



Rochester Institute of Technology
National Technical Institute for the Deaf
Information and Computing Studies Department

Name: Edward Riley

NACA.161 Programming Fundamentals II
In-class Exercise #16 - Inheritance

Overview

This series of exercises is designed to help you understand inheritance concepts.

Relationships

For the next few steps, choose a word that best describes the relationship between the bolded word and the words that follow it.

Use these 3 words to describe the relationships below:

subclass, **attribute**, or **neither**.

- | | |
|--|------------------------------------|
| 1) Train : Amtrak, freight train | <u>sub-class</u> |
| 2) Baseball Team : name, mascot | <u>attribute</u> |
| 3) Racquet Sport : tennis, racquetball | <u>sub-class</u> |
| 4) Sister : brother, aunt, grandfather | <u>neither</u> |
| 5) Computer : manufacturer, purchase date | <u>attribute</u> |
| 6) Motorcycle : bicycle, scooter | <u>Subclass neither</u> |
| 7) Flower : rose, daisy | <u>Subclass</u> |
| 8) Address : city, state, zipcode | <u>attribute</u> |

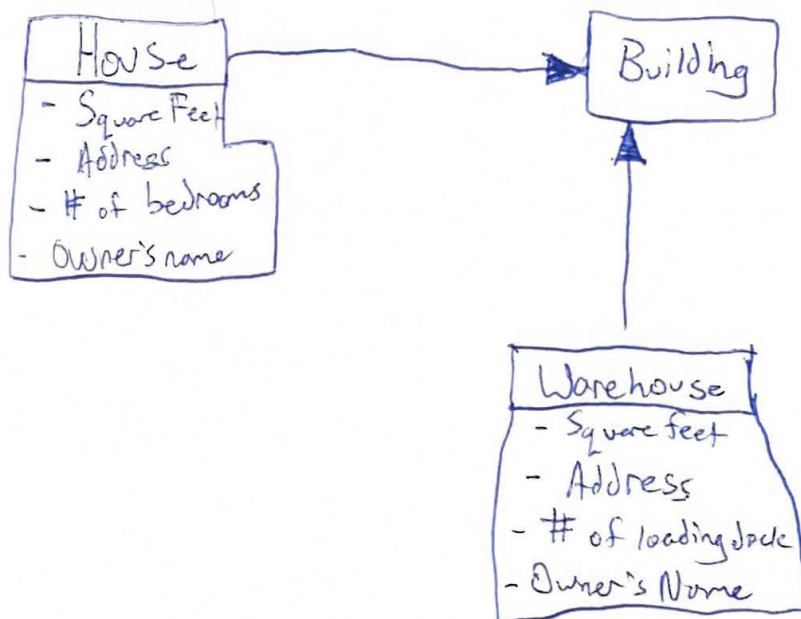
Class Design - UML

9) Organize the following words. You should have one super class, subclasses, and attributes.

- Square feet
- House
- Address
- Number of loading docks
- Number of bedrooms
- Building
- Owner's name
- Warehouse

10) Draw a UML diagram for this exercise. You can use the back of this page to draw a draft of your UML diagram.

Show your instructor your answer before continuing because the rest of the exercise requires you to get this step correct.



Instructor Signature: _____

Create the Superclass

The first class you will create will be the superclass since the other classes will inherit everything from it.

Which class is the superclass?

Building

- 11) Create a **Building** class with the 3 attributes you defined in the previous step.
- 12) Create a default constructor that initializes the attributes to:

 address: "No address specified"

 squareFeet: 0.0

 ownerName: No Owner
- 13) Create accessors and mutators for all the attributes. Make sure the address and owner name has at least one character, and the square feet cannot be set to less than zero.
- 14) Compile the file and fix all the errors. ✓

Create the SubClasses

Now you will create the two sub-classes, **House** and **Warehouse**, that inherit from the superclass **Building** class.

- 15) Create a class named **House** that inherits from the **Building** class.

What Java keyword did you use to inherit from the Building class?

extend

Write the line with this keyword.

Public class House extends Building

Since the **House** class inherits from the **Building** class, what kind of class is it?

subclass

- 16) Create the attribute **numBedrooms**.

- 17) Create a default constructor that sets **numBedrooms** to 0.
- 18) Create an accessor and mutator for the number of bedrooms. Make sure the number of bedrooms can not be set to a value less than zero.
- 19) Compile the file and fix all the errors.
- 20) Create a class named **Warehouse** that inherits from the **Building** class.

What Java keyword did you use to inherit from the Building class?

extends

Write the line with this keyword.

public class Warehouse extends Building

Since the Warehouse class inherits from the Building class, what kind of class is it?

Subclass

- 21) Create the attribute **numLoadingDocks**.
- 22) Create a default constructor that sets **numLoadingDocks** to 0.
- 23) Create an accessor and mutator for the number of loading docks. Make sure the number of loading docks can not be set to less than zero.
- 24) Compile the file and fix all the errors

Testing the Inherited Classes

- 25) Write a class named **TestBuildings** that contains a main method.
- 26) Write the statement that creates a **House** object using the default constructor.

House house1 = new House();

- 27) Add code that will print the address of the **House** object using an accessor.

In which class is the accessor for the address defined?

House

Since the accessor is defined in another class, how are you able to use it?

"house1.address"

- 28) Compile and run **TestBuildings**.

What was the value of the address?

No address specified

In which class did this value get set?

Building

How did the default constructor for **Building** get called?

Building ← House ← TestBuilding

- 29) Print out all of the attributes of the **House** object using the appropriate accessors. Compile and run the file.

What are the values of the **House** attributes?

address: No AddressSpecified

owner name: No Owner

square feet: 0.0

number of bedrooms: 0

In which class are the attributes of the **House** object defined:

address Building

owner name Building

square feet Building

number of bedrooms House

30) Add code (before you call the accessors from the previous step) that will:

- ask the user to enter an address
- get user input
- call the proper mutator to store the address

Assume that the user will enter valid data (no loop or exception catching necessary).

What mutator did you call?

setAddress

In which file is the mutator code located?

Building.java

Why can you access that code?

Because it is inherited to House.java

31) Compile and run the code and make sure you see the address change.

- 32) Create a **Warehouse** object using the default constructor.
- 33) Write the code to display all the attributes values for the **Warehouse** object using the appropriate accessors. Complete the following table:

Attributes for Warehouse Object	Class in which defined
OwnerName	Building
Address	Building
Square feet	Building
Loading Docks	Warehouse

- 34) Add code that will:
- ask the user to enter an the square feet for your **House** object
 - get user input
 - call the proper mutator to store the square feet for the **House** object.
- Do the same for the **Warehouse** object.

You can assume that the user will enter valid data (no loops or exception catching is necessary). Compile the code and make sure it works.

Why is the value of the square feet different for each object?

Because it was given to two different objects

When you complete all of the steps successfully and answer all of the questions, contact your instructor or teaching assistant to check if your application(s) executes correctly and to review your code. We will initial the line below.

_____ Successful execution of code

If you do not finish the program during the class period, contact your instructor or teaching assistant to check to review your code and initial below.

_____ Code not completed during lab time

You may then submit your work at the start of next class. You may not use the work period of the next class to complete this assignment. If you do not have a signature, then you can not receive any points for this assignment.