Introduction to Inheritance

Inheritance is the process of creating a new class from an old class.

The original class is called any of the following names.

parent class or super class or base class

The derived class is called.

child class or subclass

Inheritance forms an is a relationship between the original class and the derived class.

For example, if the parent class is Book and the derived class is Novel then

Novel is a Book.

The idea behind inheritance is software reuse. Take a component that is already available and create a new class that extends the original class' capabilities.

Derived Classes

Derived classes inherit the public methods of the original class.

Derived classes do not inherit the private variables, private methods, and constructors of the original class.

Create an Address Class that has the following data

street address

city

state

postal code

Create appropriate accessor and mutator methods to return and modify the private data.

Create a test plan to test all of the Person methods and Address methods

Create a test client to implement your test plan.

Create a class Person that contains fields for

first name

last name

social security – String

Address data (use other class)

Get the Person class and Address class working together.

public class address

 {

 public address(String s)

 {street = s;

 }

 private String street;

 public String getStreet()

 {return street;

 }

 public void setStreet(String s)

 {street = s;

 }

 }

public class Person

 {

 private String na;

 private address a;

 public Person(String n,String s)

 {a = new address(s);

 na = n;

 }

 public String getAddress()

 {return a.getStreet();

 }

 }

Part 2

Create a derived class called Student by extending the Person class

This class must have the following private data

student ID (String)

parent name (String)

classes (ArrayList of Strings)

grades (ArrayList of Integer – 1 grade for each class – parallel array)

This class must have the following methods

default constructor

accessor and modifier methods for private data (add to classes, add to grades,

toString (creates a report card)

Write a driver class (not a main method in Student) which will create a Student and fill in the information for him.