Objective

Unified Modeling Language (UML)

Tutorial : UML

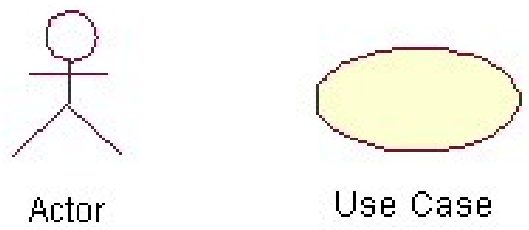
http://www.tutorialspoint.com/uml/index.htm

History

- The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems.
- The UML uses mostly graphical notations to express the design of software projects.
- Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software.

Use Case Diagrams

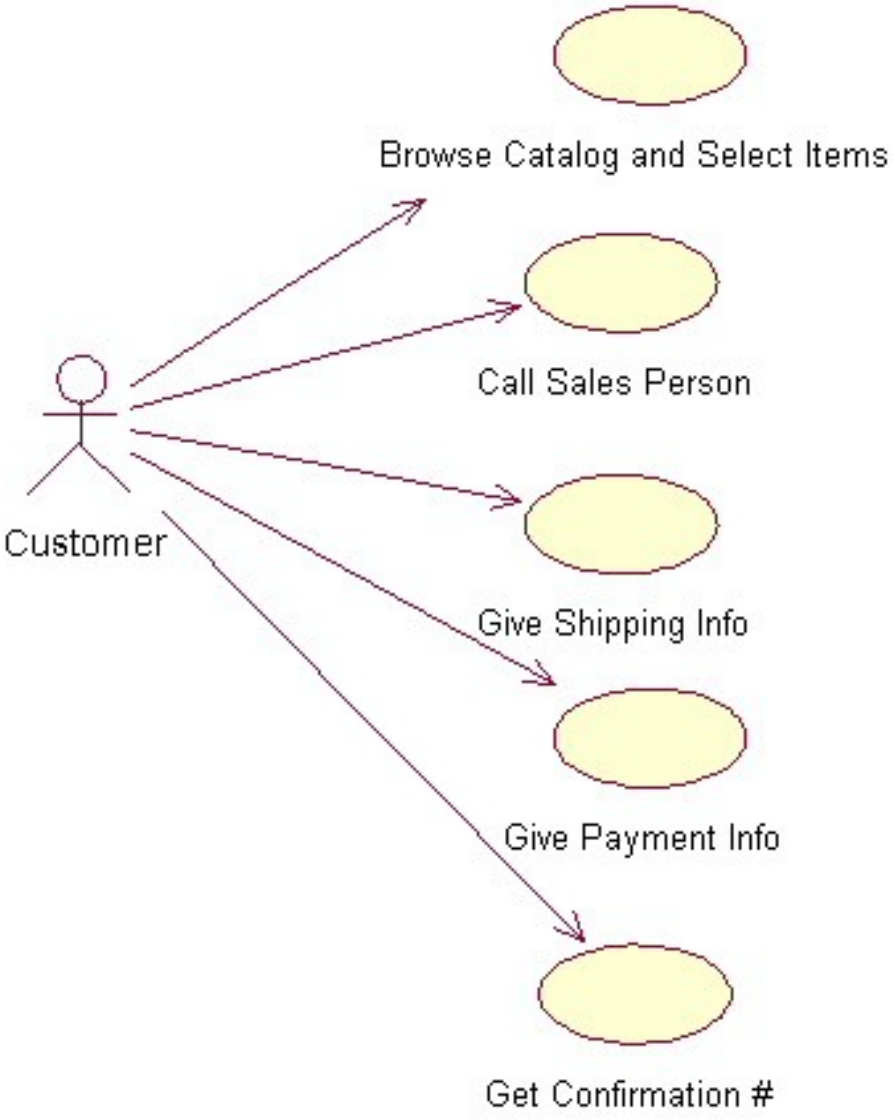
A use case is a set of scenarios that describing an interaction between a user and a system.



An actor is represents a user or another system that will interact with the system you are modelling.

Use cases are used in almost every project. They are helpful in exposing requirements and planning the project. During the initial stage of a project most use cases should be defined, but as the project continues more might become visible.

Example of Use Cases Diagrams

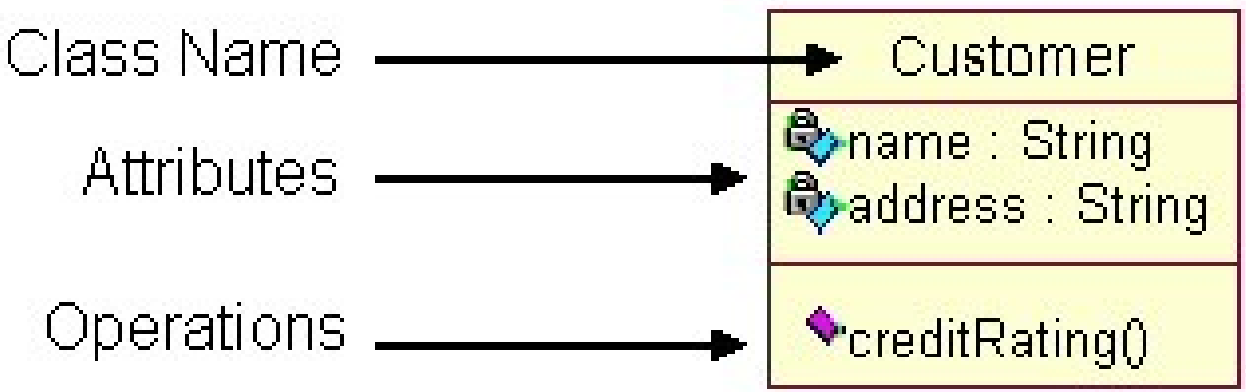


Class Diagrams

Class diagrams are widely used to describe the types of the classes in a system and their relationships.

Classes are composed of three things:

- Name
- Attributes
- Operations



Visibility

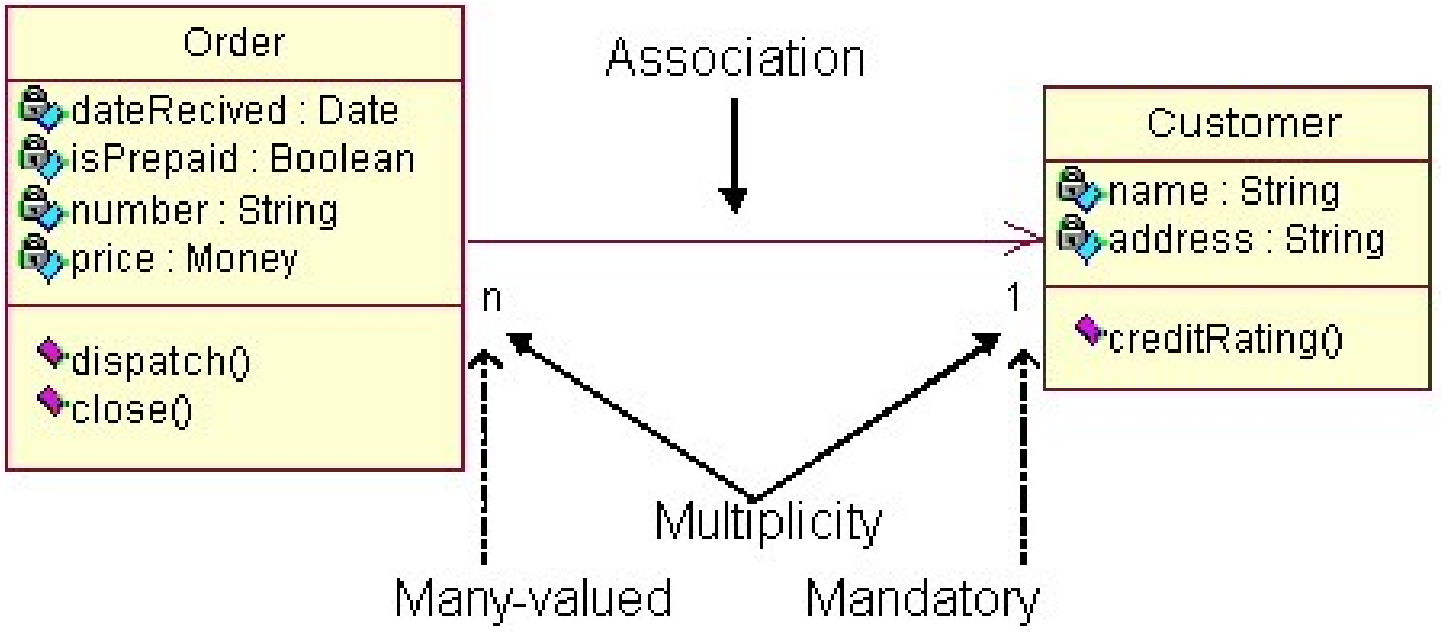
+ Public
- Private
Protected
~ Package

Show static members underlined.

Associations (Class Diagrams)

Class diagrams also display relationships such as

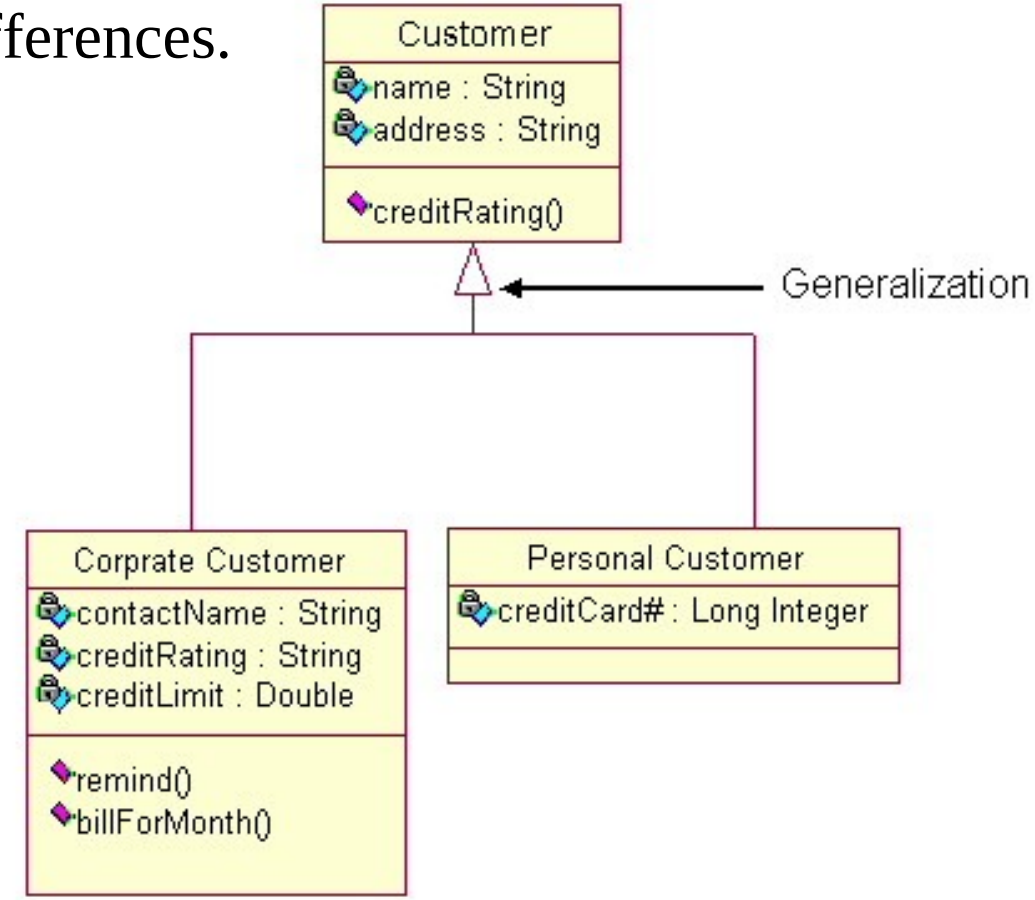
- Inheritance
- Associations.



The association shows the relationship between instances of classes.

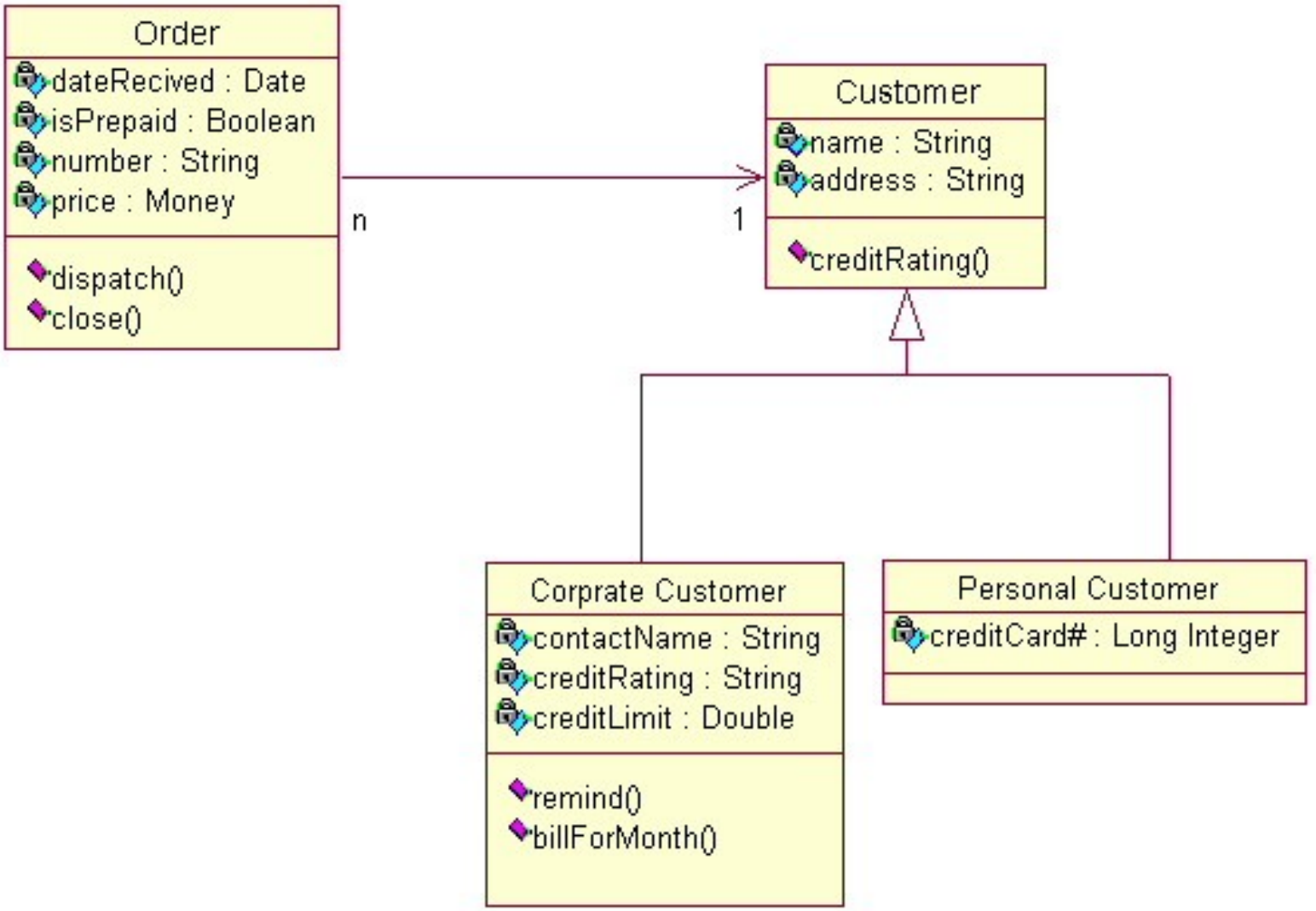
Generalization (Class Diagrams)

A generalization is used when two classes are similar, but have some differences.



Class diagrams are used in nearly all Object Oriented software designs. Use them to describe the Classes of the system and their relationships to each other.

Example of Class Diagram



Bonus

Interaction diagrams

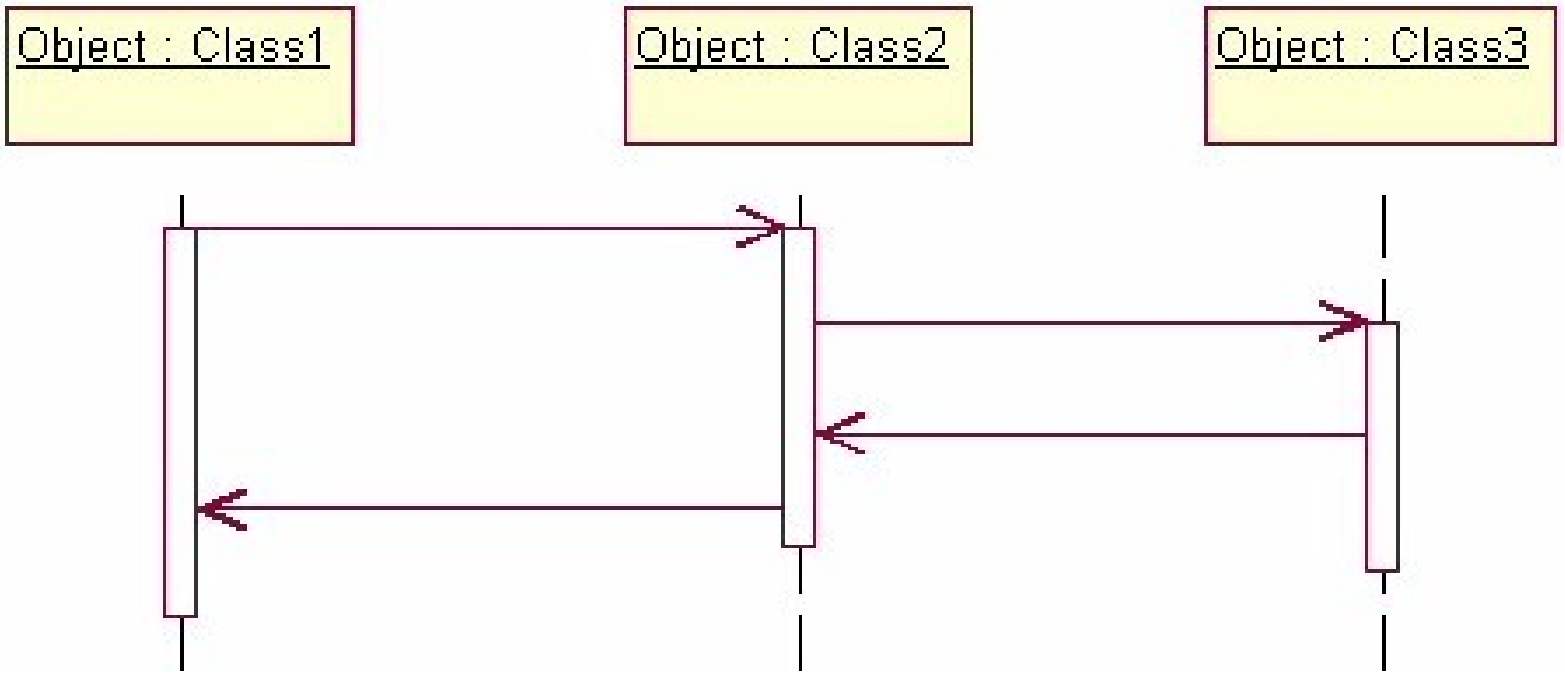
- Interaction diagrams model the behavior of use cases by describing the way groups of objects interact to complete the task. The two kinds of interaction diagrams are **sequence** and **collaboration** diagrams.

- Interaction diagrams are used when you want to model the behavior of several objects in a use case.

Bonus

Sequence diagrams (Interaction diagrams)

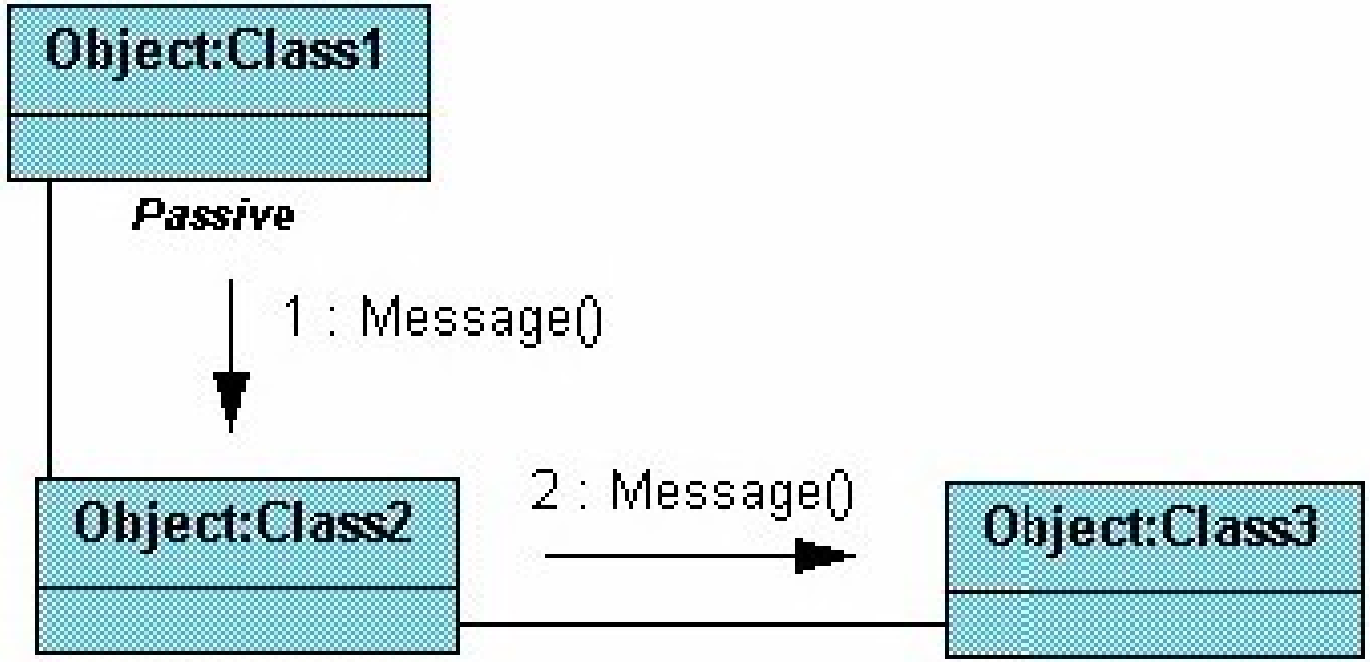
Sequence diagrams demonstrate the behaviour of objects in a use case by describing the objects and the messages they pass. the diagrams are read left to right and descending.



Bonus

Collaboration diagrams (Interaction diagrams)

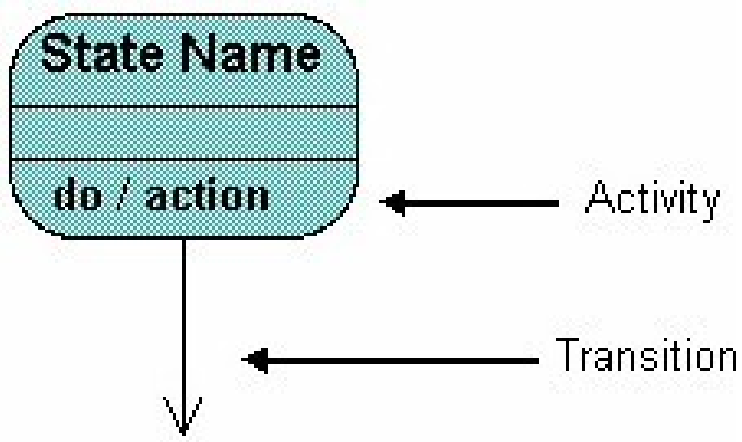
- They show the relationship between objects and the order of messages passed between them.
- The objects are listed as icons and arrows indicate the messages being passed between them.
- The numbers next to the messages are called sequence numbers.



Bonus

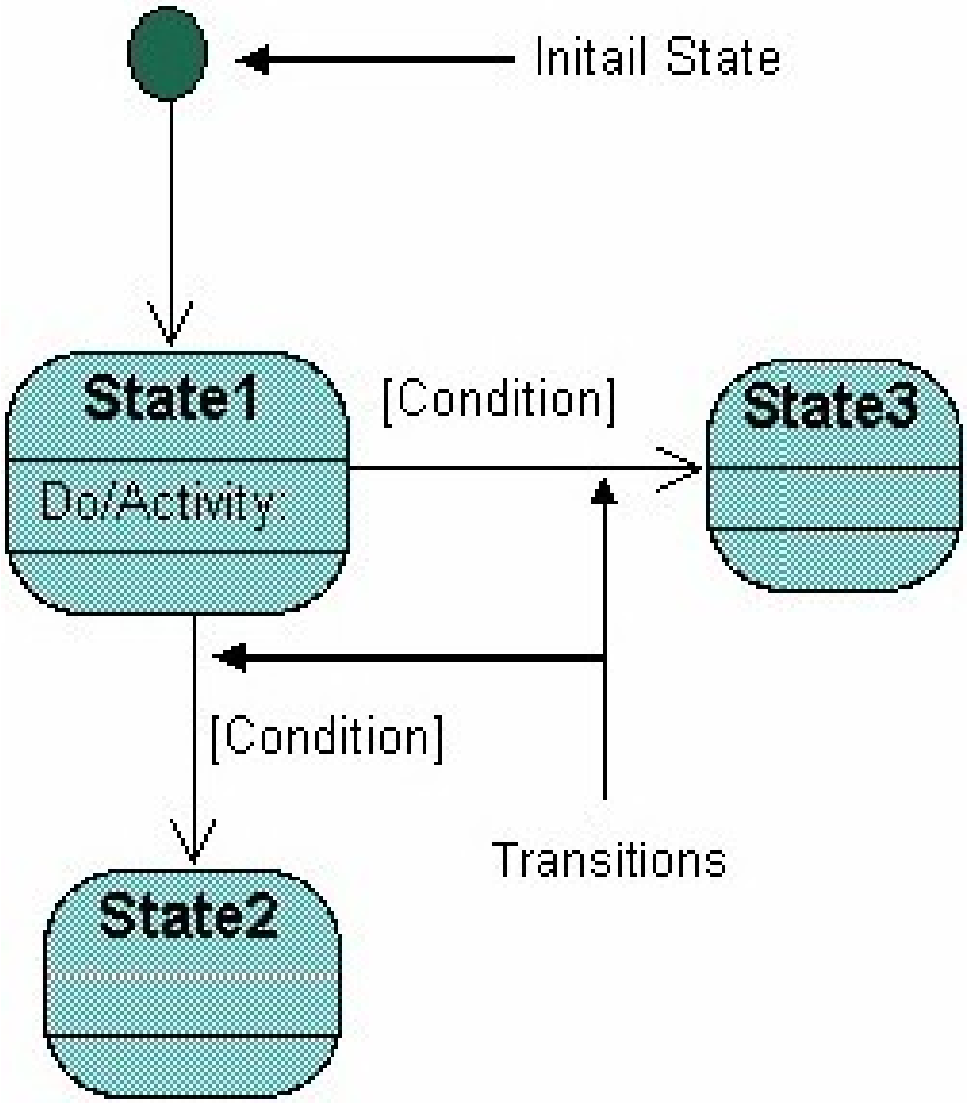
State diagrams

- State diagrams are used to describe the behavior of a system.
- State diagrams describe all of the possible states of an object as events occur.
- Each diagram usually represents objects of a single class and track the different states of its objects through the system.
- Use state diagrams to demonstrate the behavior of an object through many use cases of the system.
- Only use state diagrams for classes where it is necessary to understand the behavior of the object through the entire system.



Bonus

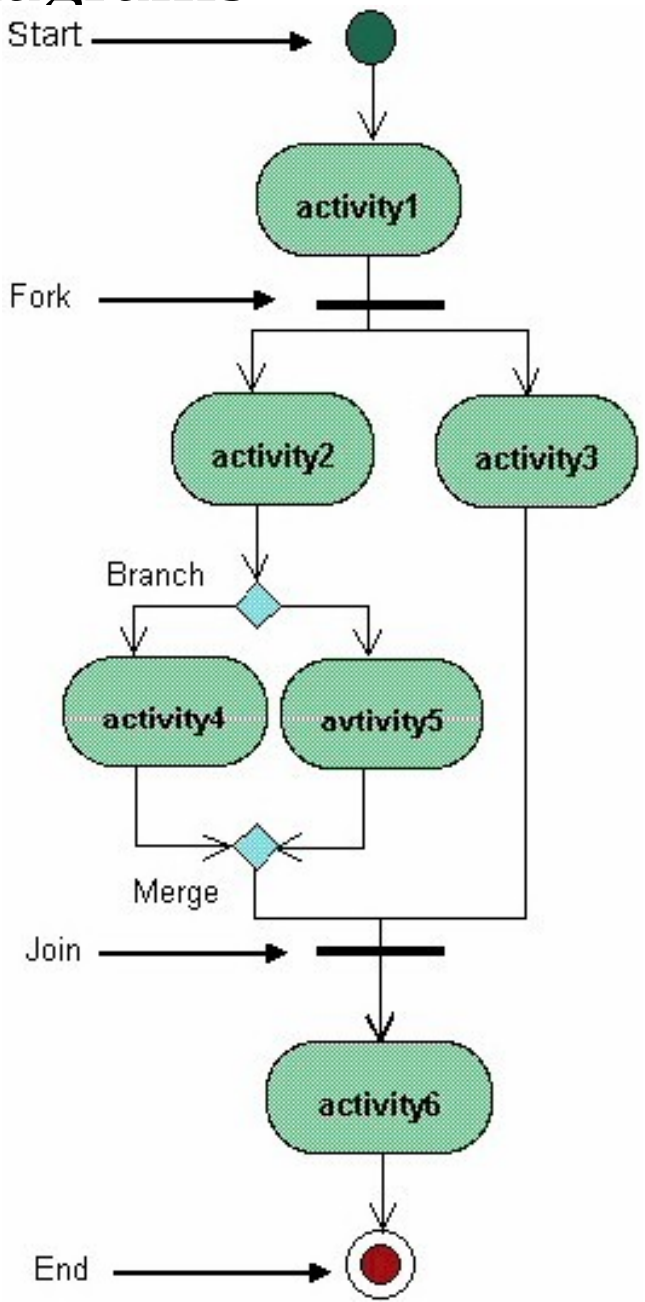
Example of State diagrams



Bonus

Activity diagrams

Activity diagrams describe the workflow behaviour of a system.



Bonus

Example of Activity diagrams