BroncolD: 014758458

Last Name: Siwiecki First Name: Joseph

Github Repo: https://github.com/JRSiwiecki/CS3560.03-Assignment2

CS 3560.03 Assignment 2

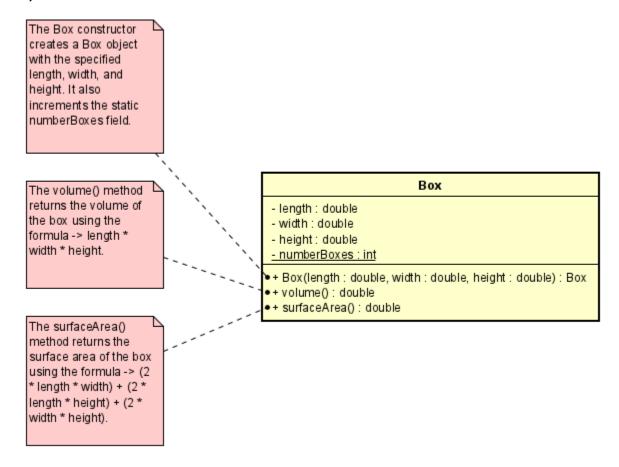
1.A)

```
package Q1;
        private String name;
        private String major;
        private double gpa;
        public Student()
             name = null;
             major = null;
             gpa = 0.0;
        public Student(String name, String major, double gpa)
18
             this.name = name;
             this.major = major;
             this.gpa = gpa;
        public String getName()
             return name;
         public void setName(String name)
             this.name = name;
        public String getMajor()
             return major;
         public void setMajor(String major)
40
             this.major = major;
41
```

```
public double getGpa()
   return gpa;
public void setGpa(double gpa)
   this.gpa = gpa;
@Override
// Created with Eclipse's toString generator
public String toString()
   return "Student [name=" + name + ", major=" + major + ", gpa=" + gp
public boolean equals(Student student)
   if (!this.name.equals(student.name))
       return false;
    if (!this.major.equals(student.major))
       return false;
   return true;
```

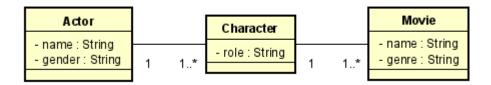
1.B)

```
package Q1;
    public class StudentTest
        public static void main(String[] args)
            // One with arguments, other with no arguments
            Student student1 = new Student(name: "John", major: "CS", gpa: 3.5);
            Student student2 = new Student();
            // Update student2's name, major, and gpa
            student2.setName(name: "Mary Ann");
            student2.setMajor(major: "CE");
            student2.setGpa(gpa: 3.3);
L6
8
            System.out.println(student1.toString());
19
20
            System.out.println(student2.toString());
```

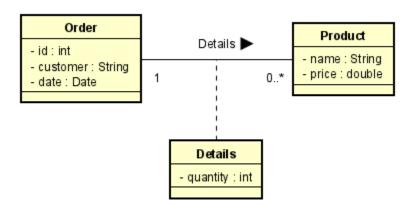


2.B) The practical implications of making length, width, and height instance fields but numberBoxes as a static field is that each Box object is allowed to have its own set length, width, and height while still being able to track how many boxes there are in total, even though each Box may have different values for its instance fields.

3.A)



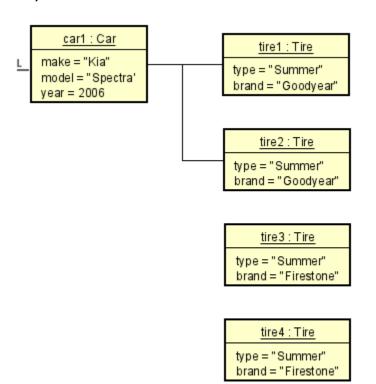
3.B)



4.A)



4.B)



```
public class Student

private String name;
private Transcript transcript;

public Student(String name, Transcript transcript)

public Student(String name, Transcript transcript)

this.name = name;

this.transcript = transcript;

}

}
```

```
public class Transcript

private String course;
private double grade;

public Transcript(String course, double grade)

this.course = course;
this.grade = grade;

}
```

```
public class Book

private String name;
private String author;
private Course course;

public Book()

name = null;
author = null;
course = null;

course = null;

}
```

```
import java.util.ArrayList;

public class Course

{
  private int code;
  private ArrayList<Book> books;

public Course(int code, ArrayList<Book> books)

  {
  this.code = code;
  this.books = books;

}
```

```
import java.util.ArrayList;

public class Team

function

public class Team

private int code;
private ArrayList<Player> players;

public Team()

code = 0;
players = new ArrayList<Player>();

players = new ArrayList<Player>();

players = new ArrayList<Player>();
```

```
public class Player

{
    private String name;
    private boolean expert;

    public Player(String name, boolean expert)

    {
        this.name = name;
        this.expert = expert;

}

10     }

11     }

12
```

```
import java.util.ArrayList;
     public class Dog
         private String breed;
         private String name;
         public ArrayList<Paw> paws;
         public Dog()
11
             breed = null;
12
             name = null;
             paws = new ArrayList<Paw>();
13
14
             paws.add(new Paw(position: 1, this));
15
             paws.add(new Paw(position: 2, this));
16
17
             paws.add(new Paw(position: 3, this));
             paws.add(new Paw(position: 4, this));
18
19
21
```

```
public abstract class Employee

private String name;
private int hours;

public Employee(String name, int hours)

this.name = name;
this.hours = hours;

public double calculateSalary()

return hours * 20;

return hours * 20;

}
```

```
public class Professor extends Employee

private String field;

public Professor(String name, int hours, String field)

super(name, hours);
this.field = field;

public double calculateSalary()

return super.getHours() * 30;
}
```

```
public class Staff extends Employee

private int role;

public Staff (String name, int hours, int role)

super(name, hours);

this.role = role;

}
```

```
public interface SaleableItem
public void sellCopy();

public void sellCopy();

public void sellCopy();

public void sellCopy();
```

```
public class Magazine implements SaleableItem

public Magazine()

public Magazine()

@Override
public void sellCopy()

System.out.println(x: "Selling a magazine.");

}

}
```

```
public class Ticket implements SaleableItem

public Ticket()

public Ticket()

outpublic Ticket()

outpublic Ticket()

system.out.println(x: "Selling a ticket.");

system.out.println(x: "Selling a ticket.");
```

```
import java.util.ArrayList;

public class Person

for private String name;
private ArrayList<watch> watches;

public Person(String name)

function this.name = name;

function import java.util.ArrayList;

public class Person

function import java.util.ArrayList;

public Person

function import java.util.ArrayList;

function import
```

```
import java.util.ArrayList;

public class Movie

frivate String name;
private String genre;

private ArrayList<watch> watches;

public Movie(String name, String genre)

frivate ArrayList<watch> watches;

public Movie(String name, String genre)

frivate ArrayList<watch> watches;

public Movie(String name, String genre)

frivate ArrayList
public Movie(String name, String genre)

frivate String name;
private String genre;

private String genre;

private String genre;

private ArrayList
public Movie(String name, String genre)

frivate String genre;

private String genre;

private String genre;

private ArrayList
private ArrayList
```

```
public class watch

private Person person;
private Movie movie;
private int rating;

public watch(Person person, Movie movie, int rating)

this.person = person;

this.movie = movie;
this.rating = rating;
person.addWatch(this);
movie.addWatch(this);
}
```

```
public class Worker

private String name;
private double hourlyRate;

public Worker(String name, double hourlyRate)

this.name = name;

this.hourlyRate = hourlyRate;

}
```

```
public class Payroll

public void processPayments(Worker worker)

{
    System.out.println("Payment processed for worker " + worker.getName());
}

}
```