CLASS DIAGRAMS

As it sounds, a UML class diagram is used to map out and to give a visual representation for a class and how it is to be implemented in a program. This will include any methods in relation to the class and attributes that it possesses. The class diagram is the main essence of object-oriented modeling.

Example of a class in a class diagram:



Above, you can see the basic makeup of a class in a class diagram. Here, you can see the BankAccount class name at the top in the top box and then in the box below that, you can see the attributes. In this case, there is the owner attribute, which is represented as a String. In the next box below that, you see different actions that are taken by the class, or methods. The two methods defined in this example are deposit and withdrawal, which both have amount as a parameter. Presumably, these methods will either add money to or remove money from the account, respectively.

Relationships

In a class diagram different kinds of symbols and lines are used to describe relationships among classes.

*Dependency* is the basic relationship between two classes. It basically describes a relationship between two classes where one has relies upon another for certain elements

*Association* represents a family of links. There are different types of associations that can be attributed to the relationship between two classes. They can either be bi-directional or uni-directional. In a class diagram, the association relationship is usually represented as a line.

*Aggregation* is a type of relationship that can be described with a “has a” statement. For example, a teacher “has a” class, and if there was a “teacher” class and a “student” class, the relationship between the two could be an aggregation. Each student would be a part of the “class”. In UML, this is typically represented as a hollow diamond.

Class diagram example:

