**Tổng quan ORM**

Object-Related Mapping là một công nghệ lập trình để tương thích dữ liệu từ các cơ sở dữ liệu quan hệ RDBMS với các ngôn ngữ lập trình theo hướng đối tượng.

**Kiến trúc ORM**

* Ba tầng cơ bản của công nghệ ORM:

|  |
| --- |
| * Oriented: Thiết kế các lớp, thuộc tính, hàm trong, tạo CLI(call level interface) ngôn ngữ lập trình theo hướng đối tượng. |
| * QueryBuilder: Dùng để sinh các câu truy vấn giao tiếp với cơ sở dữ liệu. |
| * RDBMS: Các hệ quản trị cơ sở dữ liệu quan hệ RDBMS. |

**Hướng đối tượng trong ORM**

|  |  |
| --- | --- |
| Cơ sở dữ liệu | Khái niệm trong lập trình |
| Bảng | Lớp |
| Hàng | Đối tượng |
| Cột | Thuộc tính |

**Quan hệ giữa các đối tượng**

One-To-Many

Many-To-Many

One-To-One

**Một số thư viện**

* Có nhiều thư viện hỗ trợ công nghệ ORM
  + Java: SQLAlchemy, Dijango.
  + Python: Hibernate, 4DFLib.
  + PHP: Doctrile, Propel.

**Ví dụ**

Bài toán: Một lớp học có nhiều sinh viên, mỗi sinh viên đăng kí nhiều môn học, một trường học có một lớp.

Các bảng mục tiêu:

Class(id(primary\_key),name,school\_id(foreign\_key))

School(id(primary\_key,name))

Program(id(primary\_key,name))

Student\_Program(strudent\_id(foreign\_key),program\_id(foreign\_key))

Ta sử dụng ngôn ngữ python với thư viện sqlalchemy.

Các lớp

From flask\_alchemy import SQLALchemy

Db = SQLALchemy()

class School(db.Model):

id = db.Column(db.Integer,primary\_key=True)

name=db.Column(db.String(200),nullable = False)

db.relationship(“Class”,backref=”school”,lazy=True,uselist=False)

class Class(db.Model):

id = db.Column(db.Integer,primary\_key=True)

name = db.Column(db.String(200),nullable = Flase)

school\_id = db.Column(db.Integer,db.Foreign\_Key(school.id))

db.relationship(“Student”,backref=”class”,lazy=True)

student\_program = db.Table(

db.Column(“student\_id”,db.Integer, db.Foreign\_Key(student.id)),

db.Column(“program\_id”,db.Integer,db.Foreign\_Key(program.id))

)

class Program(db.Model):

id = db.Column(db.Integer,primary\_key=True)

name = db.Column(db.String(200),nullable=False)

db.relationship(“Student”,secondary=student\_program,backref=”programs”)

calss Strudent(db.Model):

id = db.Column(db.Integer,primary\_key=True)

name = db.Column(db.String(200),nullable=False)

class\_id = db.Column(db.Integer,db.Foreign\_Key(class.id))

db.relationship(“Program”,secondary=student\_program,backref=”students”)

Ví dụ queryBuilder: <https://flask-sqlalchemy.palletsprojects.com/en/2.x/queries/>