Lesson 7 (Chapter 7) Programming Assignment

    a. Suppose you have a Shape hierarchy as follows:

```
                    ┌─────────────────────┐
                    │       Shape         │
                    ├─────────────────────┤
                    │                     │
                    ├─────────────────────┤
                    │ +setLocation()      │
                    │ +getLocation()      │
                    │ +display()          │
                    │ +fill()             │
                    │ +setColor()         │
                    │ +undisplay()        │
                    └─────────────────────┘
```

Point	Line	Rectangle
+display()	+display()	+display()
+fill()	+fill()	+fill()
+undisplay()	+undisplay()	+undisplay()
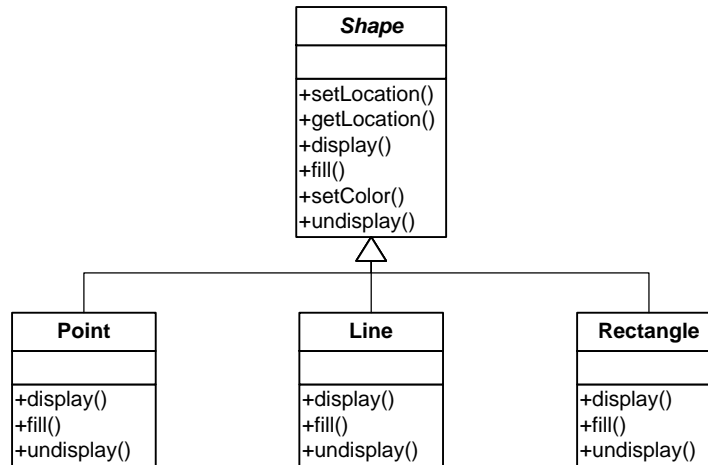
Now you want to add a Circle class, but you hear that someone else has already created one, named XXCircle, and you just want to reuse it. Its interface is slightly different though:

```
┌─────────────────┐
│    XXCircle     │
├─────────────────┤
│                 │
├─────────────────┤
│ +setLocation()  │
│ +getLocation()  │
│ +displayIt()    │
│ +fillIt()       │
│ +setItsColor()  │
│ +undisplayIt()  │
└─────────────────┘
```
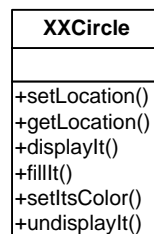
Draw a UML diagram that adapts this class to your hierarchy. Write skeleton code that just uses print statements for all the methods and write a test driver that creates a sequence of Shapes and displays all of the shapes it contains, verifying that your Circle class nicely coexists with the other types of Shape. Submit your UML diagram, working code and a screen shot of your program's output.

    b. A web application uses a database to get Employee data. The application could use the database API directly, but this would result in a number of low-level SQL calls scattered throughout the code (setting the connection, building queries, extracting results, building Employee objects, etc.). Explain how you can design the application so that it is not directly dependent on the database used, and so getting and setting employee data won't be such a pain.