<Mangesh Raut mbr63>

<04/27/2022>

Week5Meet - 10 pts

Turn in on BBL as soon as complete, but before end of day Friday following the lecture.

We are changing it up this time!

Answer these questions as we progress through the meeting.

1. Please circle/highlight the nouns in the following program requirements:  
     
   The Great Lakes Shipping company ships pallets of goods from an origin to a destination. Sometimes the pallets get loaded onto boats and shipped across a lake, but many pallets get loaded onto trucks for some portion of their trip. The company is making use of cutting-edge tech and has invested in self-driving electric vans for last mile delivery. They need to be able to track pallets, assign them to a boat or truck or van, load, and unload pallets. Each boat has a captain, and each truck has a driver. In addition, with fuel prices fluctuating, they need to track fuel on the boats and trucks. All vehicles that use fuel, use diesel.
2. Make a list with the most likely candidates for classes at the top of the list. Who did you work with?  
   Dayo, Akshay, Sharon
3. The keyword this. Take a look at the toString() method in the Pallet class:  
     
   public String toString(){  
    String label = "PalletID: " + this.getID() + "\nOrigin: " + this.getOrigin() + "\nDestination: " + this.getDestination() + "\nDim: " + this.getDimensions();  
    return label;  
    }  
   In your own words, why are we using the this keyword here?  
   this keyword is used to get the reference of variables if user enter the palletID this.getID which we declare in constructor that will get the id and print it. this keyword is reference variable which we use to get the value in that variable which we declare in constructor.
4. Read the Fuelable interface:  
   public interface Fuelable {  
    public void addFuel(int fuel);  
    public int getFuel();  
   }  
   Then read the implementations of each method in Boat and Truck:  
   public void addFuel(int fuel){  
    fuelGallons += fuel;  
   }  
   public int getFuel(){  
    return fuelGallons;

}

If the boat can hold 100 gallons of diesel, how can you improve the implementation of addFuel() so it is impossible to add fuel over the 100 gallon limit?

@Override

**public** **void** addFuel(**int** fuel) {

**if** (fuelGallons + fuel <= 100) {

fuelGallons += fuel;

}

}

@Override

**public** **int** getFuel() {

**return** fuelGallons;

}

If Truck has two fuel tanks of 50 gallons each(a not uncommon occurrence) how could you improve the implementation code for addFuel() for the truck?  
**public** **void** addFuel(**int** fuel) {

**int** tankArem = 50 - **this**.tankA;

**if** (tankArem > 0) {

**if** (fuel - tankArem > 0) {

**this**.tankA += tankArem;

fuel -= tankArem;

} **else** {

**this**.tankA += fuel;

fuel = 0;

}

}

**int** tankBrem = 50 - **this**.tankB;

**if** (fuel > 0 && tankBrem > 0) {

**if** (fuel - tankBrem > 0) {

**this**.tankB += tankBrem;

fuel -= tankBrem;

} **else** {

**this**.tankB += fuel;

fuel = 0;

}

}

}

@Override

**public** **int** getFuel() {

**return** **this**.tankA + **this**.tankB;

}

1. For the load and unload methods, is the pallet object changed in any way?  
   public boolean load(Pallet p){  
    boolean fits;  
    if ((capacity - p.getVolume()) > 0){  
    fits = cargo.add(p);  
    capacity -= p.getVolume();  
    }  
    else {  
    fits = false;  
    }  
    return fits;   
    }  
      
    public Pallet unload(Pallet p){  
    int loc = cargo.indexOf(p);  
    int vol = p.getVolume();  
    capacity += vol;  
    return cargo.remove(loc);   
    }  
   pallet p is same object for both the methods as you can see load method having Boolean which return the boolean value. And upload method is having pallet which is accessing that pallet class which the parameter pallet p.
2. Develop your white-box test cases for your chosen class. Who did you work with?  
   I work with Truck class, and I work with Dayo, Akshay, and Sharon I discussed with them and we use all the methods and print the result.
3. What is the name of your tester file? Submit the .java file  
   TruckTester.java
4. What are the results of your test cases? How do you KNOW you tested every line of code in your class?

I use every method which is available in Truck class using the truck class object I access them and print the results using the system.out.println.

100

Akshay

50

100

0

true

PalletID: 1001

Origin: NA

Destination: NA

Dim: H: 0 W: 0 D: 0

true

PalletID: 1002

Origin: NA

Destination: NA

Dim: H: 0 W: 0 D: 0

1

Gowtham

30

200

0

true

PalletID: 1001

Origin: NA

Destination: NA

Dim: H: 0 W: 0 D: 0

true

PalletID: 1002

Origin: NA

Destination: NA

Dim: H: 0 W: 0 D: 0

500

Mangesh

100

200

1000

700

Dayo

75

300

2000

Reflect on your learning and your needs. After this class meeting, what topics do you feel like you learned and what topics do you feel like you need more information on to learn?

I want to learn about the interface like it is used for more classes and we use implements. Abstract class use extends and it also has abstract methods and values or variables.