ICT104

Program Design and Development

Lecture 4– A second look at Classes and Objects (cont'd)

Adopted from: Gaddis & Gaddis (2019) Starting Out with Java: From Control Structures through Objects, 7th Edition.

King's Own Institute

Focus for this week

Garbage Collection

Collaboration

Focus for this week (cont'd)

Revision For MCQ Quiz A

Focus for this week (cont'd)

A Second Look at Classes and Objects (cont'd)

- Garbage Collection
- The `finalize()` Method
- Class Collaboration
- Revision for MCQ Quiz A

Garbage Collection

- When objects are no longer needed, they should be destroyed
- This frees up the memory that they consumed
- Java handles all of the memory operations for you
- Simply set the reference to *null* and Java will reclaim the memory

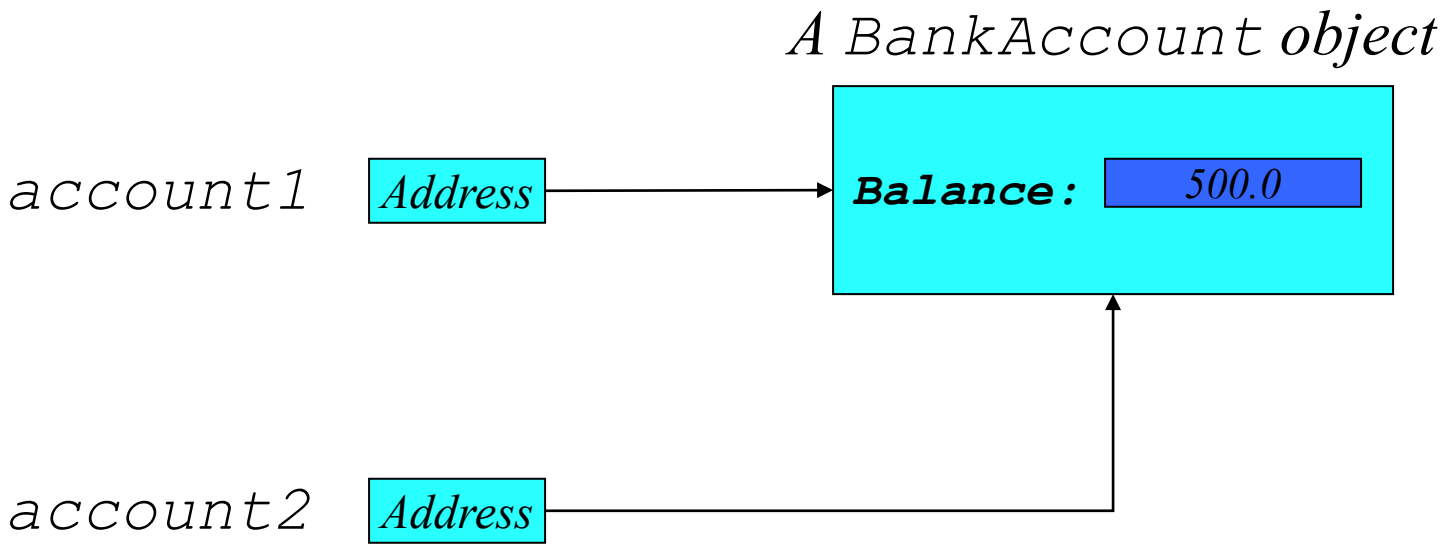
Garbage Collection (cont'd)

- The Java Virtual Machine has a process that runs in the background that reclaims memory from released objects
- The *garbage collector* will reclaim memory from any object that no longer has a valid reference pointing to it

```
BankAccount account1 = new
BankAccount(500.0);
BankAccount account2 = account1;
```

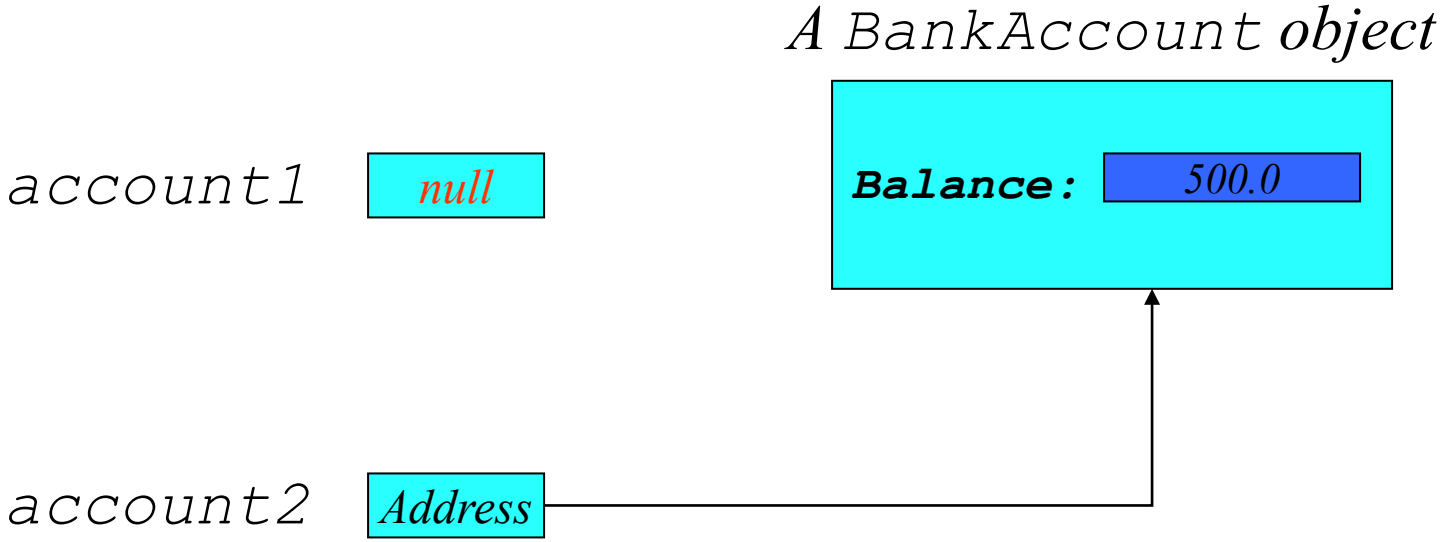
- This sets `account1` and `account2` to point to the same object

Garbage Collection (cont'd)



Here, both `account1` *and* `account2` *point to the same instance of the* `BankAccount` *class*

Garbage Collection (cont'd)

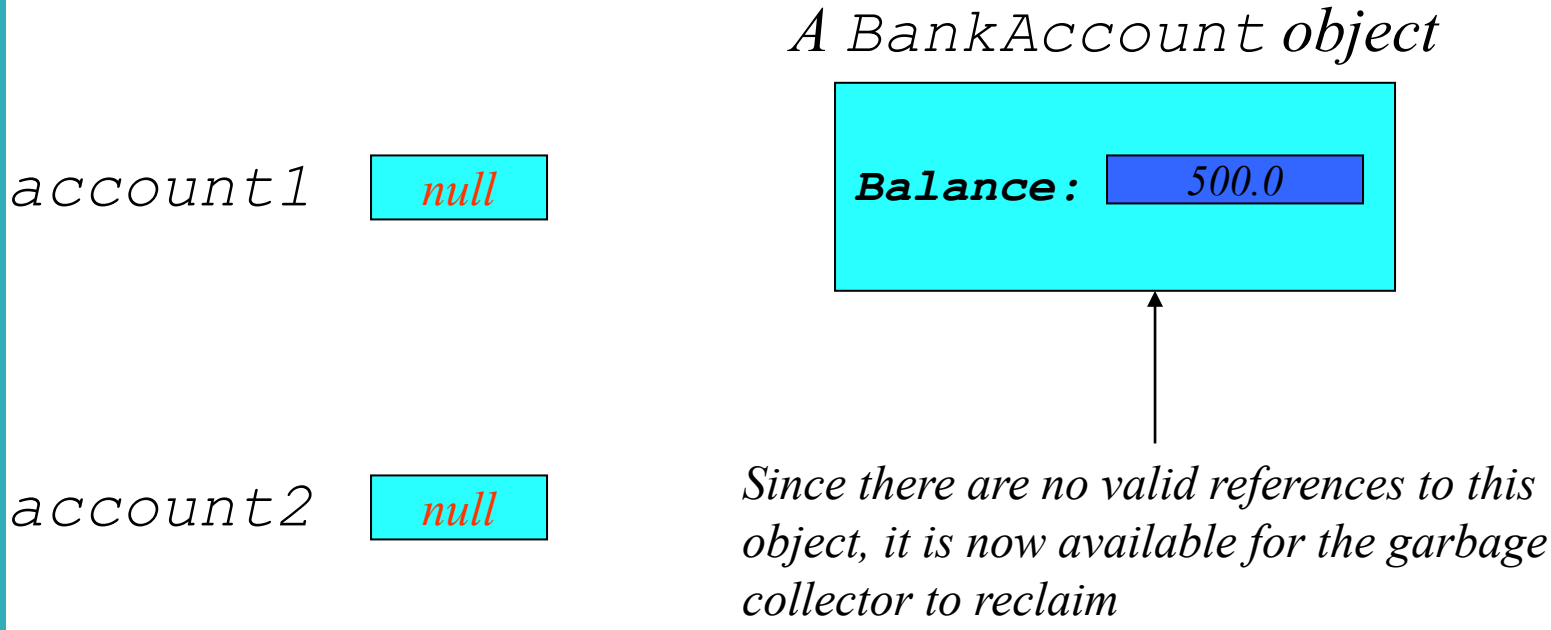


However, by running the statement:

account1 = null;

only account2 will be pointing to the object

Garbage Collection (cont'd)

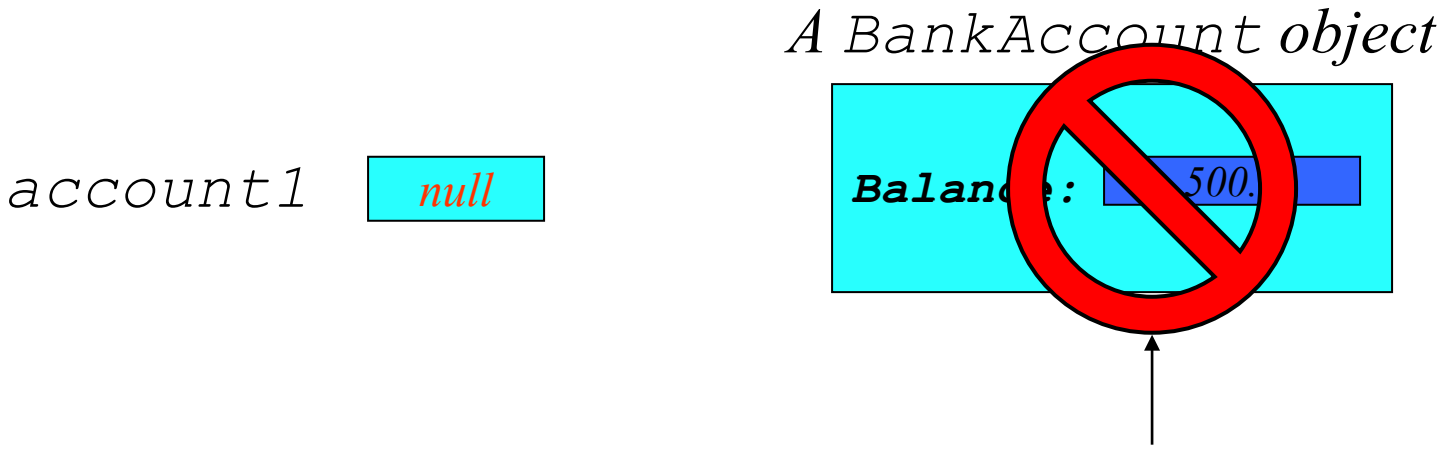


If we now run the statement

account2 = null;

neither account1 or account2 will be pointing to the object

Garbage Collection (cont'd)



The `finalize()` Method

- If a method with the signature:

```
public void finalize(){…}
```

  is included in a class, it will run just prior to the garbage collector reclaiming its memory

- The garbage collector is a background thread that runs periodically
- It cannot be determined when the `finalize()` method will actually be run

Class Collaboration

- Collaboration – two classes interact with each other
- If an object is to collaborate with another object, it must know something about the second object's methods and how to call them
- If we design a class `StockPurchase` that collaborates with the `Stock` class (previously defined), we define it to create and manipulate a `Stock` object
- See `Stock.java`
  `StockPurchase.java`

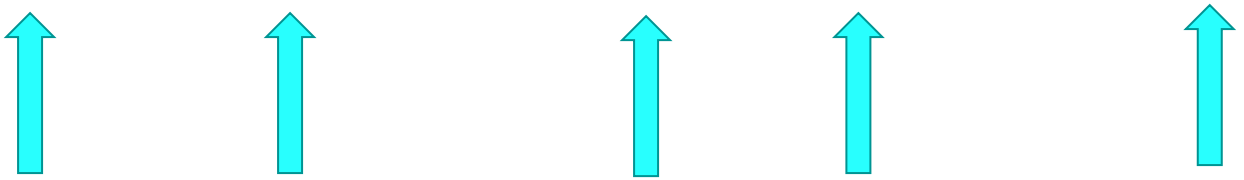
Revision

Revision For MCQ Quiz A

Question 1 Fill in the boxes:

Question 2 Label and explain each of the following parts:

```
Scanner Keyboard = new Scanner (System.in);
```



Question 3 What is the difference between:

int a

double a

Question 4 Label and explain each of the following parts:

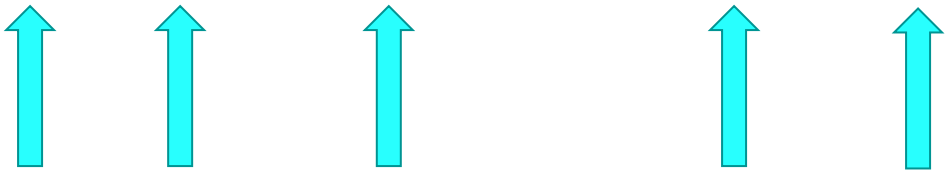
Kite reel = new Kite();

Question 5 Fill in the blank boxes of UML diagram

Class Name

Question 6 Label and explain each of the following parts:

public void setLength (double len)



Activity 1:
Question 7 What will be the output of the following program:

```
public class LengthDemo
{
   public static void main(String[] args)
   {
      Rectangle box = new Rectangle();
     System.out.println("Sending the value 10.0 " +"to the setLength method.");
      box.setLength(10.0);
      System.out.println("Done.");
   }
}
```

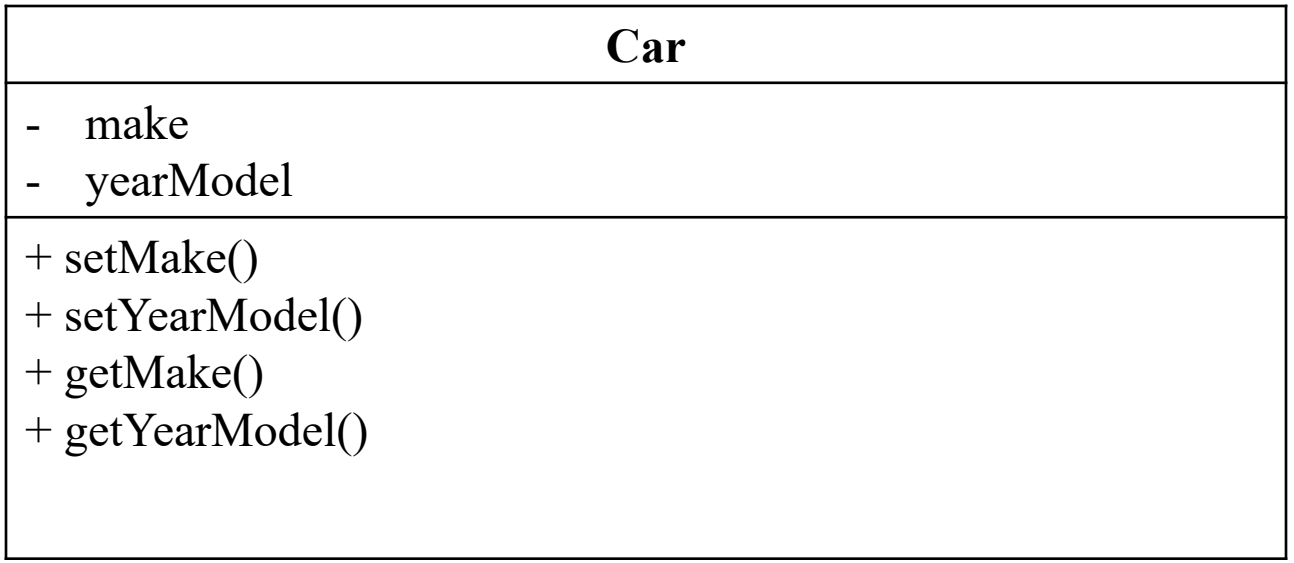
Question 8 Find the error:

CellPhone phone = new CellPhone;

Question 9 Match the following:

A	B
• Mutators	• public
• Object	• method
• Accessors	• getters
• -	• private
• +	• setters
• ()	• instance of a class

Question 10 Look at the UML diagram and answer the following:



a) What is the name of the class?

b) What are the fields?

c) What are the methods?

d) What are the private members?

e) What are the public members?

Question 11 Fill in the boxes:

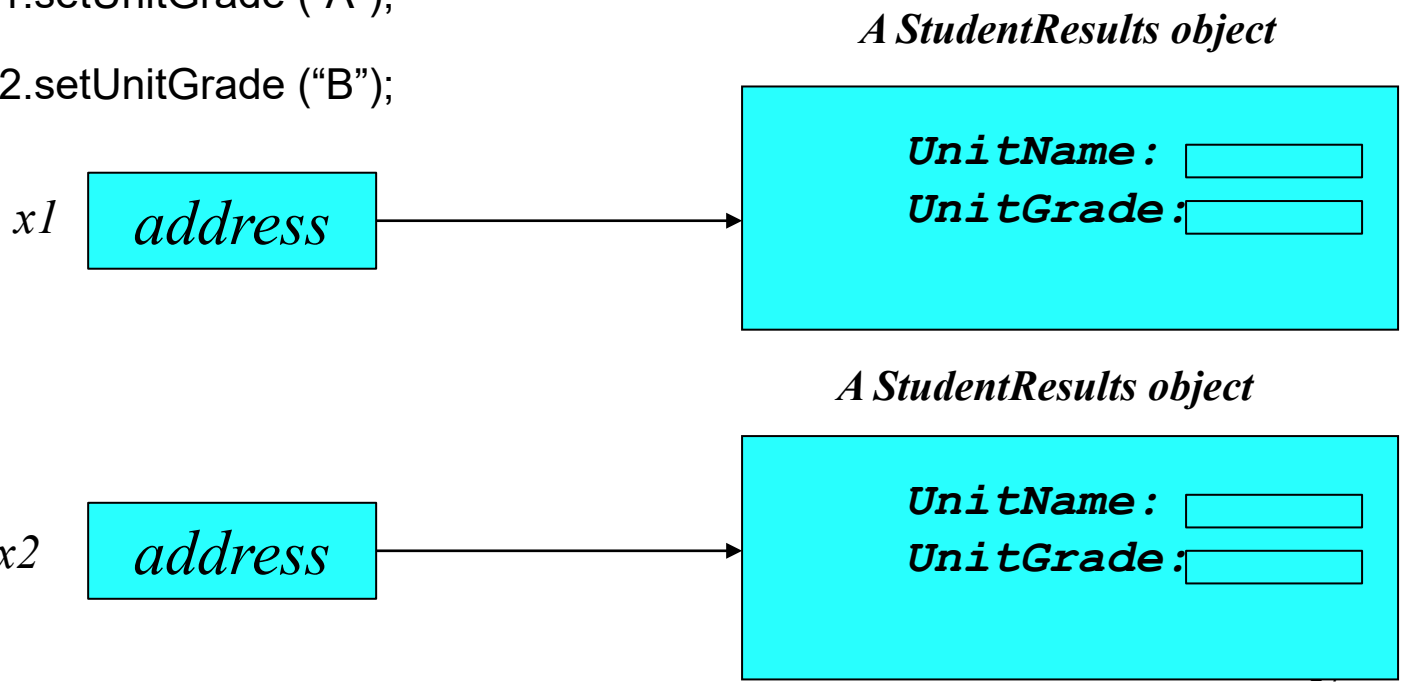
Assume that x1 and x2 are variables that reference StudentResults objects and the following statements are executed:

x1.setUnitName (“Programming Fundamentals”);

x2.setUnitName (“Discrete Maths”);

x1.setUnitGrade (“A”);

x2.setUnitGrade (“B”);



Question 12 Fill in the blank

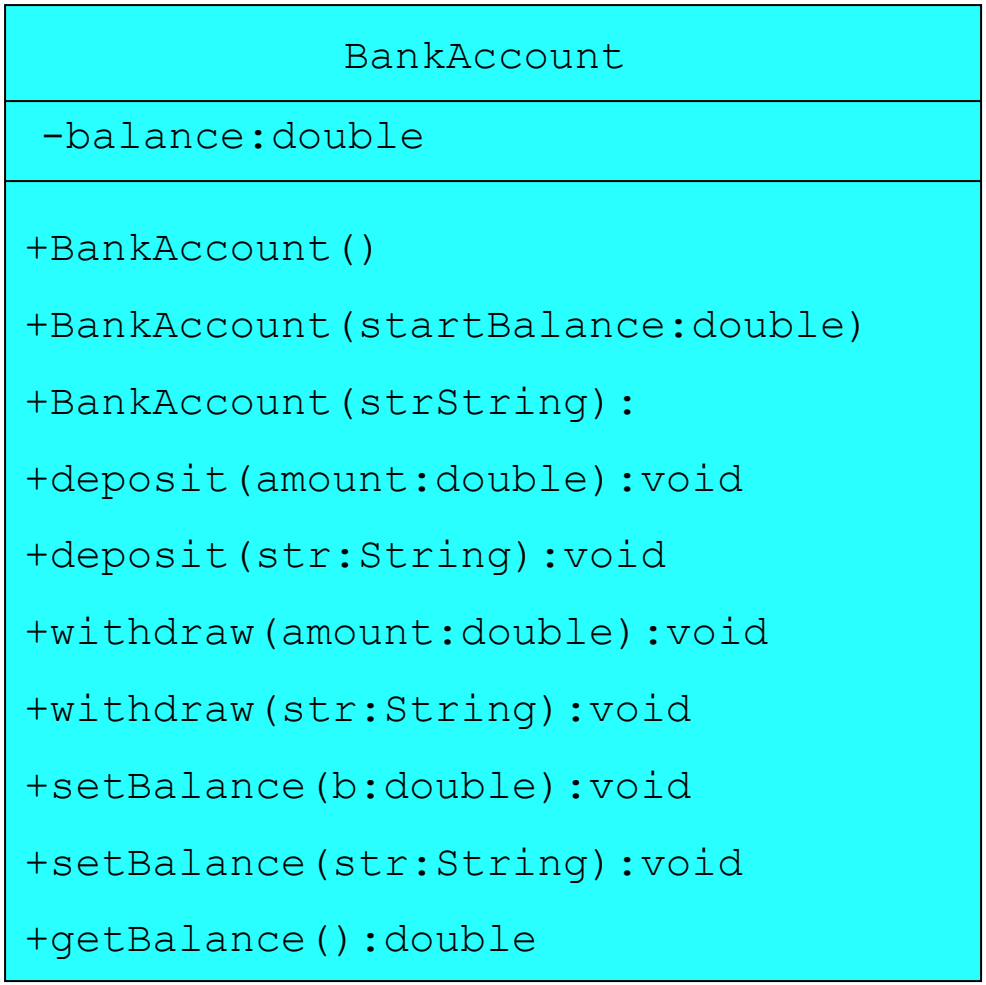
A _____ is a method that is automatically called when an instance of a class is created

Question 13 State TRUE or FALSE. If FALSE correct it.

Statement I: Constructors cannot be overloaded

Statement II: Each version of the constructor must have different parameter list

Question 14 How many methods are overloaded in the following UML diagram

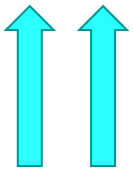


Activity 2:
Discussion question

Question 15 What do you know about java.lang package

Question 16 Label and explain each of the following parts:

```
import java.util.*;
```



Question 17 Identify Objects and Classes from the following list:

- Car
- BMW
- Employee
- Address
- Customer
- Manager
- Temperature
- Mercedes
- Thursday
- Month

Question 18 Find the error in the following class:

```
public class MyClass
{
        private int x;
        private double y;
        public void MyClass (int a, double b)
        {
              x = a;
              y = b;
        }
}
```

Question 19 Find the error in the following class:

```
public class TwoValues
{
        private int m n;
        public TwoValues()
        {
                m = 0;
        }
        public TwoValues(int m, int n)
        {
                m = 0;
                m = 0;
        }
}
```

Activity 3:

Question 20 Look at the following statement:

```
enum Flower { ROSE, DAISY, ZINIA, JASMINE,
  SUNFLOWER }
```

a) What is the name of the data type?

b) What is the ordinal value for the enum constant ROSE? For JASMINE?

Question 21 MCQ

Java automatically stores this value in all uninitialized static member variables of type int:

A) 0

B) -1

C) null

D) false

Question 22 MCQ

The whole-part relationship created by object aggregation is more often called:

A) a *has a* relationship

B) an inner class relationship

C) an extra class relationship

D) an inside class relationship

Question 23 MCQ

When the “this” variable is used to call a constructor:

A) it must be the first statement in the constructor making the call

B) it must be the last statement in the constructor making the call

C) it can be anywhere in the constructor making the call

D) you cannot use the this variable in a constructor call

Question 24 MCQ

To compare two objects in a class:

A) use the == operator, e.g., object1 == object2

B) write a method to do a byte-by-byte compare of the two objects

C) write an equals method that will make a field by field compare of the two objects

D) Since objects consist of several fields, you cannot compare them

Activity 4:

Question 25 Draw UML Diagram and write output for already studied classes and programs

Summary of today's lesson

A Second Look at Classes and Objects (cont'd)

- Garbage Collection
- The `finalize()` Method
- Class Collaboration
- Revision for MCQ Quiz A

Activity 5:
Reflection Exercise

List any four concepts you have learnt in today's lesson

Activity 6: Homework Exercise

Write a program using either Eclipse or NetBeans to implement any one concept you have learnt in today's lesson