HTML - Assessment

Introduction

This week you will design and create a **static website for an admin dashboard of a Food order application**. The website will serve as the administrative interface for managing various aspects of the food ordering system. The aim is to provide a user-friendly interface for administrators to efficiently handle food items, cuisines, categories, restaurants, orders, and user accounts.

You will be building static web pages using HTML only. Because you will be building static web pages so for all the categories, cuisines, food items and restaurants information, you can fill up arbitrary data. You can have at least 3 arbitrary records for each of these information. Kindly note you should also have images displayed for each of the records as shown in the sample screenshots shared below.

Housekeeping points

- This is a minimal example and may not follow some standard practices.
- We focus on the main flow, and not much error handling.
- You can either use external URLs for images or image files.
- Scores will be deducted if images are not displayed correctly. So make sure you use relative paths and share image files along with the html files.

Problem Statement

Your task is to build the following web pages -

1. **Admin Dashboard –**
    a. The homepage should present an organized layout displaying sections for managing food items, cuisines, categories, restaurants, orders, and users.
    b. Each section should have clickable links directing the user to specific functionalities related to that category (e.g., adding or viewing).
    c. It should have the following sections –
       - **Food Items –** Should have links to allow administrators to view all food items (list_fooditems.html) and add new ones (add_fooditem.html).
       - **Cuisines –** Should have links to allow administrators to view all cuisines (list_cuisines.html) and add new ones (add_cuisine.html).
       - **Categories –** Should have links to allow administrators to view all categories (list_categories.html) and add new ones (add_category.html).
       - **Restaurants –** Should have links to allow administrators to view all restaurants (list_restaurants.html) and add new ones (add_restaurant.html).
       - **Orders –** Should have links to allow administrators to view all orders placed (The link can be an empty link as there is no corresponding page for displaying list of orders).
       - **Users –** Should have links to allow administrators to view all registered users (The link can be an empty link as there is no corresponding page for displaying list of users).

Here is a sample Admin webpage -

Admin Dashboard

Food Items

View Food Items Add Food Item

Cuisines

View Cuisines Add Cuisine

Categories

View Categories Add Category

Restaurants

View Restaurants Add Restaurant

Orders

View Orders

Users

View Registered Users

2. **Web pages for Create Operation –**
   a. **Add Cuisine –** Web page having form to add a new cuisine.

Add Cuisine

Cuisine Name: [   ]

Cuisine Description: [   ]

Image URL : [   ]

Is Active ☐

[Discard] [Save]

b. **Add Category–** Web page having form to add a new category.

Add Category

Category Name: [   ]

Category Description: [   ]

Is Active ☐
Image URL : [   ]

[Discard]   [Save]

c. **Add Food item –** Web page having form to add a new food item.

Add FoodItem

FoodItem Information

FoodItem Name : [   ]

FoodItem Description : [   ]

Available? ☐
Vegeterian? ☐
Category : [Pasta ▼]
Cuisine : [Italian ▼]
Image URL : [   ]

[Discard]   [Save]

d. **Add Restaurant –** Web page having form to add new restaurant.

Kindly note that as you are required to build only a static website so you don't need to handle any event on form submission on any of the web pages.

3. Web pages for List Operation –

a. **List Cuisines–** Web page having a list of all available cuisines information along with images.

All Cuisines

Cuisine No.	Cuisine Name	Description	Active?	Image	Edit
1	Italian	All Italian food	Yes		Edit
2	Asian	All Asian foods	Yes		Edit
3	Chinese	All chinese food	Yes		Edit

b. **List Categories–** Web page having a list of all available categories information along with images.

All Categories

Category No.	Category Name	Description	Active?	Image	Edit
1	Pasta	All types of pasta	Yes		Edit
2	Pizza	All types of pizza	Yes		Edit
3	Sandwich	All types of sandwiches	Yes		Edit
4	Soup	All types of soups	Yes		Edit

c. **List Food items –** Web page having a list of available food items information along with images.

All Food Items

FoodItem No.	Name	Description	Available?	Vegeterian?	Category	Cuisine	Image	Edit
1	Veg Pizza	Veggies include onion, tomato, capsicum etc	Yes	Yes	Pizza	Italian		Edit
2	Hot & Sour Soup	Made with mushrooms, tofu, spring onions & chicken	Yes	No	Soup	Chinese		Edit
3	Lasagna Sandwich	Bacon, sour cream, mozzarella cheese and white bread	Yes	No	Sandwich	Italian		Edit
4	Chick-Fil-A-Sandwich	Chicken pattice, Egg, Milk & Burger bun	Yes	No	Sandwich	Asian		Edit
5	Grilled Mac & Cheese Sandwich Macaroni, Cheddar cheese, Jack cheese, Heavy cream & white bread. Yes				Yes	Sandwich	Asian		Edit

d. **List Restaurants –** Web page having a list of available restaurants information along with images.

All Restaurants

Restaurant No.	Restaurant Name	Restaurant Image	Address	Contact	Edit
1	Food Town		Abbey Street, Berkshire	27287	Edit
2	Green Leaf Hotel		Northwest street, Skyville	37484	Edit
3	Kignston Hotel		Dudley Road, Hillwood	7448	Edit

Kindly note the screenshots available are sample web pages. You can structure your web pages as per your choice.

Program Organization

- You will be getting a zip folder containing a folder named **HTML Assessment** (Available under the header **Source Code** in the Olympus page).
- The **HTML Assessment** folder contains a folder named **Food order app** within which folder named **admin_view** is available. As you will be building a static website for the admin dashboard, so you will be working within this folder. This folder has the following empty files -
- admin_page.html – To implement Problem Statement 1.
- add_fooditem.html, add_category.html, add_cuisine.html, add_restaurant.html – To implement Problem Statement 2.
- list_fooditems.html, list_categories.html, list_cuisines.html, list_restaurants.html – To implement Problem Statement 3.

Evaluation Rubric

Total Project Points: 60

- Correctness:
  Correctness of implementation
  - Problem statement - point 1 (**20%**)   : **12 Points**
  - Problem statement - point 2 (**40%**)   : **24 Points**
  - Problem statement - point 3 (**40%**)   : **24 Points**

Program Instructions

- There should be 9 HTML files within the admin_view folder.
- All the HTML files must be present in the admin_view folder only.
- There should also be the image files(if you are not using external URLs for images) either in the admin_view folder or out of admin_view folder. But you will have to set the path of the images within the corresponding img tags accordingly so that all images are displayed correctly. (Use relative path in img tag). All the images should be visible on the corresponding web pages.
- The **admin_view** folder should be a subfolder of the **Food order app** folder which is a subfolder of **HTML Assessment** and the **HTML Assessment** folder must be submitted as the zip folder.
- Project will not be evaluated if the submitted project is not in the zip format.