Refactoring Python Web Application

CS324 Advanced Programming

Refactor your web-based email application using the Model/View/Controller design pattern and Data Access Object pattern that we’ve been talking about in class.

Each web page generated by a python program should have:

* a controller python file with a main method that is responsible for connecting the user’s web interactions to models and data stores of those models (e.g. LoginController.py class file)
* a view python file that is responsible for sending html to the user’s browser (e.g. LoginView.py class file)
* use python classes for both the controller and the view

Your data stores should be encapsulated using the Data Access Object design pattern… each DAO should have:

* a model python file that represents data you want to manipulate and display (e.g. User.py class)
* a dao python file that encapsulates the mechanism for data store and manipulation (e.g. UserDao.py class file).
* use python classes for both the model and the dao

For both the MVC and DAO design patterns use the code at <http://web.stonehill.edu/compsci/cs324/Python/bookvotercgiframework.tar>

Finally… clean up your code. Use meaningful variable names, organize statements in each method, add useful comments… make your code beautiful so that it is easy to read and easy to maintain.

READ CAREFULLY: Your elearn submission should contain:

* a tar file of the directory that contains your python program and data
* a list of your usernames/passwords so I can login and test your system

GRADING: Your grade for this problem will be based primarily on:

* a functioning email system that uses the MVC & DAO design patterns to simplify your code
* clearly written code