Module 133.0.1.5 :: Prerequisites :: UML Class Diagrams

UML
Unified Modeling Language

From Wikipedia: " The **Unified Modeling Language** (**UML**) is a general-purpose, developmental, **modeling language** in the field of **software engineering** that is intended to provide a standard way to visualize the design of a system.[1]

The creation of UML was originally motivated by the desire to standardize the disparate notational systems and approaches to software design. It was developed by **Grady Booch**, **Ivar Jacobson** and **James Rumbaugh** at **Rational Software** in 1994–1995, with further development led by them through 1996.[1]

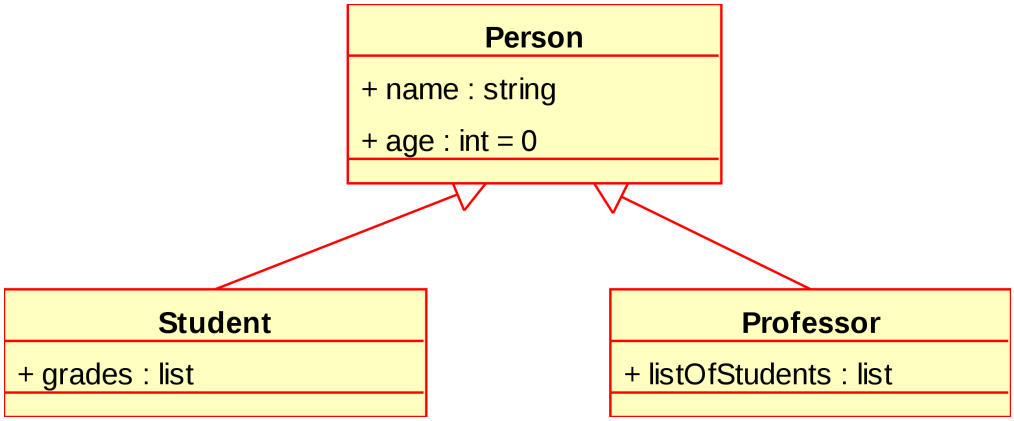
In 1997, UML was adopted as a standard by the **Object Management Group** (OMG), and has been managed by this organization ever since. In 2005, UML was also published by the **International Organization for Standardization** (ISO) as an approved ISO standard.[2] Since then the standard has been periodically revised to cover the latest revision of UML.[3] "

We will spend some time talking about UML, however, note that it's a fairly significant topic by itself and so our exploration of the topic will be limited. At this point, you should have seen UML and you should be able to talk, at least at a basic level, about the relationship between a **UML class diagram** and the classes that it represents.

UML Class Diagrams

From Wikipedia: "In **software engineering**, a **class diagram** in the **Unified Modeling Language** (UML) is a type of static structure diagram that describes the structure of a system by showing the system's **classes**, their attributes, operations (or methods), and the relationships among objects.

The class diagram is the main building block of **object-oriented** modeling. It is used for general **conceptual modeling** of the structure of the application, and for detailed modeling translating the models into **programming code**. Class diagrams can also be used for **data modeling**.[1] The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed."



Video Lecture

Don't spend too much time here, this excellent ten minute video by Zach from **LucidChart** is enough to get you going.