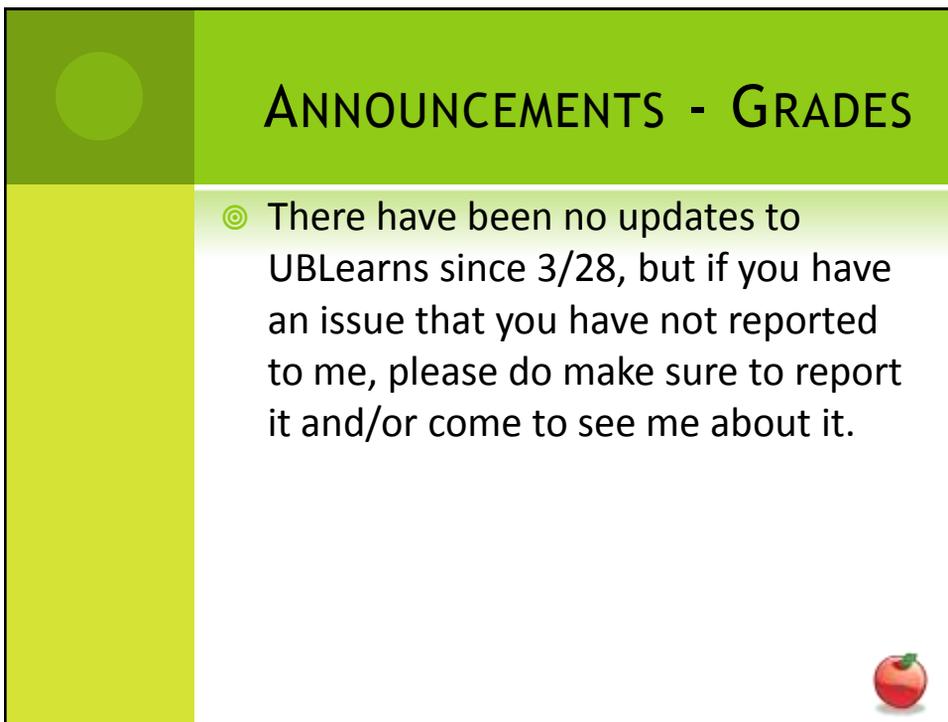


CSE 113 B  
April 4 - 8, 2011



## ANNOUNCEMENTS - GRADES

- ⦿ There have been no updates to UBLearns since 3/28, but if you have an issue that you have not reported to me, please do make sure to report it and/or come to see me about it.



## ANNOUNCEMENTS - LAB

- ⦿ Lab 8 started in lab this week.
- ⦿ Labs 6, 7, 8 will be graded by Web-CAT, but the grading is not functional at this time.
- ⦿ Practice Assignment 6 has been posted and grading is functional.
- ⦿ Practice Assignments 7 & 8 will be posted at some point soon.



## ANNOUNCEMENTS - PRACTICAL EXAM 3

- ⦿ May 5, 6, 9, 10
- ⦿ Schedule will be posted on the Practical Exam 3 information page (which will be linked off of the Schedule page).
- ⦿ Information about what material will be on the exam is also posted there.



## ANNOUNCEMENTS - EXAMS

- ⦿ Pick up Exam 1 & 2 if you have not already done so.
- ⦿ Exam 3 Monday, April 11<sup>th</sup> in lecture (covers Chapters 6-8).
- ⦿ Review for Exam 3 on Friday, April 8<sup>th</sup>.
- ⦿ Review sheet posted.



6

## LAB 8 TIPS

- ⦿ Create a method inside Obstacle named `getImageName`.
  - ⦿ The return type of this method should be `String`.
  - ⦿ The method takes no parameters.
  - ⦿ Inside the method, you should “return” the instance variable that stores the string that represents the name of the image file that the obstacle displays.
    - ⦿ If you don't have an instance variable storing this information, you need to create one.



7

## LAB 8 TIPS

```
public String getImageName() {  
    return imageName;  
}
```



8

## TYPECASTING

- ⦿ Explicitly coding into a program to convert the type of an object to some other type.
- ⦿ Must be used carefully.



9

## CHECKING FOR INTERSECTIONS

```
Actor a = getOneIntersectingObject(Obstacle.class);
If(a != null) {
    ...
}
```

- ⊙ **a** is an Actor (Notice it's type)
- ⊙ But we also know **a** is an Obstacle. How?



10

## So....

- ⊙ We could typecast **a** to be an obstacle instead of an Actor.

```
Actor a = getOneIntersectingObject(Obstacle.class);
If(a != null) {
    Obstacle obs = (Obstacle) a;
    ...
}
```

- ⊙ **obs** is an Obstacle



11

## WHY WOULD WE DO THIS?

- ⦿ With the variable **a**, we can call any methods from **Actor** on the a object.
- ⦿ BUT
- ⦿ We can't call any methods from **Obstacle**.
- ⦿ With the variable **obs**, we can call methods from BOTH **Actor** and **Obstacle**.



12

## OK - So WHAT?

- ⦿ **getImageName** is only defined in Obstacle (NOT Actor)
- ⦿ We will need to call that method to determine which type of obstacle the hero has run into during the course of the game.



13

## EQUALS METHOD

- ⦿ Once we have gotten our image name from the obstacle, we will need to compare it to known image names to determine which type of object we collided with.
- ⦿ We will use the **equals** method from the String class to do this.



14

## EXAMPLE

```
Actor a = getOneIntersectingObject(Obstacle.class);
If(a != null) {
    Obstacle obs = (Obstacle) a;
    String image = obs.getImageName();
    If(image.equals("shamrock.png")) {
        ...
    }
    ...
}
```



## BACK TO FUN WITH IMAGES

- ⊙ New actor in our drawing scenario
  - ⊙ Gets inserted into the world at random times at random locations.
  - ⊙ When inserted, it starts off small and begins to grow.
  - ⊙ When it reaches a certain size, it disappears and puts  $n$  more Square objects into the world.

