

CSE1341 - Lab 7 Assignment

Overview

In Lab 6 you created the BatterUp baseball simulation using object oriented programming. In this lab, you will make modifications to that program.

Pre-Lab

No pre-lab this week. Just make sure Lab 6 is completed and working to use as the start of Lab 7.

Lab (100 Points)

Create the BatterUp system using object oriented programming, following the design provided on the following pages. Your output should match the format shown on the last page, although your actual output will vary based on the outcome of the game.

Submit the java and class files via Canvas (as a single zip-file). Include a comment block at the top of each *Java* file that includes your name, student id number, and “Lab 7-Fall 2018”. Also be sure to include the answers to the post-lab questions found in these instructions.



NOTES:

Each program should include comments that explain what each block of code is doing. Additionally, the programs should compile without errors, and run with the results described in the exercise. The following deductions will be made from each exercise if any of the following is incorrect or missing:

Proper formatting [5 points]

Proper names for classes and variables [5 points]

Comments [5 points per class]

Program doesn't compile [10 points]

Source code (java file) missing [10 points]

Executable (class file) missing [10 points]

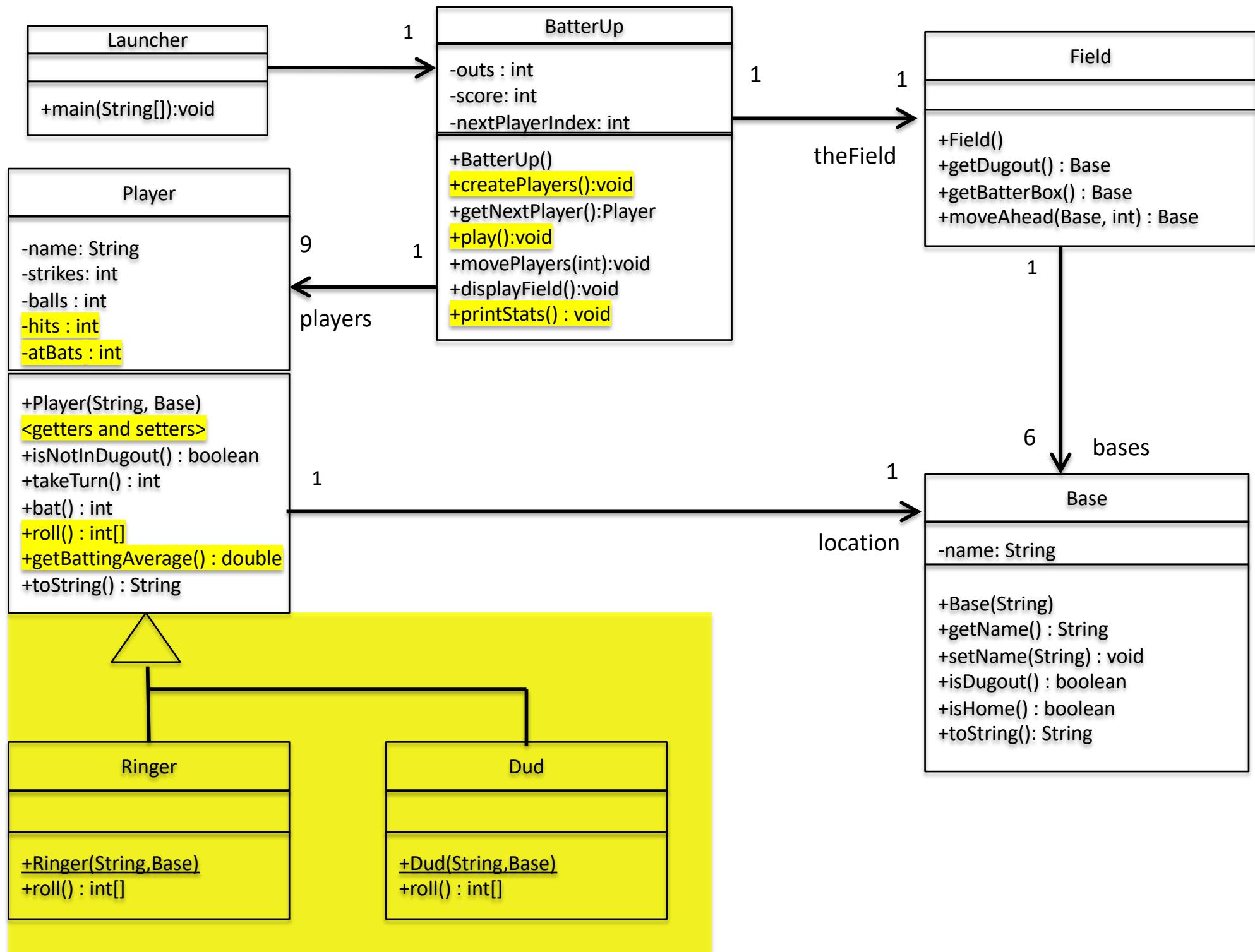
Missing array where an array was required [5 points each]

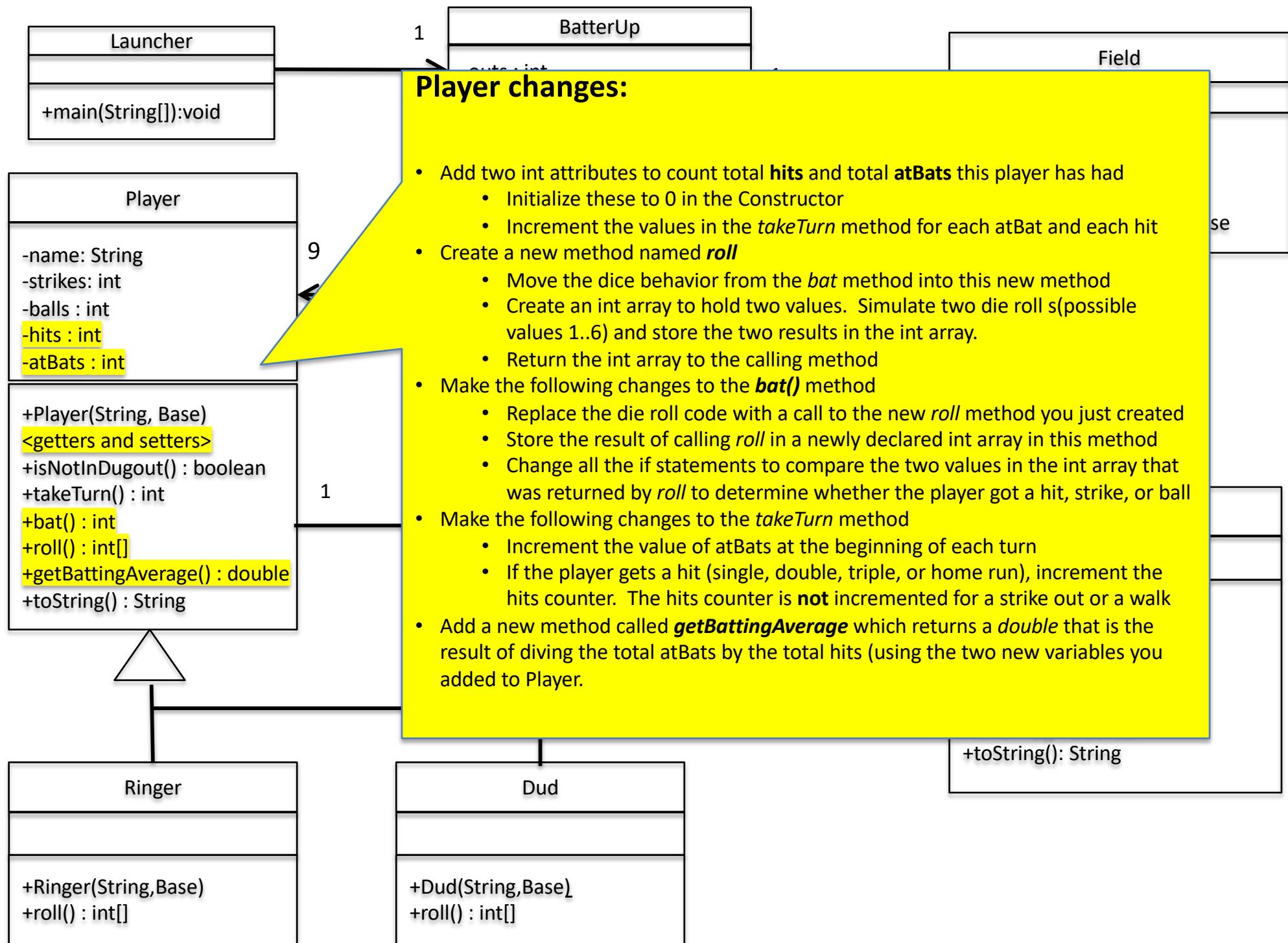
Missing loop where a loop was required [5 points each]

Missing class from the design provided [10 points each]

Missing method from the design provided [5 points each]

This Lab is due Saturday November 17 at 6:00am.





Players, Ringers and Duds

The odds of a player getting a hit using two 6-sided dice is $4/36$ or 11%. See the table below:

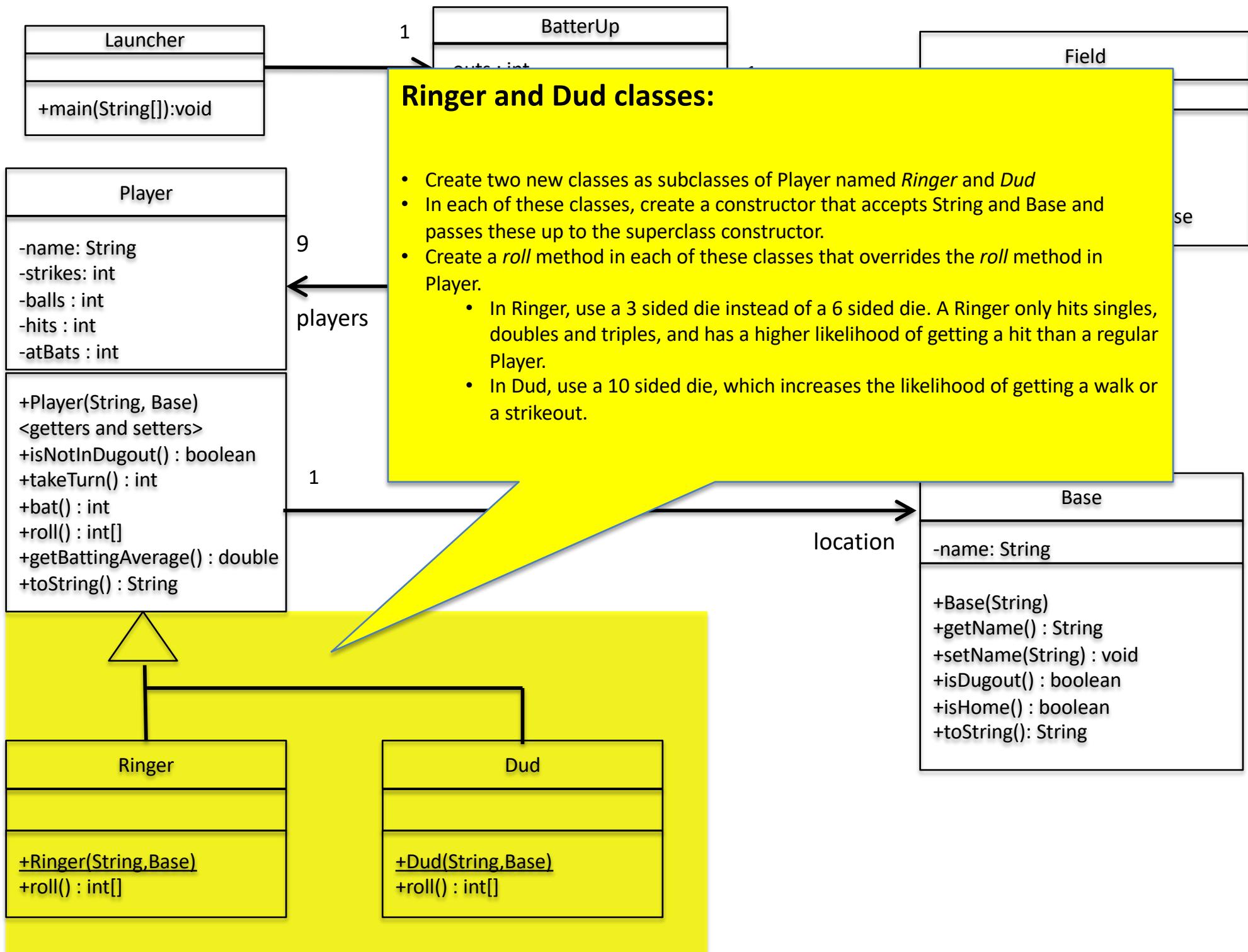
		1	2	3	4	5	6	
		1	Single	Ball	Strike	Ball	Strike	Ball
Die 2	1	Ball	Double	ball	Strike	Ball	Strike	
	2	Strike	Ball	Triple	Ball	Strike	Ball	
	3	Ball	Strike	Ball	Homer	Ball	Strike	
	4	Strike	Ball	Strike	Ball	Strike	Ball	
	5	Ball	Strike	Ball	Strike	Ball	Strike	
	6	Ball	Strike	Ball	Strike	Ball	Strike	

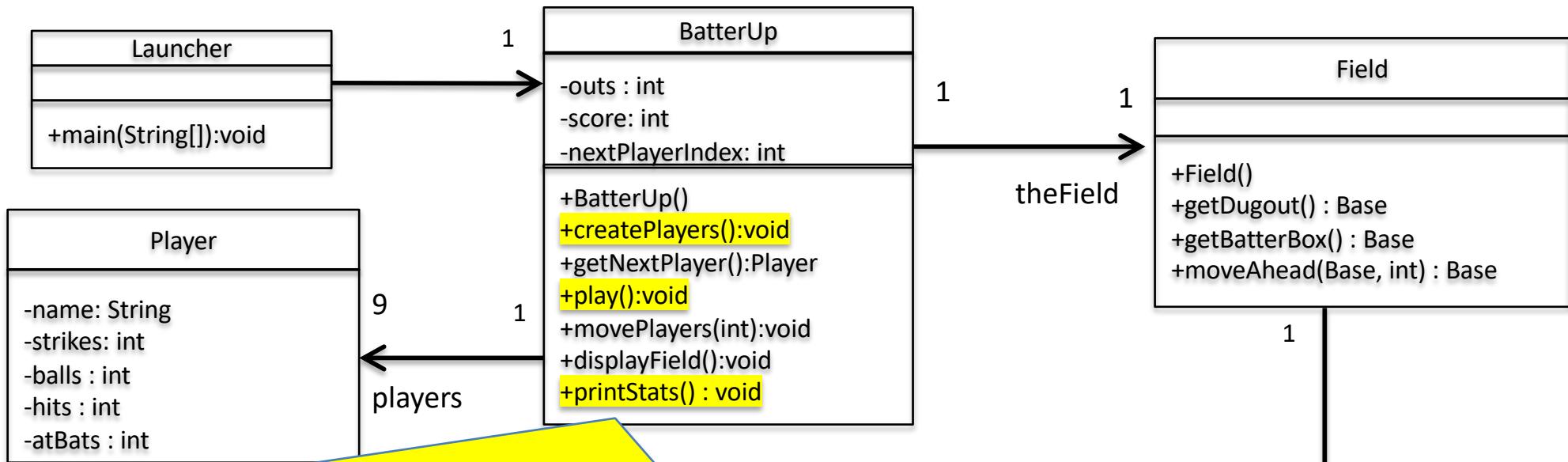
Let's introduce a different type of player called a "Ringer" who is much better at hitting singles, doubles and triples, but never hits home runs. Let's simulate this by having a Ringer roll 3-sided dice. This increases the odds of getting a hit to $3/9$ or 33%.

		1	2	3	
		1	Single	Ball	Strike
Die 2	1	Ball	Double	ball	
	2	Strike	Ball	Triple	
	3	Ball	Strike	Ball	

Let's also introduce another type of player called a "Dud" who can make any type of hit, but has a much better chance of getting balls and strikes. Let's simulate this by having a Dud roll 10-sided dice. This reduces the odds of getting a hit to $4/100$ or 4%.

		1	2	3	4	5	6	7	8	9	10
		1	Single	Ball	Strike	Ball	Strike	Ball	Strike	Ball	Ball
Die 2	1	Ball	Double	ball	Strike	Ball	Strike	Ball	Strike	Ball	Strike
	2	Strike	Ball	Triple	Ball	Strike	Ball	Strike	Ball	Strike	Ball
	3	Ball	Strike	Ball	Homer	Ball	Strike	Ball	Strike	Ball	Strike
	4	Strike	Ball	Strike	Ball	Ball	Strike	Ball	Strike	Ball	Ball
	5	Ball	Strike	Ball	Strike	Ball	Ball	Strike	Ball	Strike	Ball
	6	Strike	Ball								
	7	Ball	Strike	Ball	Strike	Ball	Strike	Ball	Strike	Ball	Ball
	8	Strike	Ball								
	9	Ball	Strike								
	10	Ball	Strike								





BatterUp changes:

- In the **play** method, create another outer loop to repeat for nine innings. Print the inning number with the score at the end of each inning.
- In the **createPlayers** method, read in the player name and type for the nine players from an external text file provided with the lab instructions. This file is named **players.txt**.
 - Based on the type (Average, Dud or Ringer) create an instance of *Player*, *Dud*, or *Ringer* and add to the *players* ArrayList.
- Create a new method called **printStats**, which creates an external text file with a list of stats for each player that includes name, hits, at-bats, and batting-average.
 - You will need to retrieve each Player from the *players* ArrayList and send the *getters* to those objects to retrieve the data needed to print in the output file.
 - Use printf substitution symbols to align the data in the output file.
 - Call *printStats* in the *play* method after the end of the 9th inning
 - See sample output file to the right ->

GAME STATS:			
PLAYER	HITS	AT-BATS	AVERAGE
Amazing Amy	11	11	1.000
Bozo Bob	2	11	0.182
Quick Carl	11	11	1.000
Dizzy Diana	0	11	0.000
Average Ed	6	11	0.545
Fast Francis	9	10	0.900
So-So Sarah	2	10	0.200
Hitter Hank	9	10	0.900
Dizzy Izzy	0	10	0.000

```
$ java Launcher  
Inning 1
```

Sample Output:

```
SCORE: 0
```

```
[ 1 ] empty [ 2 ] empty [ 3 ] empty
```

```
Amazing Amy is batting  
Rolled 3 1 STRIKE!  
Rolled 1 1 Single!
```

```
SCORE: 0
```

```
[ 1 ] Amazing Amy [ 2 ] empty [ 3 ] empty
```

```
Bozo Bob is batting  
Rolled 3 9 STRIKE!  
Rolled 7 7 STRIKE!  
Rolled 7 6 BALL!  
Rolled 4 8 STRIKE!  
Strike out!!
```

```
SCORE: 0
```

```
[ 1 ] Amazing Amy [ 2 ] empty [ 3 ] empty
```

```
Quick Carl is batting  
Rolled 1 1 Single!
```

```
SCORE: 0  
[ 1 ] Quick Carl [ 2 ] Amazing Amy [ 3 ]  
empty
```

```
Dizzy Diana is batting  
Rolled 2 9 BALL!  
Rolled 3 2 BALL!  
Rolled 6 8 STRIKE!  
Rolled 3 6 BALL!  
Rolled 9 8 BALL!  
Walk
```

```
SCORE: 0
```

```
[ 1 ] Dizzy Diana [ 2 ] Quick Carl [ 3 ] Amazing Amy
```

```
Average Ed is batting  
Rolled 1 5 STRIKE!  
Rolled 4 3 BALL!  
Rolled 5 2 BALL!  
Rolled 5 2 BALL!  
Rolled 2 4 STRIKE!  
Rolled 2 4 STRIKE!  
Strike out!!
```

```
SCORE: 0
```

```
[ 1 ] Dizzy Diana [ 2 ] Quick Carl [ 3 ] Amazing Amy
```

```
Fast Francis is batting  
Rolled 2 2 Double!  
Amazing Amy SCORED!!  
Quick Carl SCORED!!
```

```
SCORE: 2
```

```
[ 1 ] empty [ 2 ] Fast Francis [ 3 ] Dizzy Diana
```

```
So-So Sarah is batting  
Rolled 3 5 STRIKE!  
Rolled 2 5 BALL!  
Rolled 3 6 BALL!  
Rolled 2 4 STRIKE!  
Rolled 1 3 STRIKE!  
Strike out!!
```

```
THREE OUTS!  
INNING 1 OVER WITH A SCORE OF 2  
Inning 2
```

<data omitted>

Expect a high score with a voluminous amount of output for nine innings of play.

```
SCORE: 65
```

```
[ 1 ] empty [ 2 ] Amazing Amy [ 3 ] empty
```

```
Quick Carl is batting  
Rolled 2 1 BALL!  
Rolled 3 2 BALL!  
Rolled 2 2 Double!  
Amazing Amy SCORED!!
```

```
SCORE: 66
```

```
[ 1 ] empty [ 2 ] Quick Carl [ 3 ] empty
```

```
Dizzy Diana is batting  
Rolled 2 5 BALL!  
Rolled 10 8 STRIKE!  
Rolled 8 8 STRIKE!  
Rolled 10 4 STRIKE!  
Strike out!!
```

```
THREE OUTS!  
INNING 9 OVER WITH A SCORE OF 66
```