

6.092 Assignment 6: Modeling a Social Network

Your assignment is to use your current knowledge about Java, OOP, and Software Design to model certain aspects of a simple Social Network. These specifications are intentionally loose in order to give you creative freedom and to test your problem-solving skills.

Part 1: `SocialEntity` Interface

The `SocialEntity` is the basic model for anyone or anything in the social network. Every `SocialEntity` has a name (`String`) and a unique ID (`long`) that can be accessed only through its `getName()` and `getId()` methods. *Note that fields in interfaces are automatically final and methods are automatically abstract!*

Part 2: Subclasses of `SocialEntity`

Types of `SocialEntity`s:

- `Person` - has acquaintances : `Person[]`, networks : `Network[]`, and a location : `String`
- `Group` - members : `Person[]`
- `Network` - members : `Person[]`
- `Band` - fans : `Person[]`, description : `String`, location : `String`

Write the appropriate constructors and methods for each class. Be mindful of how and when you initialize or assign each class its fields.

Part 3: `YourFace` Class

The `YourFace` class is our entryway into the social network. That is, it should have a `main` method, which should create seven `Person`s (each with three acquaintances and any number of `Network`s), two `Group`s (each with one or more member), two `Network`s (each with one or more member), and two `Band`s with any number of fans. After creating and populating your social network, you should print out all of the properties of each `Person` in the network.

Remember to use encapsulation techniques, follow other OOP principles mentioned in lecture, and add comments where the code is doing something that isn't obvious! Email any questions to the staff, and send your solutions in an archive (jar/zip/tar/rar) by **4pm before the next class**.

(Information about jar files can be found here:

<http://java.sun.com/docs/books/tutorial/deployment/jar/basicsindex.html>)