Unit 14.3 Highlights

ORM - Object Relational Mapping - Programming technique for converting data between incompatible type systems using object-oriented programming languages. This creates, in effect, a "virtual object database" that can be used from within the programming language.
MVC - (Model-View-Controller) Design pattern that separates an application into three main logical components: the model, the view, and the controller. Each of these components are built to handle specific development aspects of an application.

Ex. 1

```
//============connection.js (MySQL config, connection statements)==============//
var mysql = require("mysql");

var connection = mysql.createConnection({
  host: "localhost",
  port: 8889,
  user: "root",
  password: "root",
  database: "pets_db"
});

connection.connect(function(err) {
  if (err) {
    console.error("error connecting: " + err.stack);
    return;
  }
  console.log("connected as id " + connection.threadId);
});

module.exports = connection;

//============orm.js (object relational mapping)==============//

//Load your db config as a module
var connection = require("./connection.js");

var orm = {
  select: function(whatToSelect, tableInput) {
    var queryString = "SELECT ?? FROM ??";
    connection.query(queryString, [whatToSelect, tableInput], function(err, result) {
      if (err) throw err;
      console.log(result);
    });
  },
  selectAndOrder: function(whatToSelect, table, orderCol) {
    var queryString = "SELECT ?? FROM ?? ORDER BY ?? DESC";
    console.log(queryString);
    connection.query(queryString, [whatToSelect, table, orderCol], function(err, result) {
      if (err) throw err;
      console.log(result);
    });
  }
}
module.exports = orm;


//============ server.js (entry point)==============//
var orm = require("./config/orm.js");

// Find all the pets ordering by the lowest price to the highest price.
orm.selectAndOrder("animal_name", "pets", "price");

// Find a pet in the pets table by an animal_name of Rachel.
orm.selectWhere("pets", "animal_name", "Rachel");
```