

# CS 2410 – Fall 2016

## Assignment #7 Polyabstraheritance

### Introduction

The purpose of this assignment is to get experience:

- Using basic GUI components
  - Label (hint)
  - TextField
  - Layouts (Choose what you think is best to meet requirements)
- Java Concepts
  - Polymorphism
  - Abstract classes
  - Abstract methods
  - Inheritance
  - Interfaces
  - instanceof operator

### General requirements

In order to receive full credit your code must follow appropriate class conventions

### Program overview

You will create a simple GUI that will display a number of buttons. Users will choose from a ComboBox to display characteristics of the objects you have stored, or they will select buttons to create a new object.

### Requirements

Your program should implement the following

1. A GUI application with a Tool area at the top or side, and a main display area
2. Create a package called view, and put the following inside it
  - a. A class with a start method to create and display your GUI
    - i. Contains an instance of ToolBar
    - ii. Uses appropriate Layouts
    - iii. Contains an area for display and input
      1. This area changes depending on what the user chooses
    - iv. Display area
      1. Print out messages as described below
    - v. Add area
      1. Lets a user create a new object
      2. Cancel and Save buttons
        - a. Located in the lower right area, even if window is resized
        - b. Cancel
          - i. Erases all entered text
        - c. Save
          - i. Creates object based on input
          - ii. Displays a confirmation dialog pop up
          - iii. Erases all entered text so user can add another
  3. You may choose to make another class (or classes) to define this Add area and then use an instance of the class(es) to display.
- b. ToolBar.java class
  - i. Includes a ComboBox to choose what to display (see below)
  - ii. Includes Buttons to select a type of object to add

- iii. Use an appropriate Layout type
  - iv. You can use ToolPane.java from the last assignment for guidance
- 3. Create a package called model, and put the following inside it
  - a. Create an interface called Simpleton
    - i. Include the following abstract methods
      - 1. String getName();
      - 2. String doMath();
      - 3. String saySomethingSmart();
  - b. Create an interface called PersonType
    - i. Include the following abstract method
      - 1. String getPersonType();
  - c. Create an abstract class called Smarty
    - i. implement Simpleton
    - ii. implement PersonType
    - iii. Include the following methods
      - 1. Non-abstract
        - a. getIQ()
      - 2. Abstract
        - a. getIncome()
  - d. Create additional classes
    - i. HourlyWorker (works so many hours for so much pay per hour)
      - 1. Uses Simpleton and Smarty (think about what that means first!)
      - 2. Create an appropriate constructor based on needed info
        - a. This means don't use any public setters. All initialization data should be passed as a parameter to the constructor
      - 3. Create necessary data members
      - 4. Overrides all necessary methods
        - a. Can do multiplication
        - b. Knows a lot about animals
      - 5. Adds method
        - a. getHoursWorked
    - ii. ContractWorker (complete so many contracts for so much pay per contract)
      - 1. Uses Simpleton and Smarty
      - 2. Create an appropriate constructor based on needed info
        - a. This means don't use any public setters. All initialization data should be passed as a parameter to the constructor
      - 3. Create necessary data members
      - 4. Overrides all necessary methods
        - a. Can do division
        - b. Knows a lot about astronomy
      - 5. Adds method
        - a. getContractsCompleted
    - iii. Hobbit
      - 1. Uses Simpleton and PersonType
      - 2. Create an appropriate constructor based on needed info
      - 3. Create necessary data members
      - 4. Overrides all necessary methods
        - a. Can do addition
        - b. Knows a lot about gardening
      - 5. Adds method
        - a. getCarrotsPicked
- 4. Create a package called control and put the following inside it
  - a. Create a single class with the following functionality
    - i. **Create single ArrayList** that will hold the instances of the classes listed previously
    - ii. You may implement the following methods (this is meant as a hint, and is not a

- complete list)
      - 1. void createNewHobbit(<parameters needs>)
      - 2. String getSayList()
      - 3. String getIncome()
  - b. For each item selection in your drop down list, print the following. Note: don't use the toString() method. Print output based on the getter methods in the classes. Also, you should print info for each object instance that has the information available needed to print.
    - i. Math
      - 1. Print name, type, and a math problem they can do.
    - ii. Income
      - 1. Print name, type, and a calculation of much money they make
    - iii. Hours
      - 1. Print name, type, and a statement of how many hours they worked
    - iv. IQ
      - 1. Print name, type, and a statement of of what their IQ is
    - v. Say
      - 1. Print name, type, and a statement of something smart they can say
    - vi. Carrots
      - 1. Print name, type, and a statement of how many carrots they picked
    - vii. Contracts
      - 1. Print name, type, and a statement of how many carrots they picked
  - c. Use a for-each loop to iterate through the ArrayList and print the appropriate information.
5. Other items
  - a. The type for each person should indicate the class of each object's actual type

## **What/How To Turn In (READ THIS)**

Submit your files on Canvas according to class conventions

**Due: November 18, 2016**