GROUP 12

Member Name

XINXIN ZHAO, 105215004

ANSHUL SHARMA, 110016388

CHINTAN PATEL, 110022265

Github Project: <https://github.com/MaxiAgrippa/IADjangoLab>

REQUIRED FEATURE COMPLETED

1. In admin.pycreate a class TopicAdmin(admin.ModelAdmin), register this with the admin site and show the nameand lengthfields, for each Topic, in the admin interface page that lists all Topics. Create an TabularInline class called CourseInlinebased on Coursemodel and include this under the TopicAdmin,so Coursesfor a particular Topicmay be edited on the Topicpage itself.
2. In admin.pywrite an actionfor CourseAdminclass that will reduce the priceof the course by 10%for the selected coursesand savethe updated pricein the database.
3. Create‘register’ view that allows a user to register as a Student.Update myapp/urls.pyand create register.html
4. Create myorders view. Define a view function myorders(request) in your views.pyfile. The user must be logged in to access this function. For a logged in user: if the user is a Student, return all the orders that have been placed by the user; otherwise, display a message: ‘You are not a registered student!’. Update myapp/urls.pyappropriately and create myorders.htmlto display the orders that have been placed by the studentor a suitable message (if the studenthas not placed any orders).
5. Update the user\_login viewcreated in Lab 10so that if an user who is not logged in goes to url ‘/myapp/myaccount/’they will be directed to the login page and after successful login they will go directly to the ‘/myapp/myaccount/’ page (instead of the main index page).Note: The myaccountview function was created in lab 10.
6. Update base.htmlso that if a user is logged in, it will display Logout(myapp/logout) link. Otherwise it will display Register Here(myapp/register) and Login(myapp/login) links. Each link should go to the corresponding view function defined earlier (in Lab 10or in step 3 above).
7. Update base.htmlso that if a user is logged in, it will display “Hello <first\_name>” instead of “Hello User”. Here <first\_name> is the first name of the user that is currently logged in

OPTIONAL FEATURES COMPLETED

1. Save db in JSON format. Load initial data using fixtures
2. Add validators for pricefield in Coursemodelso that it is between $50 and $500.
3. Upload image file. Add an optional image field for a Studentto upload his/her photo.
4. Update index anddetail view functions in views.pyto use class-based views (<https://docs.djangoproject.com/en/3.1/topics/class-based-views/>).
5. Use Bootstrap to style your pages
6. In admin.pycreate a class StudentAdmin(admin.ModelAdmin), register this with the admin site and show the first\_name, last\_name, levelfields and list of registered\_coursesfor each memberin the admin interface page that lists all studets.
7. Add a ‘Forgot password’ link on login page. It should email a new password to the user.
8. Update theviews.reviewfunction (created in lab 9)so that only users who are logged in and are either a ‘Undergraduate’studentor ‘Postgraduate’ studentcan submit a review.