OO Analysis: Domain Model
Case Study

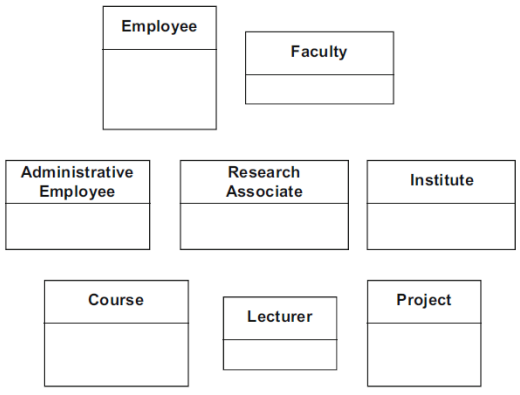
For the following case study "Information System of a University" description we create Domain Model.

> A university consists of multiple faculties which are composed of various institutes. Each faculty and each institute have a name. An address is known for each institute.
>
> Each faculty is led by a dean, who is an employee of the university.
>
> The total number of employees is known. Employees have a social security number, a name, and an e-mail address. There is a distinction between research and administrative personnel.
>
> Research associates are assigned to at least one institute. The field of study of each research associate is known. Furthermore, research associates can be involved in projects for a certain number of hours, and the name, starting date, and end date of the projects are known. Some research associates teach courses. They are called lecturers.
>
> Courses have a unique number (ID), a name, and a weekly duration in hours.

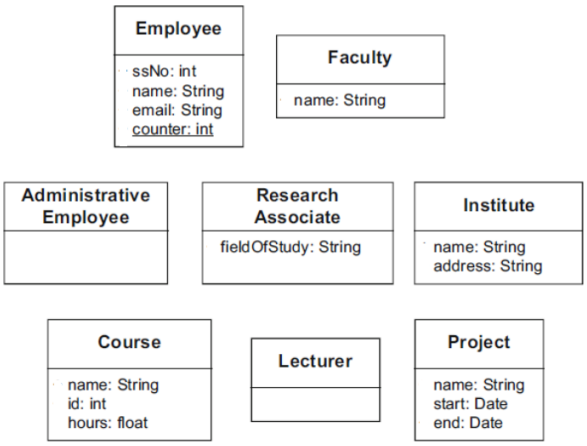
First step for domain model (Basic Class Diagram) is to identify classes by doing noun analysis in the context of system's working. Following classes are identified:

Employee	Faculty	
Administrative Employee	Research Associate	Institute
Course	Lecturer	Project

We can depict classes by using class symbol as shown below:



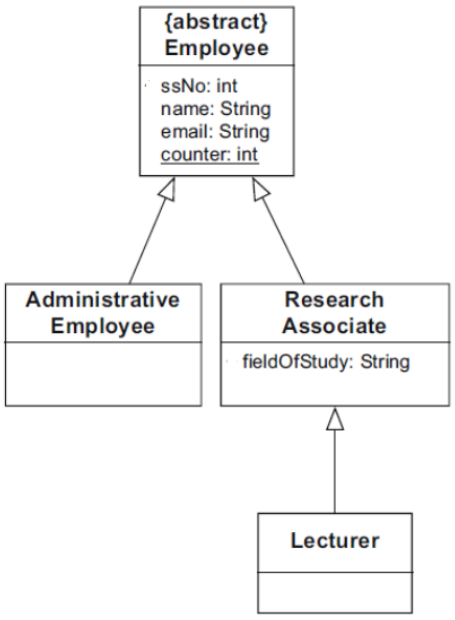
Next, we identify the attributes, and we can describe our classes in more detail using attributes. One can define meaningful data types for attributes even though these may not be included in the specification. We write attributes of respective classes in attributes section of visual classes as shown on next page.



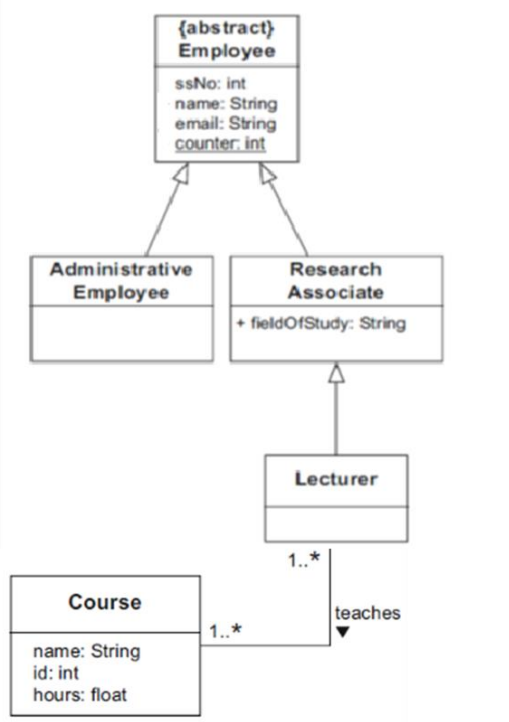
Next, we determine relationships between classes. As we know that relationships between classes can be linked with one another in three ways:

- a sub-/superclass relationship (Generalization)
- by means of an Aggregation/Composition
- via Associations

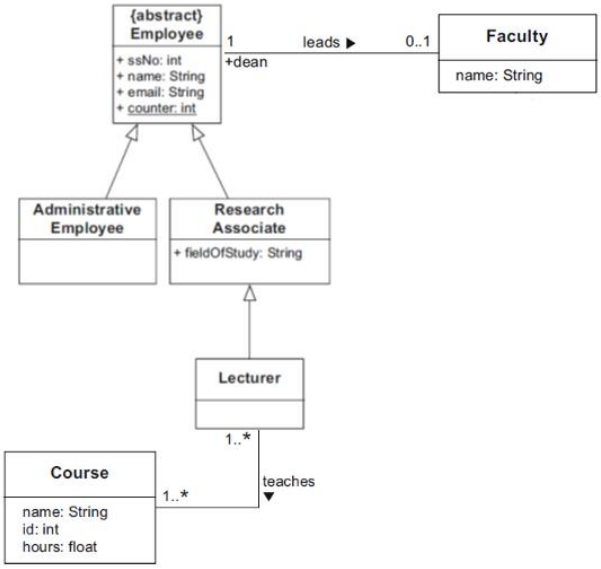
In system description “There is a distinction between research and administrative personnel”. “Some research associates teach courses. Then they are called lecturers.” As every employee of a university belongs to either the research or administrative personnel, we can set the class Employee to abstract. Sentences strongly indicate a generalization relationship.
We model these generalization relationships as follows:



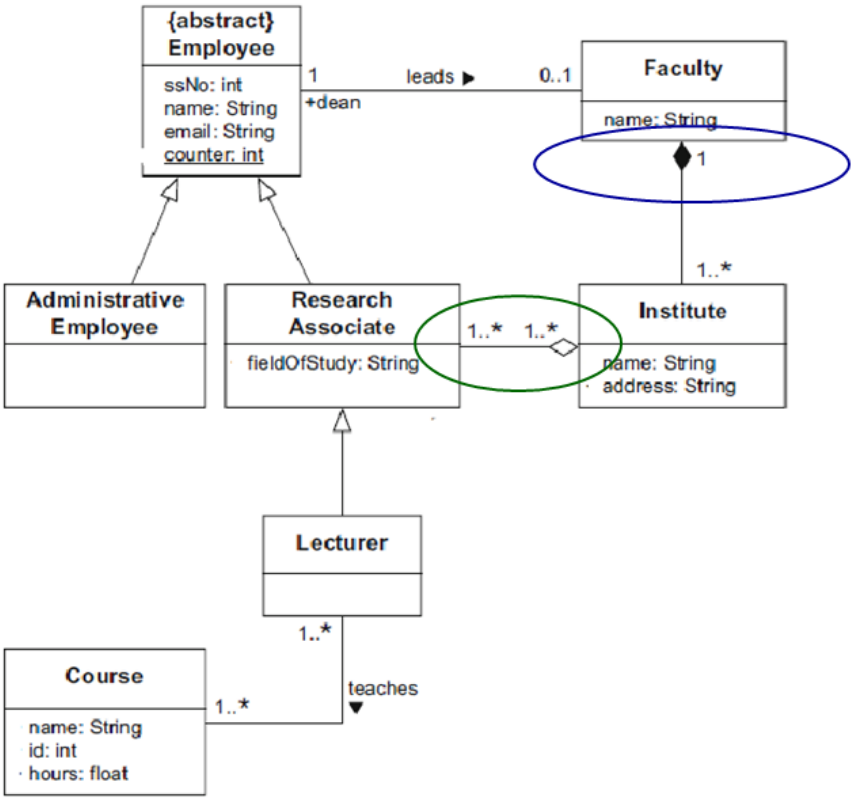
Classes *Lecturer* and *Course* are linked by means of the association teaches. *Lecturer* has generalization link with Research Associate. We extend above shown diagram by linking these identified links as shown on next page



An employee leads (role of a dean) the faculty (*Faculty*) and that link is added in the above diagram as shown below:



A *Faculty* consists of multiple *Institutes*, which indicates that there is an existence dependency which we model with a composition. *Research Associates* are assigned to an Institute, meaning they are part of an institute. Using a composition here would be incorrect as there is no existence dependency between instances of *Employee* and *Institute*. Here we identify parts-whole relationship, which is represented as a shared aggregation relationship. We add these two links to the above diagram and with added links we have the diagram as shown on next page. Composition symbol is within blue oval and aggregation symbol within green oval.



Research associates can be involved in projects for a certain number of hours, and the name, starting date, and end date of the projects are known. Association link between Research Associate (1..*) and Project (0..*) is added in the above diagram which completes our domain model (basic class diagram) as shown below: