

**1. What consistency checks should be applied between sequence diagrams and class diagrams in OOD?**

Consistency checks that should be applied are:

- number and name of objects in all sequence diagrams must match with classes in class diagrams
- number and name of methods in a class must match with all messages sent to its objects in all sequence diagrams

**2. Discuss if generalization relationship between classes can be determined by using information available directly in sequence diagrams. Use an example to explain how generalization relationship between classes can be established in a class diagram.**

Generalization relationship between classes cannot be determined by using only information in sequence diagrams. One would need more information from the domain to determine generalization. Generalization relationship between classes can be established in a class diagram by using a hollow arrowhead going from the child class to the parent class. The name of the parent class is italicized and a tagged value {abstract} may be used to indicate that the class is abstract. For example, if class "circle" is a subclass of superclass "Shape", we italicize "Shape" and draw a hollow arrowhead from "circle" to "shape".

**3. Use Visual Studio to reproduce the sequence diagram for the "Login" use case as illustrated below. For simplicity, you may choose to ignore most of the commentary notes and the object for class "DALUserInfo". However, at least one commentary note must be included in the reproduced diagram where the initials "XY" need to be replaced by your initials. In addition, all interactions involved with the object for "DALUserInfo" need to be eliminated should you choose not to include the object in the sequence diagram.**

**4. Design a class diagram according to the whole sequence diagram below for the "Login" use case, and the fact that parameter "UserInfo" in the sequence diagram is a type itself. Complete the following steps to create the class diagram with Visual Studio.**

- a. Create classes from the sequence diagram;
- b. Create a class diagram, and drop the classes into the diagram;
- c. Complete the diagram by adding adequate attributes, relationships, types, and visibility.