Objectives

- Eclipse debugger
- Preparation for Preliminary Implementation Demo on Monday

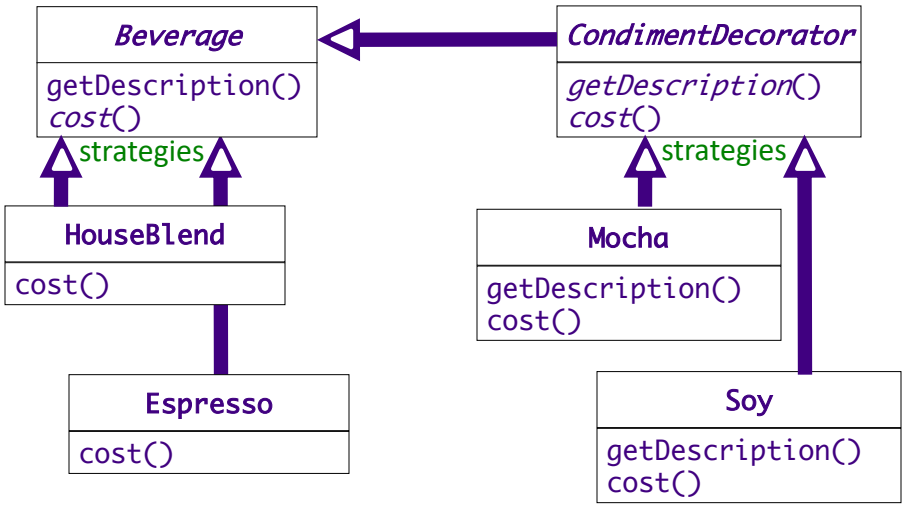
1

Review

- What is the decorator design pattern?
  - When is it useful?
  - How is it implemented?
- What is our workflow in git?
  - Even more important with team projects

2

Review: Using **Decorator**



UML Diagram

Review: Mocha's Implementation

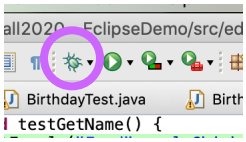
```
public class Mocha extends CondimentDecorator {

   private Beverage beverage;

   public Mocha(Beverage beverage) {
      this.beverage = beverage;
   }

   public String getDescription() {
      return beverage.getDescription() + ", Mocha";
   }

   public double cost() {
      return .20 + beverage.cost();
   }
}
```

Handles part it knows about,
Delegates rest to Beverage;
Example of OCP

ECLIPSE DEBUGGER

5

Eclipse Debugger



1. Set breakpoint
   - Near and BEFORE point of failure
2. Run program in debug mode
   1. Program pauses when it hits a breakpoint
3. Inspect variables
4. Step through program, inspecting variables
   - Step into, over, and return

6

Commands

- Step Into
  - Executes the current line
  - If the current line is a method call, the debugger steps into the method's code
- Step Over
  - Executes a method without stepping into it in the debugger
- Step Return
  - Steps out to the *caller* of the currently executing method
  - Finishes the execution of the current method and returns to the caller of this method

Testing Code

- Main methods in
  - picasso.Main.java
  - picasso.parser.Tokenizer.java
  - test.ParserTestDriver

Smooth seas
do not make
for skillful sailors

PREPARING FOR MONDAY

9

Plan for Monday: Preliminary Functionality

- All virtual class
- 10-minute demos via Zoom
  - One member of your team will share their screen
  - Demo the required functionality
  - Then, focus on the design of the code
    - What did you do to make the functionality?
    - How will you add reading the expression from a file?
    - How will you add other components?
  - Thoughts on extensions
  - Questions/concerns

http://cs.wlu.edu/~sprenkle/cs209/projects/final_proj.php#deliverables

10

Start Thinking About

- Reporting errors to users
  - Currently: in the output but users aren't going to see that
  - Helpful errors → translated for users
- Opening a file that contains an expression
- Assignment statement
- Functions with multiple arguments, image names
- Extensions

Team Demo Order on Monday

```
>>> teams = [ "Blocks", "Duplo", "Jigsaw", "Link", "Mega" ]
>>> import random
>>> random.shuffle(teams)
>>> teams
['Jigsaw', 'Link', 'Mega', 'Duplo', 'Blocks']
```

[3 min course announcements]

- 10:03: Jigsaw Puzzles
- 10:14: Linkimals
- 10:25: MegaBloks
- 10:36: Duplos
- 10:47: Building Blocks [test breakout rooms]

Group Work and Questions

13

Secondary Goals

- You're going to figure out that your final design isn't perfect—maybe not even good!
  - Fix more critical and/or smaller things
    - Refactoring!
  - Note larger things
    - analysis/post-mortem due at end of finals week

Good judgment comes from experience.
How do you get experience?
Bad judgment works every time.

14