Interface

23-06-2020

Practice Problems: Working with Interfaces

1) Assume you want to capture shapes, which can be either circles (with a radius and a color) or rectangles (with a length, width, and color). You also want to be able to create signs (to post in the campus center, for example), each of which has a shape (for the background of the sign) and the text (a String) to put on the sign.

   Create classes and interfaces for circles, rectangles, shapes, and signs.

- In order to make signs, we need to make shapes that are large enough to fit the text for the sign. Write a method on shapes fitsText that takes the text as an argument and determines whether the length of the text is shorter than the length/radius of the shape. You can get the length of a String by calling the method length on the string.

- Practice making errors with your shapes: omit the interface, make typos in the names of the fitsText method, and so on. This will help you get familiar with the errors messages that Java produces in various situations.

2.) Here's a basic setup of classes and an interface for vehicles. Use this to answer the following question about interfaces.

```
interface Ivehicle
{
         // indicate how much a basic tune-up costs
        public double tuneUpCost();

         // determine whether vehicle can hold given num of passengers
        public boolean canCarry(int numPassengers);
}

 class Car implements Ivehicle
 {
         int mileage;
         int year;
         int numDoors;

  // constructor goes here

  // indicate whether car was built before given year
  boolean builtBefore(int otherYear) {
    return this.year < otherYear;
  }
 }

 class Bicycle implements Ivehicle
{
  int mileage;
  int numGears;

  // constructor goes here
 }
```

- What methods do you need to add to each of Car and Bicycle to get this code fragment to compile (setting aside the missing constructors)?
- Does having a class implement an interface change how you write its constructor?
- Should builtBefore be added to the IVehicle interface? Why or why not?
- In the Examples class, you want to define a Bicycle as follows:
- _____ newKidsBike = new Bicycle(0, 1);
- Which types can you use in the blank line (without causing a compile error just with this single line)?
- Assume you defined a Car object as follows:
- IVehicle oldCar = new Car(200000, 1995, 2);
- Which methods can you call on oldCar, given the type that you gave it?
- Assume you defined a Car object as follows:
- Car oldCar = new Car(200000, 1995, 2);
- Which methods can you call on oldCar, given the type that you gave it?