Assignment #3

1. Explain the purpose of an instance variable  
   a. Why instance variable should be initialized in the constructor.

The class represents real-life objects in terms of state (attribute). The object’s state is stored in variables, called fields. A field can be declared as follows:

*type identifier = expression;*

Fields are also called *instance fields,* because each object contains its own copy of them. There are also *fields* that are independent ant and you want all the objects to be able to access them, but no object to have its own coy. They are known as *class fields*. They are being declared with the keyword *static*.

*static type identifier = expression;*

Fields can be initialized when declared or not. When fields are not initialized upon declaration default values are assigned to them.   
  
Example: *0* is assigned to int. *null* is assigned to String, *false* is assigned to Boolean etc.



Although fields can be initialized through direct assignment it is best practice for them to be initialized in the constructor. Fields are usually not allowed to be modified outside of the class.



1. Solve exercise 6.2 from chapter 6

*bank account balance* – **instance –** All methods in the class need to access the object’s current balance and update it. So I would make it accessible to all of them.

*amount to deposit in the account* – **parameter –** This would be passed as a parameter to a method that adds an amount to the current balance.

*account current interest rate* – **instance –** Interest rates do not change too often and this value would have to be accessed by other methods

*amount of interest earned in the last month* – **temporary –** That would be a value returned or only used in a method that calculates the earned interest

1. Design a UML Diagram for a classes Cat and Fish

class Pet  
// both animals have common attribute name  
private String name;

class Cat class Fish  
// method SayMiou // method Swim  
// method CatchFish // method Eat  
// method Sleep

Source code …  
class Pet  
  
  
  
class Fish  
  

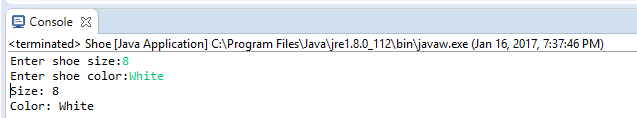

class Cat  
  


1. The Shoe Example Extended

Shoe class:

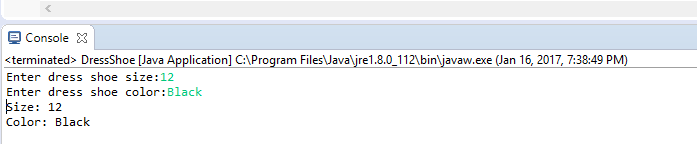


Shoe class output:



Dress class:

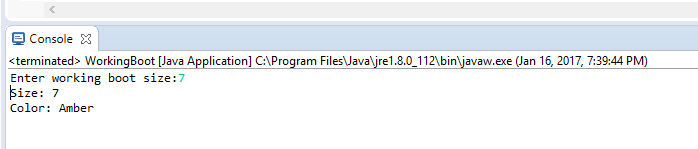


Dress class output:  


Working boot class:



Working boot class output:



5. Invoice application  
  
  


Test Invoice  
  


Test Invoice output:  
  
