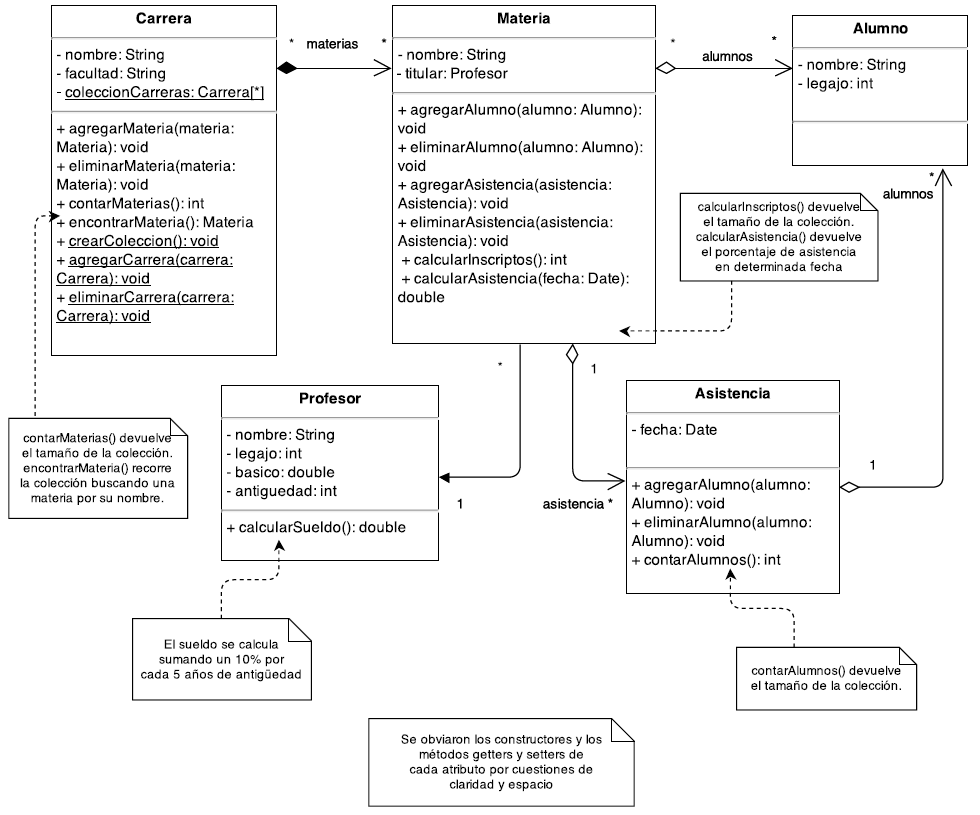
## Ejemplo de diagrama UML

A continuación se muestra un diagrama de clases que podría estar representando el sistema de inscripciones de alumnos en una universidad.

Diagrama:

[](http://patriciaemiguel.com/assets/2016-01-09-ejemplo-uml-img2.png)

## Interpretación:

Un objeto Carrera representa a una carrera universitaria. Cada carrera tiene como atributos: nombre (de tipo String), facultad (String), materias (colección de objetos de tipo Materia), coleccionCarreras (atributo estático -puede ser llamado sin instancia- de tipo colección que agrupa objetos de tipo Carrera). Además, tiene métodos para agregar y eliminar materias de la colección de materias, un método que permite contar la cantidad de materias de una carrera y devuelve un dato de tipo int, y métodos estáticos para: crear la colección de carreras, agregar y eliminar carreras a esa colección. Según la cardinalidad, una carrera tiene muchas materias y cada materia puede dictarse en sólo una carrera. De acuerdo al tipo de relación (composición), si el objeto carrera se destruye, también dejan de existir sus materias.

Un objeto Materia representa a cada una de las materias de una carrera. Cada materia tiene como atributos: nombre (String), titular (de tipo Profesor), alumnos (colección de objetos de tipo Alumno), asistencia (colección de objetos de tipo Asistencia). Además de métodos para agregar y eliminar elementos de ambas colecciones, tiene métodos para calcular la cantidad de alumnos inscriptos en la materia (que devuelve un dato de tipo int) y para calcular la asistencia en determinada fecha (recibe una fecha y devuelve un dato de tipo double). Podría interpretarse al valor de retorno del método que calcula la asistencia como un porcentaje, pero esto excede al diagrama UML (en todo caso, podría agregarse como comentario).

Un objeto Asistencia representa la asistencia de alumnos en determinada fecha. Una Asistencia tiene como atributos: fecha (Date), alumnos (colección de objetos de tipo Alumno). Tiene métodos para agregar y para eliminar alumnos de la colección, y también para devolver la cantidad de alumnos que contiene en su colección.

Un objeto Alumno representa a cada alumno de una carrera universitaria. Un Alumno tiene como atributos: nombre (String), legajo (int). No tiene métodos especiales.

Un objeto Profesor representa a cada profesor de una carrera universitaria. Un Profesor tiene como atributos: nombre (String), legajo (int), basico (double), antigüedad (int). También tiene un método para calcular el sueldo del profesor, que devuelve un dato de tipo double, y en un comentario se aclara la forma en que debe calcularse (sumar un 10% al sueldo por cada 5 años de antigüedad). La clase Profesor se relaciona con la clase Materia mediante el atributo “titular”. Según la cardinalidad, una materia tiene un único profesor (titular) y un mismo profesor puede desempeñarse en varias materias.

## Implementación:

La implementación se hace en un lenguaje concreto. En este caso, se provee un proyecto implementado en Java, que puede importarse dentro de un IDE