Welcome to Ball Worlds

Communities of interacting objects; UML
BallWorlds Intro
Work on BallWorlds

Interaction in UML Diagrams

- Many "real" programs involve dozens or hundreds of classes, with complex interactions among objects from those classes.
- For large programs can't just start writing code and hope it works out!
- UML Class Diagrams can help us to visualize the classes and their interactions before we write the code.

BallWorlds Intro

- We've done "from scratch" programs and programs with small templates

- BallWorlds is more realistic:
  - You'll write a small part of a larger program, already begun by others
  - Your parts have to fit with the other parts
  - You'll have to spend some time understanding the other parts in order to succeed.

Goals

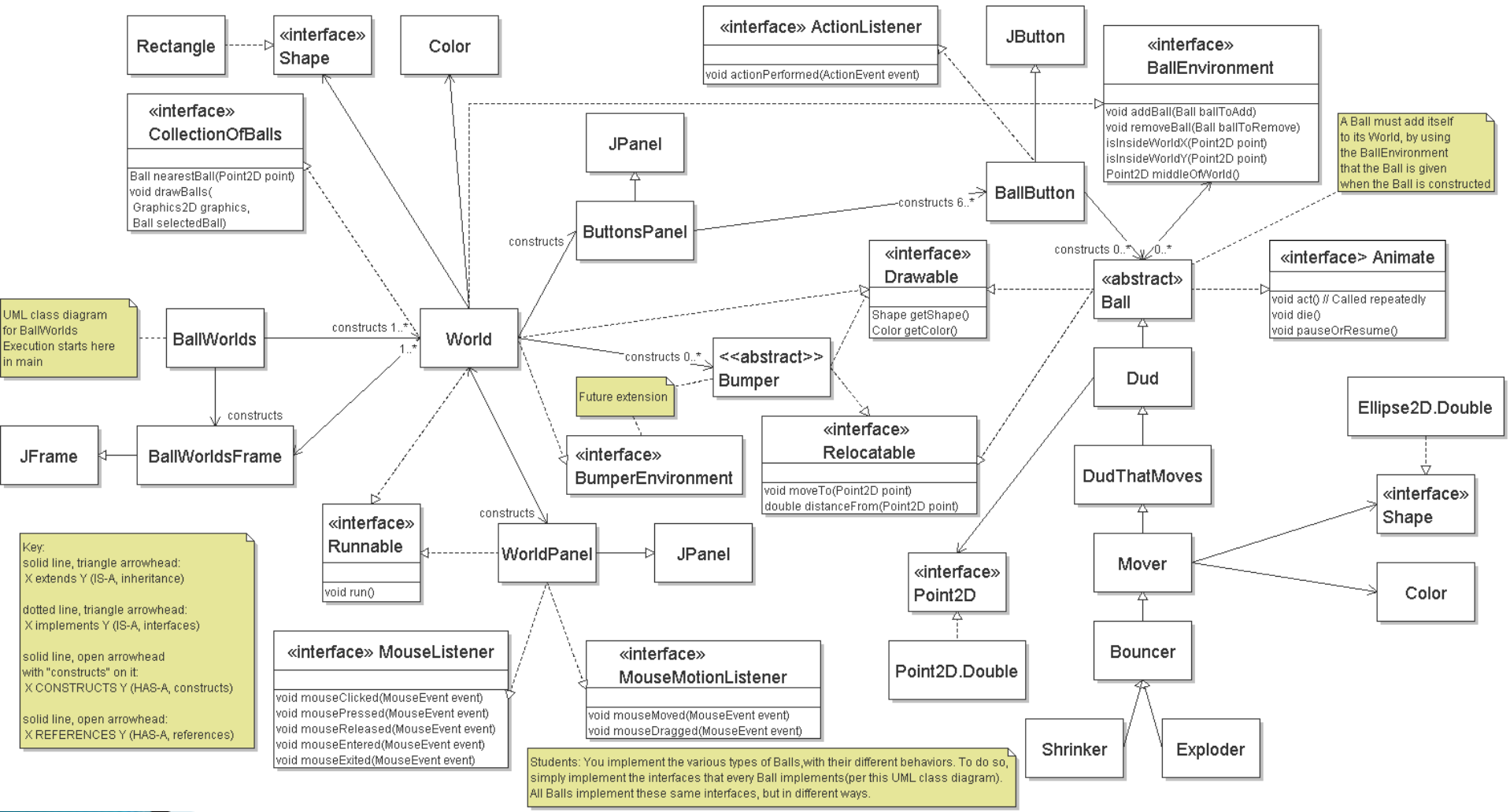
- Learn to read and use UML Class Diagrams to understand existing code
- Practice analyzing what features of existing code are relevant to your task
- Use inheritance to solve problems

- Demo:
  - Demonstrate the program
  - Can we change the number of worlds?

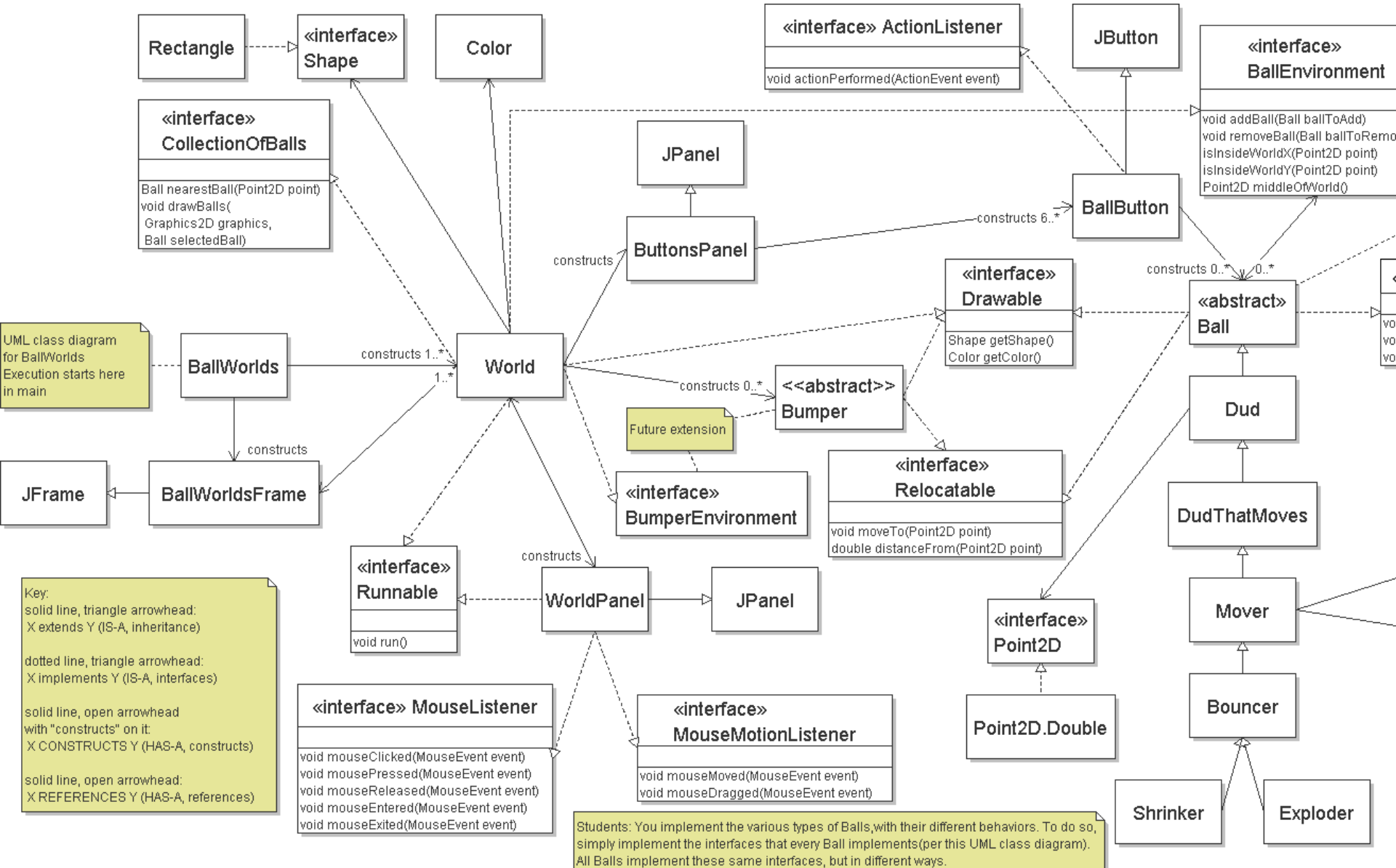
Creation of the Worlds

```
/*
 * Makes the given number of Worlds, giving each the given frame.
 * Rotates between 3 pre-assigned sizes and colors for the Worlds.
 */
private static void makeWorlds(int numberOfWorlds,
                               BallWorldsFrame frame) {
    ArrayList<Dimension> dimensions = new ArrayList<Dimension>();
    ArrayList<Color> colors = new ArrayList<Color>();

    dimensions.add(BallWorlds.world1Size);
    dimensions.add(BallWorlds.world2Size);
    dimensions.add(BallWorlds.world3Size);

    colors.add(BallWorlds.world1Color);
    colors.add(BallWorlds.world2Color);
    colors.add(BallWorlds.world3Color);

    for (int k = 0; k < numberOfWorlds; ++k) {
       new World(dimensions.get(k % 3), colors.get(k % 3), frame);
    }
}
```

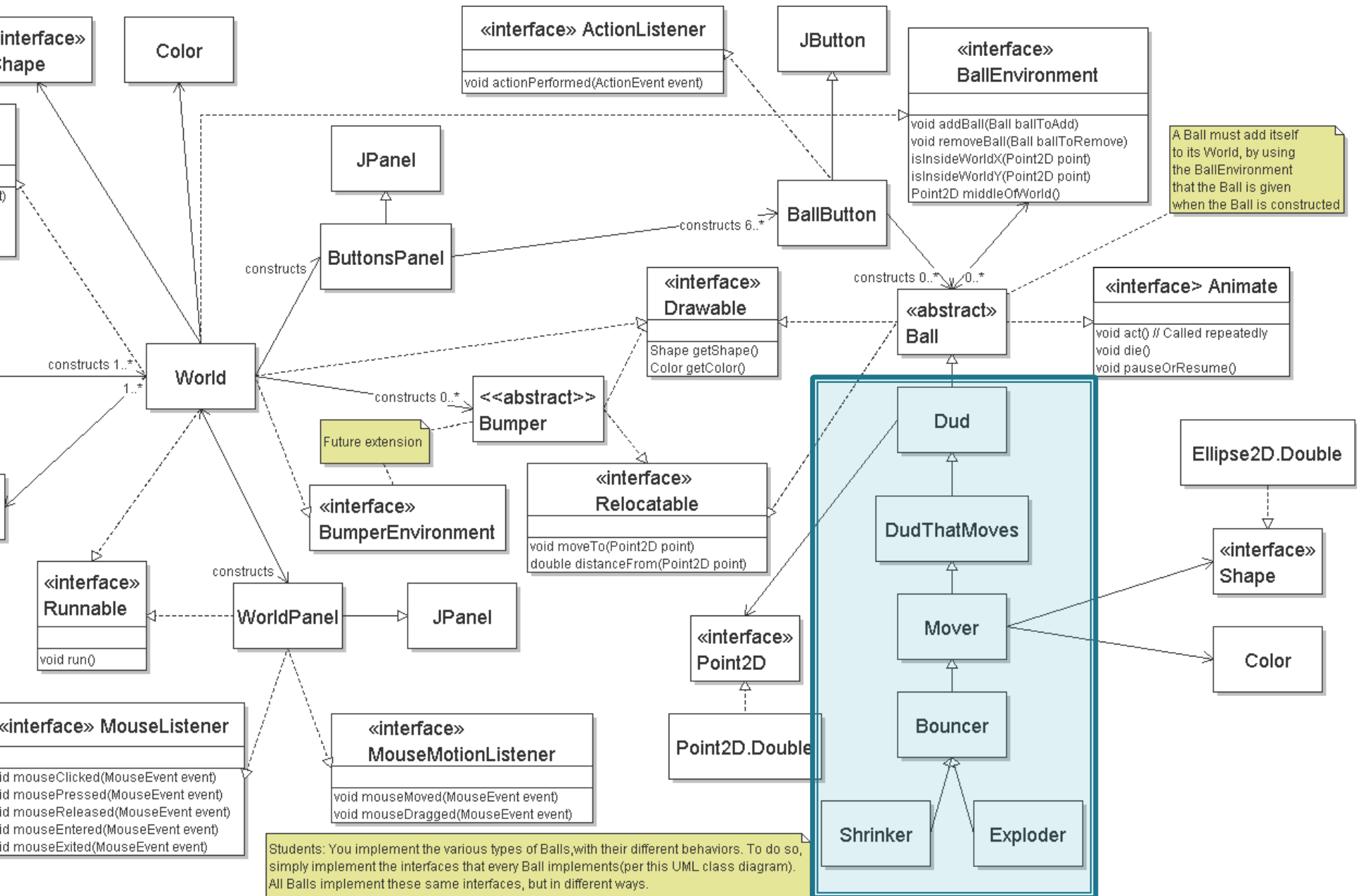
UML Class Diagram for BallWorlds



More details on Part of Diagram



Focus on the Part You Will Implement



Ball Class

- Abstract
- Implements which interfaces?
- What data might be needed for every kind of Ball?

- Let's do a little bit of code exploration, then write Dud together.