**Django course challenge**

|  |
| --- |
| **Exercise ---------------------------------------------------------------------------------------------.**  Merchant wants us to create website for her business which is selling cosmetic products. She wants to display all brand and products. Brand names, origin. Product includes product name and brand and Expire\_date. She wants to display the name of her business in home page as (Cosmetic Boutique). Then, display list in sidebar as link to home, brands and products.  If user select:  Home link 🡪 home page with name and number of product and brand.  Brand link 🡪 brand page which list all brand from database.  Product🡪 display all product and the product as link when select it new page is opened has name of product and its brand.  You should cover all following points:   1. Create directory (labproject) and project (cosmetic) 2. Create application called (makeup) 3. Diagram, schematic     Description automatically generatedUse model to make database as according to following 4. After making models you should run database migrations. 5. Register models 6. Create superuser as admin with your group number (ex. adminG1) and password 123456 7. Add data to database 8. Add information to View (function-based) 9. Create base templates (index + base\_generic) and CSS file. 10. Create pages (index, brand list, product list, brand details, product details) |

|  |  |  |
| --- | --- | --- |
| 02 | Building a basic website + new apps | Setting Up the Development Environment  Creating Your First Django Project  Using the Integrated Terminal in VSCode  Creating Your First App |
| 03 | Models + SQLite database | [Analyse NBA Data with SQL/sqlite3](https://codechalleng.es/challenges/51/) |
| 04 | HTML + Django Templates | Using Templates in your app |
| 05 | Controllers + views & urls configurations | Writing Views in your project |
| 06 | Integrating Models, Templates, Views and URL configurations + create url | Mapping URLs to Views in your project |
| 07 | User input with forms + displaying form in templates | Building an E-Commerce Data Model & Implement HTML/Bootstrap template |
| 08 | Controlling forms in views+  (Data, Schema) migrations | Organizing Models in Apps |
| 09 | Contributed library + pagination | Pagination with Function-Based ViewsPagination with Class-Based Views |
| 10 | Website Django flatpages+  Serving static content | Installing Flatpages AppCreating flatpages using Django AdminCreating template for flatpagesRendering a List of FlatpagesServing static files during developmentServing files uploaded by a user during development |
| 11 | Generic class-based views | [Book list page](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Generic_views#book_list_page)[Creating the List View template](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Generic_views#creating_the_list_view_template)[Book detail page](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Generic_views#book_detail_page) |
| 12 | Basic Authentication + Integrating permissions | [Enabling authentication](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#enabling_authentication)[Creating users and groups](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#creating_users_and_groups)[Setting up your authentication views](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#setting_up_your_authentication_views)[Login template](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#login_template)[Logout template](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#logout_template)[Password reset templates](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#password_reset_templates)[Testing the new authentication pages](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#testing_the_new_authentication_pages)q[Testing against authenticated users](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#testing_against_authenticated_users)[Testing in views](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#testing_in_views)[Example — listing the current user's books](https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Authentication#example_%E2%80%94_listing_the_current_users_books) |
| 13 | Admin library | Use Admin library in your app  * Customize Django Admin Panel |
| 14 | Python OOP + Class coding basics. | Create a To-Do List App using oop |