Objectives Students will be able to…

* **Describe** which types automatically convert into others when appearing together.
* **Predict** how an expression with mixed types will evaluate.
* **Convert** types by casting.

Assessments Students will...

* **Use** “zombie rules” of precedence to correctly write code that yields a given answer
* **Create** their own expressions
* **Predict** output by completing and trading worksheets

Homework Students will...

* **Read** HW 2.3 up to “Nested for Loops”
* **Complete** self-check question 18
* **Finish** the worksheet if not completed in class

# Materials & Prep

* **Projector and computer** (if you are able to/opt to use Eclipse with your students)
* **White paper** **and** **markers**
* **Classroom copies** of WS 2.4
* **Zombie/werewolf video** (<https://www.youtube.com/watch?v=ZL-58z3HxUI&feature=youtu.be&t=2m41s>)
  + - * Probably want to play on mute!
* **Unicorn image** ([http://1.bp.blogspot.com/\_WOWQJUlRtKQ/TRD5BW8v5GI/AAAAAAAABAY/llLQ4VSCskc/s1600/moon-wallpaper 9.jpg](http://1.bp.blogspot.com/_WOWQJUlRtKQ/TRD5BW8v5GI/AAAAAAAABAY/llLQ4VSCskc/s1600/moon-wallpaper%209.jpg))

# Pacing Guide

|  |  |
| --- | --- |
| Section | Total Time |
| Bell-work and attendance | 5min |
| Introduction to Mixing Types | 10min |
| Activity 1 | 15min |
| Introduction to Casting | 5min |
| Activity 2 | 15min |
| Turn in worksheets, wrap up | 5min |

# Procedure

*Today’s class uses the concept of infection and cure as metaphors for type conversion and type casting. Have the YouTube video of the werewolves and zombies playing as students arrive. Once students have completed bellwork, hook your class with a discussion of the zombie/werewolf fight happening on the video.*

## Bell-work and Attendance [5 minutes]

## Introduction to Mixing Types [10 minutes]

1. Here are some questions to guide this opening conversation:

* + - * Who has seen movies about werewolves or zombies?
      * Can anyone tell me how you become a werewolf?
      * How do you become a zombie?
      * What do you think would happen if a zombie bit a werewolf?

2. Introduce the following metaphor:

* + - * If a human gets bitten by a werewolf, they become a werewolf. If a werewolf or human gets bitten by a zombie, it becomes a zombie.

So: zombie > werewolf > human

* + - * The same sort of thing happens if we put together different types in an expression (aka “**mixing types**”):
        + If an int is placed in an expression with a double, Java converts it to a double. (It gets infected and becomes a double.)
        + If a double or int is placed in an expression with a String, Java converts it to a String. (The werewolf double gets bitten by the zombie String, and is now also a zombie String.)
        + So: int => double => String
      * In Java we call this “promoting” because the double holds more information than int, and String holds more information than double! (We’ll learn more about what information is stored in a String, but for now just remember its true.)
      * Spot-check your students by asking them to name the fantasy-equivalent for each type in an example on the board:
        + 2 + 2.3 (evaluates to 4.3 because human/int 2 is promoted/infected to werewolf/double 2.0)

## Activity 1 [15 minutes]

1. Have students begin WS 2.4 alone or quietly in pairs (this strategy is recommended for ELL classes). Direct students to only work through werewolf and zombie questions (stop at the unicorns).

2. Students WILL trip up on the mixed type String questions.

a. You should definitely cover an example or two of these on the board, and you may find that you have to switch to whole group instruction for the majority of these.

b. Your priority should be to thoroughly complete several examples, and to see that students can complete these questions correctly without your aid. If you need to slow down your pace, go for it!

3. If students are completing the questions on-pace, bring the class together for another round of whole-group instruction.

## Introduction to Casting [5 minutes]

1. Introduce the concept of **casting** by asking students what to do if they really want to preserve the human race (int type)?

2. Review/repeat the earlier concepts by asking students: How do we convert werewolf-doubles to human-ints?

3. Introduce casting (Ham this up: “As we all know, you can cure anything with unicorn magic!”)

* + - * Unicorn magic = “casting” (remember that Unicorns CAST a magical spell to protect humans or werewolves from infection/conversion)
      * To cure a werewolf-double, so he becomes a human-int, you can CAST it by putting the name of the type you want in front of the value you want to convert (cast):

(int) 42.9 => 42

IMPORTANT: Java just cuts off that extra part after the decimal—it always rounds toward zero.

* + - * BUT unicorn magic can’t turn zombies back into anything because magic can’t bring you back from the dead. (Once you’re a zombie, you’re always a zombie!) So casting (int)str or (double)str on a String str doesn’t work.

## Activity 2 [15 minutes]

1. Have students return to WS 2.4, starting on the unicorn/casting segment of the exercise.

2. Work through 1 or 2 problems as a whole group before you leave students to their own devices.

3. If student frustration levels are high, bring the class back to whole-group and work through a few more examples slowly and thoroughly. If you need to revisit the analogy or have students flip through their books again, you should do so at this point.

## Turn in Worksheets, Wrap Up [5]

1. At the end of class, collect the completed worksheets.

# Accommodation and Differentiation

If you have students who are speeding through this lesson, you should encourage them to:

* + - * Draw or design a poster showing the relationship between mixed types. The poster should contain large images of humans, werewolves, zombies and unicorns so that students can see the relationships from all around the room.

Even if you do not have a student do this for you, we HIGHLY recommend making a poster of this sort. Students do not forget it! If all else fails, you can use the image saved as Poster 2.4.