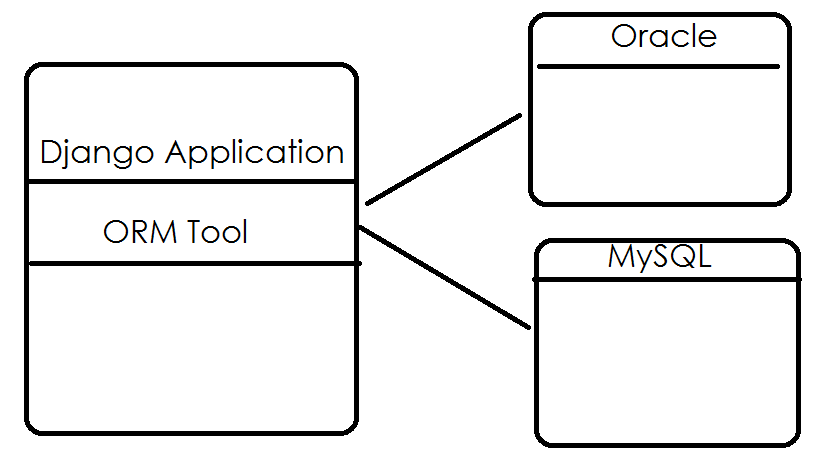
**Model**

Every project or web application required to store or save data permanently. We store or save data permanently within database.

Database is an application or software which holds data.

Database software’s or applications

1. Oracle
2. MySQL
3. SQLLite
4. SQLServer
5. PostgreSQL
6. MongoDB (NO SQL)



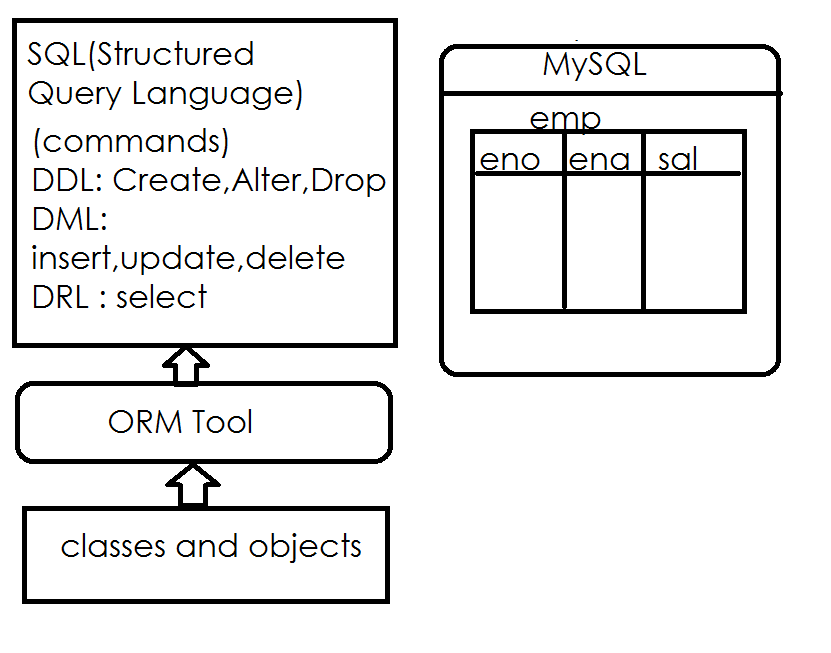
**What is ORM?**

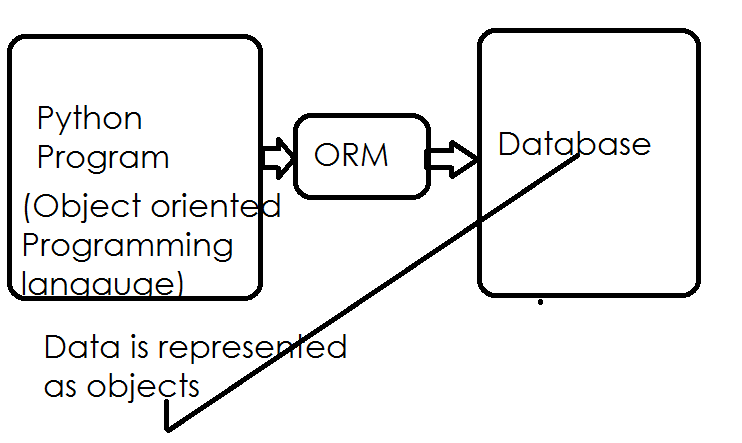
ORM stands for Object Relational Mapping

ORM is specification which define set of rules and regulation which used to develop different ORM tools.

One of the most powerful features of Django is its Object-Relational Mapper (ORM), which enables you to interact with your database, like you would with SQL. In fact, Django's ORM is just **a pythonical way to create SQL to query and manipulate your database and get results in a pythonic fashion**.

ORM tool Maps python object to relational databases.

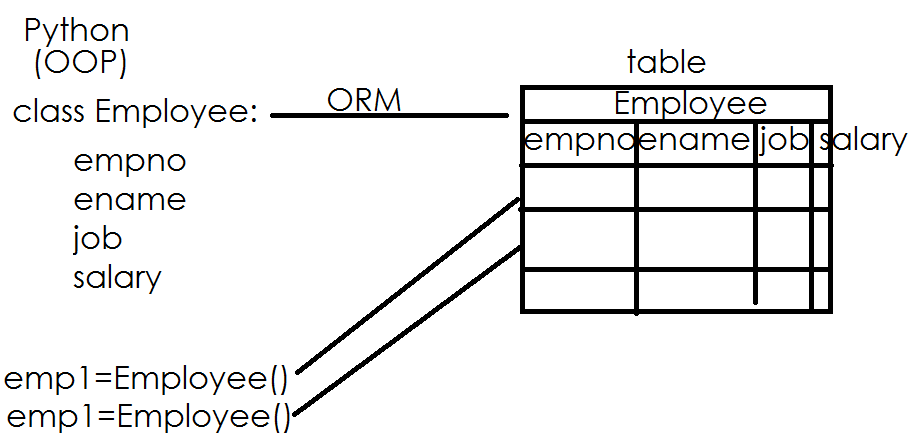


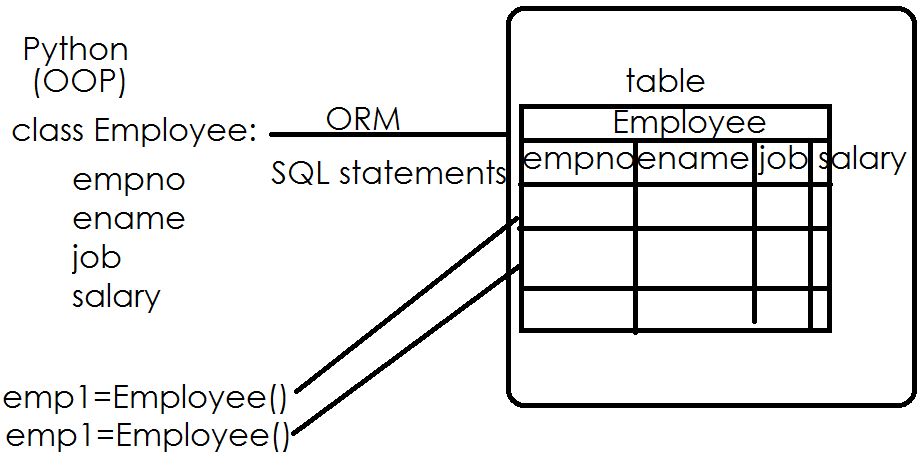


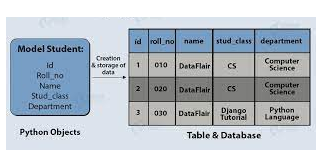
Advantage of ORM is, it allows building applications independent of databases.

SQL statements syntaxes are database dependent. In order develop application irrespective databases we use ORM tool.

ORM is a specification which followed by django developers for communicating with databases.

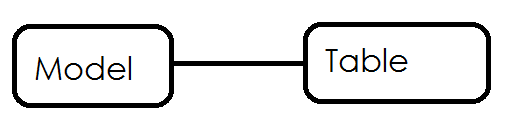




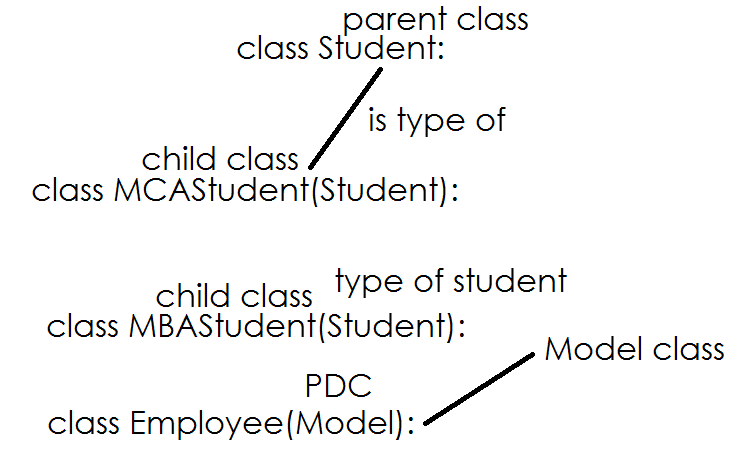


**What is model in django?**

A model represents database table.



A model is python class which inherits properties and behavior of Model class.



Model is a predefined class provided by django framework.

Every model class must inherit Model class.

Every application is having a module called **models.py**

This module consist of all models related with application.

django.db.models package

django 🡪 package

db 🡪 subpackage

models.py 🡪 module

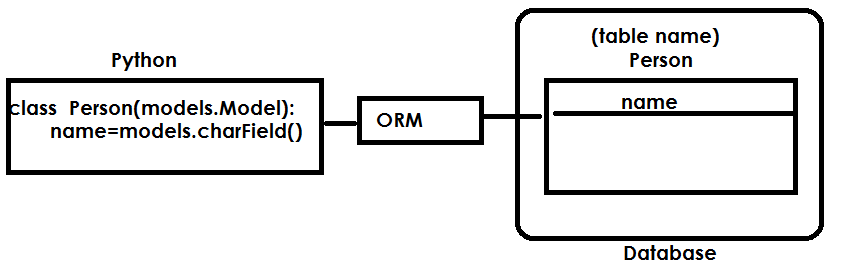
**Syntax:**

**class <class-name>(models.Model):**

variable-name/field-name=datatype

variable-name/field-name=datatype

variable-name/field-name=datatype

****

**django.db.models 🡪 is a package used to work with models in django**