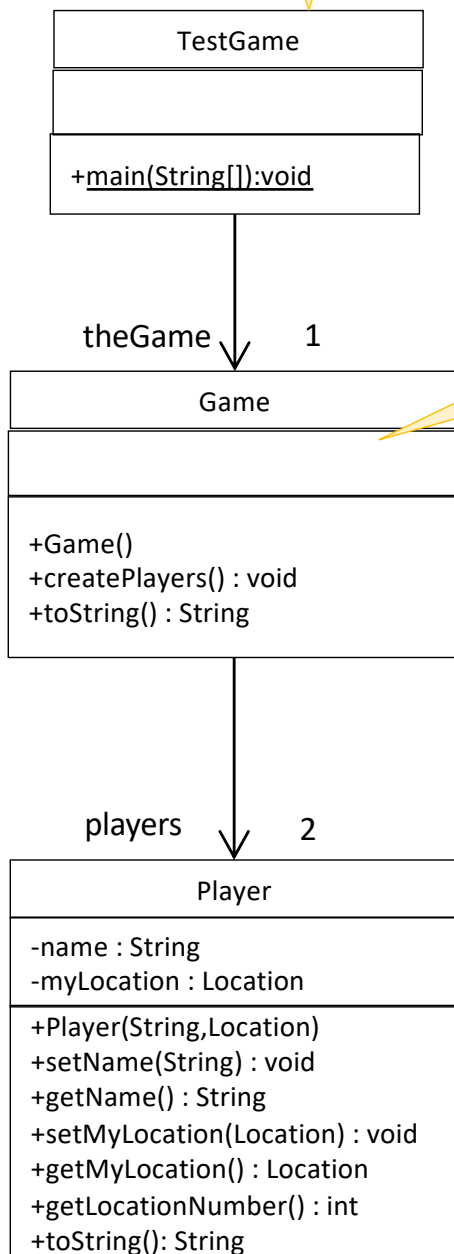
## Lab 6 - Part 3

[30 points]

Create the classes *Game*, *Player*, and *TestGame* described below. You will also use *Campus* that you created in Part 2 of this lab.

In the main method:

```
Game theGame = new Game();
System.out.println(theGame);
```



## Sample output:

```
% java TestGame
Enter player 1's name: Homer
Enter player 2's name: Marge
Graduation Game:
Homer is at Location #0: Admissions
Marge is at Location #0: Admissions
```

Reference instance variables

- `players` contains an array of *Player* objects
- `theCampus` contains a reference to a *Campus* object

Constructor

- Create a *Campus* object and assign to `theCampus` variable
- Call the `createPlayers` method

`createPlayers()`

- Create *Player* array of size 2 and assign to `players` variable
- Use `java.util.Scanner` to prompt for the two *Player* name Strings
- Send `getStartLocation` to `theCampus`, which will return the *Location* object that is the starting point of the game
- Create two *Player* objects using the name entered by the user and the *Location* object returned by `theCampus`. Pass both to the *Player* constructor
- Put each of the newly created *Player* objects in the `players` array

`toString()`

- Returns a String containing the String representation of both *Player* objects in the `players` array with their location

You already built *Campus* in Part 2

Simple instance variable

- `name` contains a String with the *Player* name

Reference instance variable

- `myLocation` contains a reference to a *Location* object passed into the constructor

Constructor

- Two parameters contain the player name and location
- Set name and `myLocation` to the passed in values

`getName()` and `setName(int)`

Getter and setter for `name` attribute

`getMyLocation()` and `setMyLocation(Location)`

Getter and setter for `myLocation` attribute

`toString()`

- Returns a String containing the player name and current
- Example: Pat is at Location #0: Admissions

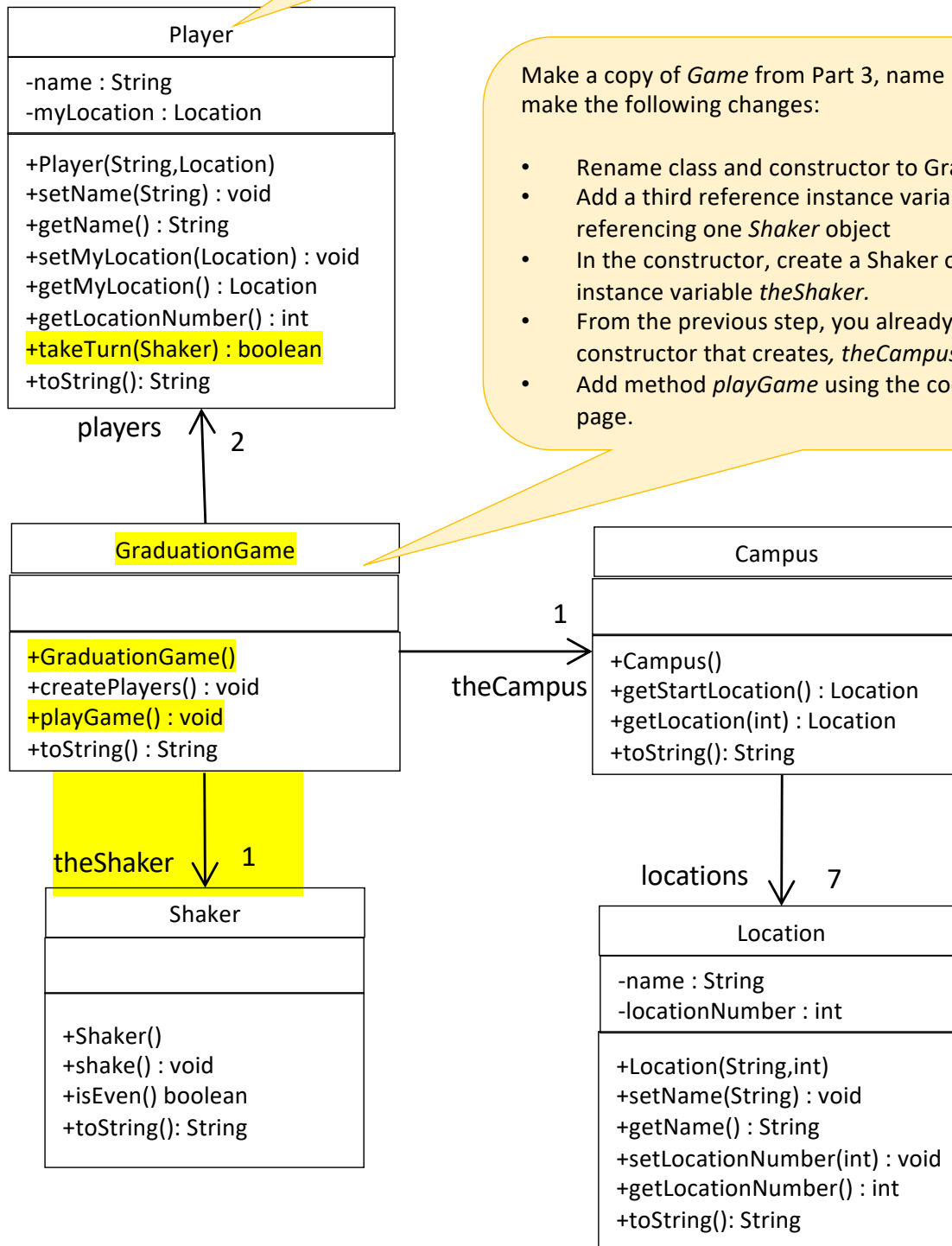
## Lab 6 - Part 4

[10 points]

Add *takeTurn* method to *Player* using the code provided on the next page.

Make a copy of *Game* from Part 3, name it *GraduationGame*, and make the following changes:

- Rename class and constructor to *GraduationGame*
- Add a third reference instance variable named *theShaker* referencing one *Shaker* object
- In the constructor, create a *Shaker* object and assign it to instance variable *theShaker*.
- From the previous step, you already have code in the constructor that creates, *theCampus* and the *players* array.
- Add method *playGame* using the code provided on the next page.



## takeTurn method in Player:

```
public boolean takeTurn(Shaker theShaker)
{
    Scanner input = new Scanner(System.in);
    System.out.printf("%s's turn at %s.\n",name, myLocation.getName());
    System.out.print("        Press Enter to shake the dice...");
    input.nextLine();

    theShaker.shake();
    System.out.println("        " + theShaker);
    if(theShaker.isEven() == true)
    {
        System.out.println("        Sum is even!");
        return true;
    }
    else
        return false;
} //end takeTurn
```

## playGame method in GraduationGame:

```
public void playGame()
{
    boolean gameOver = false;
    while(gameOver == false)
    {
        for(int i = 0 ; i < players.length ; i++)
        {
            Player aPlayer = players[i];
            boolean advance = aPlayer.takeTurn(theShaker);
            if(advance == false)
                System.out.printf("        %s is stuck\n",aPlayer.getName());
            else
            {
                //get player's current location number
                int locNum = aPlayer.getLocationNumber();
                //increment locNum to advance
                locNum++;
                if(locNum == 6)
                {
                    gameOver = true;
                    System.out.printf("Game over! %s is the winner!\n",aPlayer.getName());
                    break; //break out of playing for loop
                }
            }
            else
            {
                Location newLoc = theCampus.getLocation(locNum);
                aPlayer.setLocation(newLoc);
                System.out.printf("        %s is now at %s\n",aPlayer.getName(),newLoc.getName());
            }
        } //end else advance == false
    } //end for loop
} //end while loop

} //end playGame
```

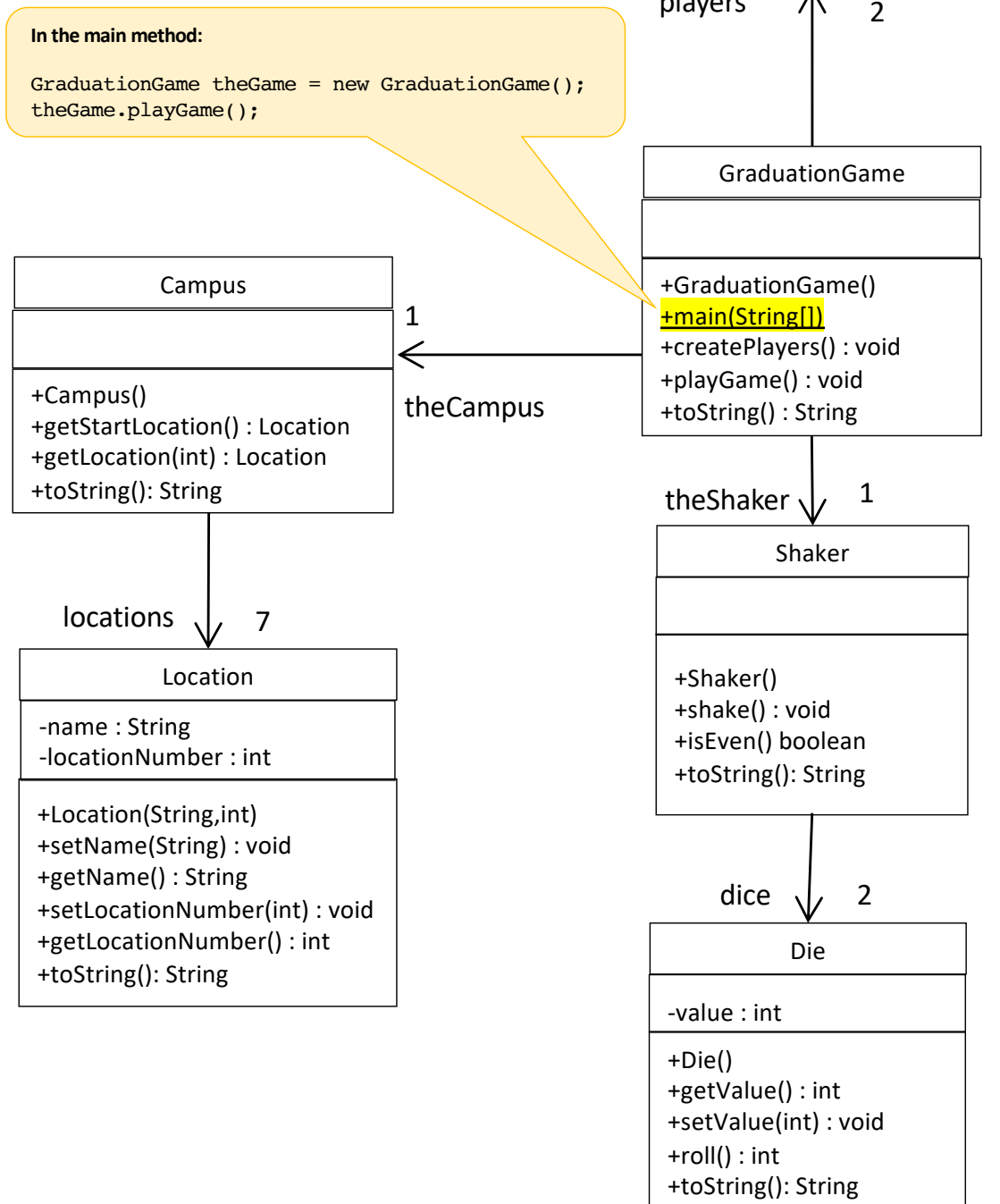


## Lab 6 - Part 5

### [10 points]

Add a *main* method to *GraduationGame* and run/test it.

See sample output on the next page.



## Sample output:

```
% java GraduationGame
Enter player 1's name: Homer
Enter player 2's name: Marge
Homer's turn at Admissions.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 2
    Sum is even!
    Homer is now at Cox School of Business
Marge's turn at Admissions.
    Press Enter to shake the dice...
    Shaker with Die 6 Die 4
    Sum is even!
    Marge is now at Cox School of Business
Homer's turn at Cox School of Business.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 1
    Homer is stuck
Marge's turn at Cox School of Business.
    Press Enter to shake the dice...
    Shaker with Die 5 Die 6
    Marge is stuck
Homer's turn at Cox School of Business.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 6
    Sum is even!
    Homer is now at Dedman College
Marge's turn at Cox School of Business.
    Press Enter to shake the dice...
    Shaker with Die 2 Die 2
    Sum is even!
    Marge is now at Dedman College
Homer's turn at Dedman College.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 2
    Sum is even!
    Homer is now at Simmons School of Education
Marge's turn at Dedman College.
    Press Enter to shake the dice...
    Shaker with Die 5 Die 2
    Marge is stuck
Homer's turn at Simmons School of Education.
    Press Enter to shake the dice...
    Shaker with Die 6 Die 1
    Homer is stuck
```

```
Marge's turn at Dedman College.
    Press Enter to shake the dice...
    Shaker with Die 2 Die 5
    Marge is stuck
Homer's turn at Simmons School of Education.
    Press Enter to shake the dice...
    Shaker with Die 3 Die 6
    Homer is stuck
Marge's turn at Dedman College.
    Press Enter to shake the dice...
    Shaker with Die 1 Die 1
    Sum is even!
    Marge is now at Simmons School of
Education
Homer's turn at Simmons School of Education.
    Press Enter to shake the dice...
    Shaker with Die 2 Die 4
    Sum is even!
    Homer is now at Meadows School of the Arts
Marge's turn at Simmons School of Education.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 5
    Marge is stuck
Homer's turn at Meadows School of the Arts.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 2
    Sum is even!
    Homer is now at Lyle School of Engineering
Marge's turn at Simmons School of Education.
    Press Enter to shake the dice...
    Shaker with Die 3 Die 2
    Marge is stuck
Homer's turn at Lyle School of Engineering.
    Press Enter to shake the dice...
    Shaker with Die 3 Die 6
    Homer is stuck
Marge's turn at Simmons School of Education.
    Press Enter to shake the dice...
    Shaker with Die 4 Die 1
    Marge is stuck
Homer's turn at Lyle School of Engineering.
    Press Enter to shake the dice...
    Shaker with Die 2 Die 2
    Sum is even!
Game over! Homer is the winner!
```

**The Lab is due in Canvas  
Saturday April 9, 2022 at 6:00am.**

### Your Zip file should contain the following:

|                     |                      |
|---------------------|----------------------|
| Campus.java         | Campus.class         |
| Die.java            | Die.class            |
| Game.java           | Game.class           |
| GraduationGame.java | GraduationGame.class |
| Location.java       | Location.class       |
| Player.java         | Player.class         |
| Shaker.java         | Shaker.class         |
| TestCampus.java     | TestCampus.class     |
| TestGame.java       | TestGame.class       |
| TestShaker.java     | TestShaker.class     |