

Front end submission

Iliyas Ahmed
ID: 2020365
Section: 03

Indronil Dey Niloy
ID: 2020339
Section: 05

Nohori Jobaida
ID: 1931154
Section: 05

Ammarul Islam Anik
ID: 1910594
Section: 03

Sadia Akter
ID: 2022719
Section: 05

Syed Shamsul Arefin
ID: 1931335
Section: 05

December 2023

1 Log In page:

1.1 Screenshot:

The screenshot shows two code editor windows side-by-side, both titled "Login page".

The top window displays the `login_page.html` file:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Login Form | By Code Info</title>
    <link rel="stylesheet" href="login_page.css">
  </head>
  <body>
    <div class="container">
      <div class="login">
        <form>
          <h1>Login</h1>
          <hr>
          <label>User ID</label>
          <input type="text" placeholder="123456">
          <label>Password</label>
          <input type="password" placeholder="enter your password!">
          <button type="submit">Login</button>
          <p>Forgot Password?<a href="#">Forgot Password?</a></p>
        </form>
      </div>
    </div>
  </body>
</html>
```

The bottom window displays the `login_page.css` file:

```
#login_page {
  span {
    font-family: verdana, geneva, sans-serif;
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: "Poppins", sans-serif;
  }
  body {
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
    background-color: #lemonchiffon;
  }
  .container {
    width: 100%;
    display: flex;
    max-width: 850px;
    border-radius: 15px;
    box-shadow: 0 10px 15px rgba(0, 0, 0, 0.1);
  }
  .login {
    width: 600px;
  }
  form {
    width: 250px;
    margin: 0 auto;
  }
  h1 {
    margin: 20px;
    text-align: center;
    font-weight: bold;
    text-transform: uppercase;
  }
}
```

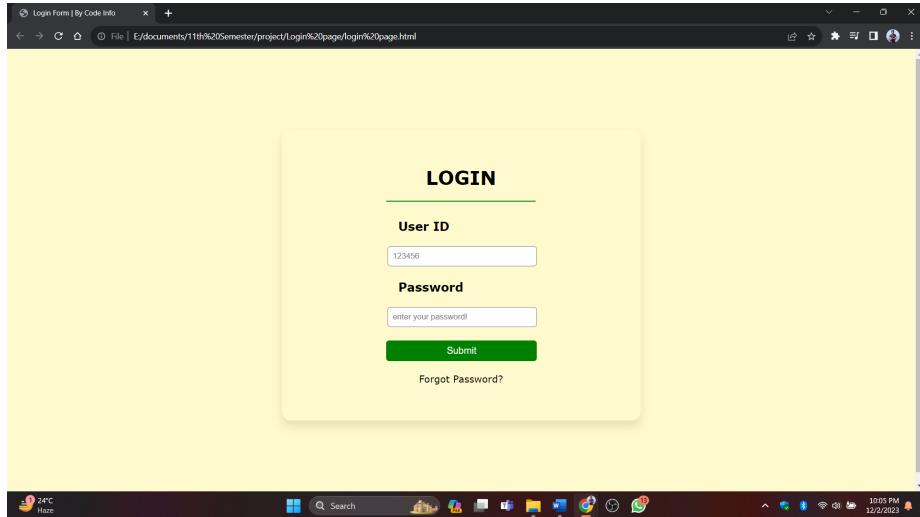
The image shows two side-by-side screenshots of a code editor interface, likely Visual Studio Code, displaying the same CSS file for a login page. Both screens show the following CSS code:

```
span { font-size: 16px; font-weight: bold; color: black; text-decoration: none; }</span>
```

The CSS file is named `login.page.css` and contains the following code:

```
38 p{ 39   text-align: center; 40   margin: 10px; 41 } 42 hr{ 43   border-top: 2px solid #green; 44 } 45 .pic img{ 46   width: 400px; 47   height: 100%; 48   border-top-right-radius: 15px; 49   border-bottom-right-radius: 15px; 50 } 51 form label{ 52   display: block; 53   font-size: 20px; 54   font-weight: 600; 55   padding: 20px; 56 } 57 input{ 58   width: 100%; 59   margin: 2px; 60   border: none; 61   outline: none; 62   padding: 8px; 63   border-radius: 5px; 64   border: 1px solid #gray; 65 } 66 button{ 67   padding: 8px; 68   border-radius: 5px; 69   border: none; 70   outline: none; 71   padding: 8px; 72   width: 250px; 73   color: #fff; 74   background-color: #green; 75   font-size: 16px; 76   cursor: pointer; 77   margin-top: 20px; 78   border-radius: 5px; 79   background: #green; 80   button:hover{ 81     background: #rgba(214, 86, 64, 1); 82   } 83   p{ 84     margin: 20px; 85   } 86   a{ 87     color: black; 88     text-decoration: none; 89   } 90 }</span>
```

The interface includes a top navigation bar with File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar. Below the search bar is a toolbar with icons for Save, Undo, Redo, Find, Replace, Copy, Paste, and others. The main area is divided into an Explorer sidebar on the left containing files like `field.jpeg`, `login.page.css`, `login.page.html`, and `login.page.png`; a Preview pane on the right showing a preview of the login page; and a bottom status bar showing system information like battery level, signal strength, and date/time.



Document Declaration:

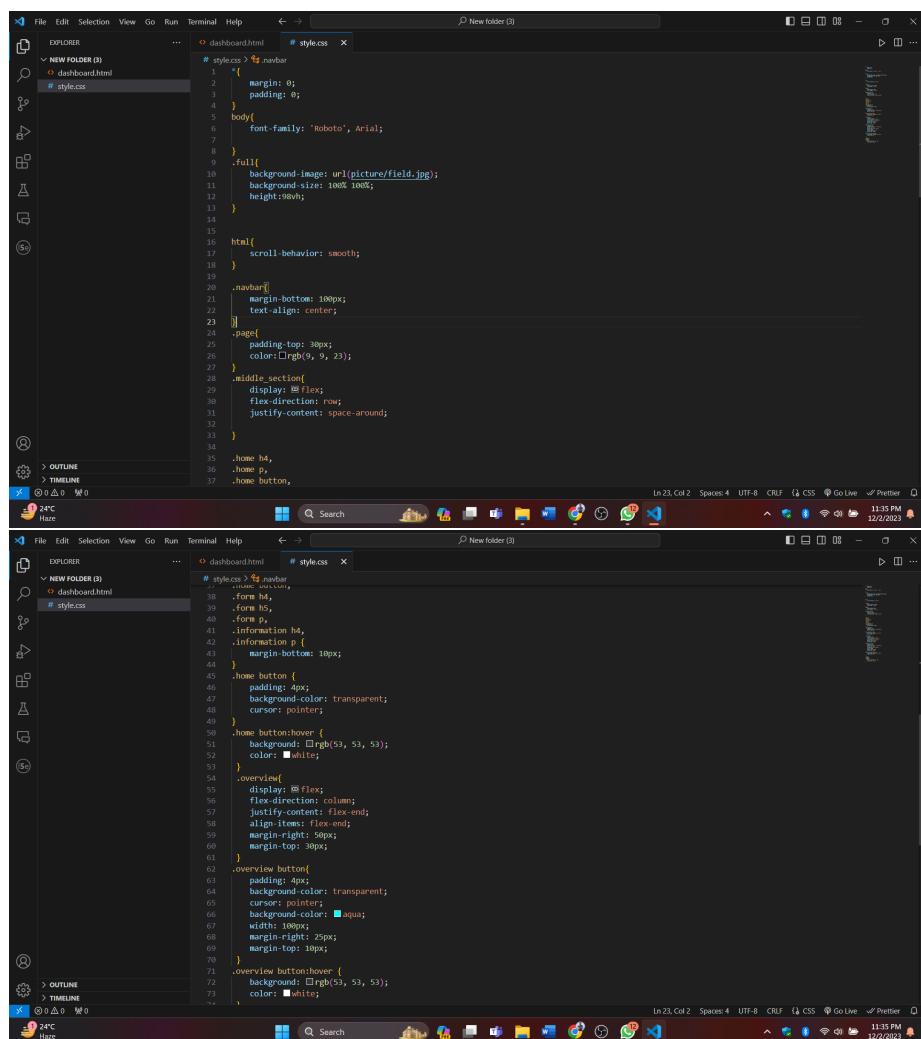
The `!DOCTYPE html` declaration specifies that the document is an HTML5 document. HTML Structure: The `html` tag defines the root of the document, and the lang attribute is set to "en" for English. Metadata: The `head` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. External Stylesheet: The code links an external stylesheet named "login page.css" to apply styles to the HTML elements. Body Content: Container Div: The main content is wrapped in a `div` with the class "container." Login Div: Within the container, there is a `div` with the class "login" representing the login form. Form: A `form` element contains the login form. Form Elements: The form includes: An `h1` heading displaying "Login." An `hr` horizontal line for visual separation. Two `label` elements for User ID and Password. Two `input` elements for entering User ID and Password. A `button` element for submitting the form. A paragraph (`p`) with an anchor (`a`) for "Forgot Password?" linked to. CSS Styling: Styling: The styles for the page are expected to be defined in the external CSS file "login page.css." The actual content of this file is not provided. Functionality: Login Form: The form allows users to input their User ID and Password. Submit Button: The "Submit" button presumably triggers form submission. Forgot Password Link: The "Forgot Password?" link provides an option for users who have forgotten their password. CSS Styling (Not Provided): The actual visual appearance and layout of the login page are dependent on the styles defined in the "login page.css" file, which is not provided in the given code snippet. The styles would determine the look and feel of the form, including colors, fonts, and layout.

2 Farmer page:

2.1 Screenshot:

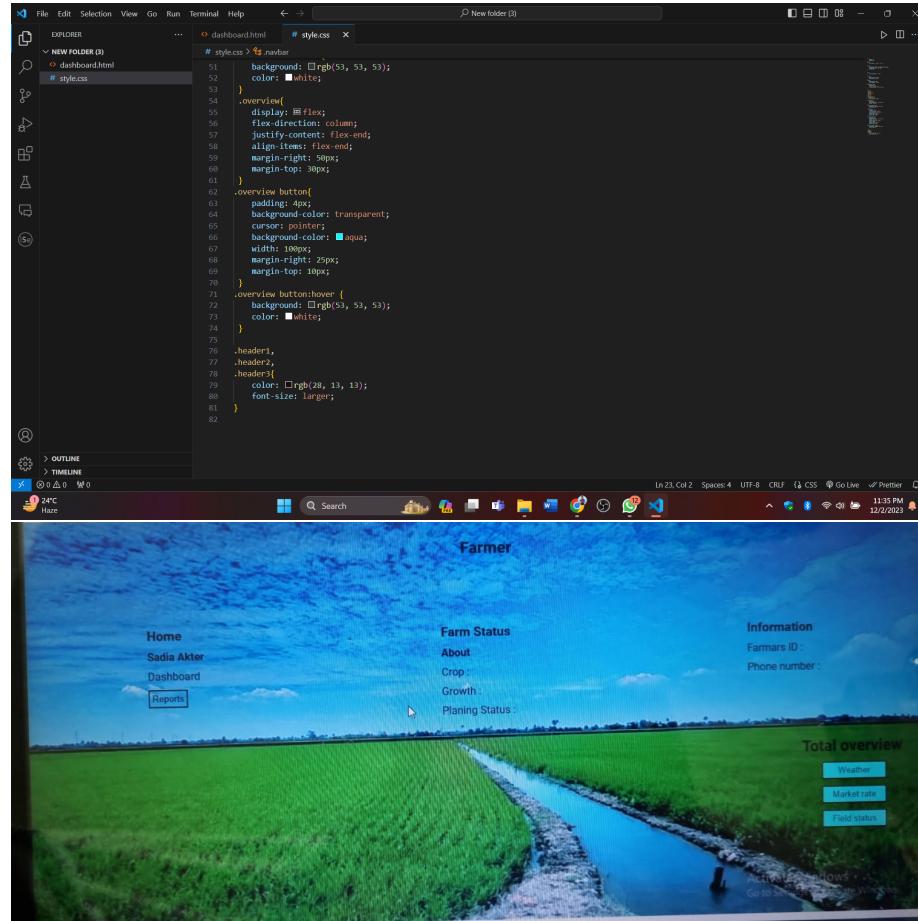
The screenshot shows a code editor interface with three vertically stacked sections of a dashboard.html file. The sections are identical except for the line numbers at the bottom.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Farmers</title>
    <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
    <link href="https://fonts.googleapis.com/css2?family=Roboto:wght@400;500;700&display=swap" rel="stylesheet">
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <div class="full">
      <nav class="navbar">
        <h1 class="page">Farmers</h1>
      </nav> / navbar
      <div class="middle_section">
        <div class="home">
          <div class="header1">Home</div>
          <div>Sadia Akter</div>
          <p>Dashboard</p>
          <button>Reports</button>
        </div>
        <div class="form">
          <div class="header2">Farm Status</div>
          <div>About</div>
          <p>Crop :</p>
          <p>Soil :</p>
          <p>Planning Status :</p>
        </div> / form
        <div class="information">
          <div class="header3">Information</div>
          <p>Farm ID :</p>
          <p>Phone number :</p>
        </div> / information
      </div> / middle_section
      <div class="overview">
        <h2>Total overview</h2>
        <button>Weather</button>
        <button>Market rate</button>
        <button>Field status</button>
      </div> / overview
    </div> / full
  </body>
</html>
```



```
# dashboard.html # style.css
1  .navbar {
2    margin: 0;
3    padding: 0;
4  }
5  body {
6    font-family: 'Roboto', Arial;
7  }
8  .full{
9    background-image: url(picture/field.jpg);
10   background-size: 100% 100%;
11   height:900px;
12 }
13
14
15  html{
16    scroll-behavior: smooth;
17  }
18
19  .navbar[] {
20    margin-bottom: 100px;
21    text-align: center;
22  }
23
24  .page{
25    padding-top: 30px;
26    color:#rgb(9, 9, 23);
27  }
28  .middle section{
29    display: flex;
30    flex-direction: row;
31    justify-content: space-around;
32  }
33
34
35 .home hd,
36 .home p,
37 .home button,
```

```
# dashboard.html # style.css
1  .form h3,
2  .form p,
3  .form p,
4  .information h4,
5  .information p {
6    margin-bottom: 10px;
7  }
8  .home button {
9    padding: 4px;
10   background-color: transparent;
11   cursor: pointer;
12 }
13 .home button:hover {
14   background: #rgb(53, 53, 53);
15   color: #white;
16 }
17 .overview {
18   display: flex;
19   flex-direction: column;
20   justify-content: flex-end;
21   align-items: flex-end;
22   margin-right: 50px;
23   margin-top: 30px;
24 }
25 .overview button{
26   padding: 4px;
27   background-color: transparent;
28   border:none;
29   background-color: #aqua;
30   width: 100px;
31   margin-right: 25px;
32   margin-top: 10px;
33 }
34 .overview button:hover {
35   background: #rgb(53, 53, 53);
36   color: #white;
```



Document Declaration: The `<!DOCTYPE html>` declaration specifies that the document is an HTML5 document. **HTML Structure:** The `<html>` tag defines the root of the document, and the `lang` attribute is set to "en" for English. **Metadata:** The `<head>` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. **External Stylesheet:** The code links an external stylesheet named "style.css" to apply styles to the HTML elements. **Body Content:** Container Div: The main content is wrapped in a `<div>` with the class "full." Navigation (Navbar): A `<nav>` element with the class "navbar" contains the page title "Farmer." Middle Section Div: A `<div>` with the class "middle_section" represents the main content area. **HomeSection :** A `<div>` with the class "home" contains: An `<h4>` element with the class "header1" displaying "Home." Another `<h4>` element displaying "Sadia Akter." A `<p>` element displaying "Dashboard." A `<button>` element labeled "Reports." **FormSection :** A `<div>` with the class "form" contains: An `<h4>` element with the class "header2" displaying "Farm Status." Two `<h4>` elements displaying "About" and "Farm Status." Several `<p>` elements displaying information about the farm. **Information :** An `<div>` with the class "information" contains: An `<h4>` element with the class "header3" displaying "Info".

p > elements displaying Farmer's ID and Phone number. OverviewSection : A < div > with the class "overview" contains : An < h2 > element displaying "Total Overview." Three < button > elements labeled "Weather," "Market Rate," and "Field Status." External Fonts : Google Fonts : The code uses the Google Fonts API to import the "Roboto" font with varying weights (300, 400, 500, Navigation : The navigation area includes the title "Farmer." Home Section : Displays the user's name, a dashboard

3 Quality Controller:

3.1 Screenshot:

```
consumer.html X
D: > db > consumer.html > html > body > nav
30   <button id="btn" type="button"><a href="/orderPage.html">Order</a>"</button>
31   <h2>Order Status</h2>
32   <table>
33     <thead>
34       <tr>
35         <th>OrderID</th>
36         <th>Product</th>
37         <th>Quantity</th>
38         <th>Amount</th>
39       </tr>
40     </thead>
41     <tbody id="orderstatus">
42     </tbody>
43   </table>
44 </div>
45
46 <div class="section">
47   <h2>Last Purchase</h2>
48   <table>
49     <thead>
50       <tr>
51         <th>OrderID</th>
52         <th>Product</th>
53         <th>Quantity</th>
54         <th>Amount</th>
55       </tr>
56     </thead>
57     <tbody id="lastpurchase">
58     </tbody>
59   </table>
60 </div>
61 </div>
```

```

consumer.html
D: > db > consumer.html > html > body > nav
1   <!DOCTYPE html>
2   <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <title>Consumer Dashboard</title>
7     <link rel="stylesheet" href="consumer.css">
8
9   </head>
10  <body>
11    <nav>
12      <a href="#">Home</a>
13      <a href="#">About</a>
14      <a href="#">Contact Us</a>
15    </nav>
16    <header>
17      <h1>Consumer Dashboard</h1>
18    </header>
19
20    <div class="dashboard-container">
21      <div class="sidebar">
22        <h2 style="color: #rgb(20, 255, 94); text-align: center;">Iliyas Ahmed</h2>
23        <h2>Dashboard</h2>
24        <h2>Change Password</h2>
25
26      </div>
27
28      <div class="main-content">
29        <div class="section">
30          <button id="btn" type="button"><a href="/orderPage.html">Order</a></button>
31          <h2>Order Status</h2>
32          <table>

```

HTML Structure:

- `!DOCTYPE html`:
 - Defines the document type and version of HTML used.
- `html lang="en"`:
 - Declares the language of the document.
- `head`:
 - Contains metadata, such as character set, viewport settings, and the page title.
- `meta charset="UTF-8"`:
 - Specifies the character encoding of the document as UTF-8.
- `meta name="viewport" content="width=device-width, initial-scale=1.0"`:
 - Sets the viewport width and initial scale for responsiveness.
- `title`:
 - Defines the title of the web page.
- `link rel="stylesheet" href="consumer.css"`:
 - Links an external CSS file for styling.
- `body`:
 - Contains the visible content of the web page.
- Body Sections:
 - `nav`:
 - Defines a navigation bar with links to Home, About, and Contact Us.
 - `header`:
 - Displays the heading "Consumer Dashboard".
 - `div class="dashboard-container"`:
 - Contains the dashboard content divided into a sidebar and main content.
 - Sidebar Section:
 - `div class="sidebar"`:
 - Represents the sidebar section.
 - `h2`:
 - Displays the user's name and navigation options (Dashboard, Change Password).
 - Main Content Section:
 - `div class="main-content"`:
 - Contains the main content of the dashboard.
 - Two sections are defined with tables to display "Order Status" and "Last Purchase".
 - JavaScript Section:
 - The script contains JavaScript code that creates dummy data for `orderstatus` and `lastpurchase`.
 - `populateTable()` function populates the tables with data by dynamically creating table rows and cells.
 - Data Display:
 - `orderstatus` and `lastpurchase` tables display Order ID, Product, Quantity, and Amount for each transaction.

The JavaScript code creates a function `populateTable()` to dynamically generate rows and populate tables based on the provided dummy data. This HTML code creates a basic consumer dashboard with order status and last purchase information displayed in tables.

4 Customer:

4.1 Screenshot:

The screenshot shows a web browser window titled "Quality Controller Dashboard". The URL in the address bar is "nohorijobaida.github.io/Database-Project/". The page has a blue header with the title "Quality Controller Dashboard" and navigation links for "Home", "About", and "Contact Us". On the left, there is a sidebar with a user profile picture of "Nohori Jobaida" and menu options "Dashboard" and "Profile". The main content area contains two tables: "Incoming Data" and "Processed Data".

Date	Product	Quantity
2023-11-01	Rice	1000
2023-11-02	Wheat	800
2023-12-03	Potatoes	500
2023-12-04	Carrots	300

Date	Product	Quality Status
2023-11-01	Rice	Pass
2023-11-02	Wheat	Fail
2023-12-03	Potatoes	Pass
2023-12-04	Carrots	Pass

This HTML code defines a simple web page for a Quality Controller Dashboard.

1. Document Type Declaration (DOCTYPE): “html !DOCTYPE html;” This declaration specifies the HTML version being used, which is HTML5 in this case. 2. HTML Structure: “html |html lang=”en”;” The ‘|html’ tag represents the root of the HTML document. The ‘lang’ attribute indicates that the document is written in English.

3. Head Section Title of the webpage |title|Quality Controller Dashboard|/title|

Link to an external stylesheet (styles.css) for styling |link rel=”stylesheet” href=”styles.css”|/head| The ‘|head|’ section contains metadata about the document, such as character set, viewport settings, and a link to an external stylesheet for styling.

4. Body Section “html |body|

Header section with the title |header| |h1|Quality Controller Dashboard|/h1| |/header|

Navigation menu |nav| |a href=””|/Home|/a| |a href=””|/About|/a| |a href=””|/Contact Us|/a| |/nav|

Dashboard container with sidebar and main content |div class=”dashboard-container”| !– Sidebar section –|div class=”sidebar”| |h2 style=”color: rgb(20, 255, 94); text-align: center;”|Nohori Jobaida|/h2| |h2|Dashboard|/h2| |h2|Profile|/h2| |/div|

Main content section Section for incoming data with a table `<div class="section"><h2>Incoming Data</h2><table>` !– Table header –`<thead><tr>`

`<th>Date</th><th>Product</th><th>Quantity</th></tr></thead>` Table body for incoming data `<tbody id="incomingDataBody"></tbody></table></div>`

Section for processed data with a table `<div class="section"><h2>Processed Data</h2><table>` !– Table header –`<thead><tr><th>Date</th><th>Product</th><th>Quality Status</th></tr></thead>` Table body for processed data `<tbody id="processedDataBody"></tbody></table></div></div>`

JavaScript section for populating tables with data `<script> // Dummy data for illustration <...> </script> </body>` “ The ‘`<body>`‘ section contains the content of the webpage. It includes a header, navigation menu, a dashboard container with a sidebar and main content, and a script tag for JavaScript.

5. JavaScript Section “`<html><script> // Dummy data for illustration const incomingData = [// ...];`

`const processedData = [// ...];`

`function populateTable(tableBody, data) const body = document.getElementById(tableBody);`

`body.innerHTML = ";`

`data.forEach(item =>`

`const row = document.createElement('tr'); row.innerHTML = '<td>dollaritem.date</td><td>`

`// Populating tables with dummy data populateTable('incomingDataBody', incomingData); populateTable('processedDataBody', processedData); </script>`

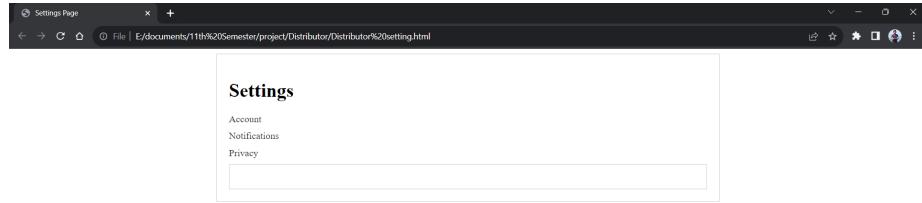
“ The ‘`<script>`‘ section contains JavaScript code. It defines dummy data for incoming and processed data, as well as a function (‘`populateTable`’) to populate HTML tables dynamically. Finally, it calls the ‘`populateTable`‘ function to fill the tables with the dummy data.

Methodology Connection: The term ”Quality Controller Dashboard” suggests a methodology for monitoring and controlling the quality of data or products. The navigation menu items (“Home,” ”About,” ”Contact Us”) represent a structured approach to navigating the dashboard or providing additional information. The use of JavaScript for dynamically populating tables demonstrates a methodology for displaying and managing data in a dynamic and interactive manner.

5 Distributor:

5.1 Screenshot:

The screenshot displays two windows of a web application. The top window is titled 'DISTRIBUTOR | By Code Info' and shows the 'DISTRIBUTOR'S DASHBOARD'. It features four main sections: 'OVERVIEW' (with fields for Total Sales, Profit Margin in \$, Customer satisfaction, and On time delivery rate), 'Sales By Region' (listing Asia, South America, North America, and Europe), 'PRODUCT' (listing Vegetables, Fruits, Dairy, and Grains and cereals), and 'Wholesaler's Details' (fields for Name, ID, E-mail, and Contact number). The bottom window is titled 'Distributor profile' and shows the user's profile information: a profile picture of Indronil Dey Niloy, the name 'Indronil Dey Niloy', an 'About Me' section stating 'I am a global food distributor of this company. I always try to do my best to serve my company.', an 'Occupation' field showing 'Distributing job', a 'Location' field showing 'Dhaka', and a 'Contact' field showing 'ainoydhe23@gmail.com'. Both windows are running on a Windows operating system, with taskbars at the bottom showing various application icons.

A screenshot of a code editor window titled "Distributor.html". The editor displays the HTML code for a distributor dashboard. The code includes a header section with meta tags, a title, and links to CSS files. The main content area features a navigation bar with items like "Home", "Profile", "Settings", and "Help". The code editor interface includes a sidebar with file navigation, a status bar at the bottom, and system tray icons at the very bottom.

Distributor.html

```

<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Distributor</title>
    <link href="Distributor.css" rel="stylesheet">
  </head>
  <body>
    <div id="container">
      <nav>
        <a href="#">Home
        <a href="#">Help
        <a href="#">Logout
      </nav>

      <section class="main">
        <div class="main-top">
          <h1>DISTRIBUTOR'S DASHBOARD</h1>
          <i class="fas fa-user-cog"></i>
        </div>
        <div class="main-bottom">
          <div class="card">
            <div class="card-body">
              <i class="fas fa-laptop-code"></i>
              <h3>OVERVIEW</h3>
              <ul>
                <li>Total Sales:</li>
                <li>Profit Margin in $:</li>
                <li>Customer satisfaction:</li>
                <li>Delivery rate:</li>
                <li>Percentage:</li>
              </ul>
            </div>
          </div>
          <div class="card">
            <div class="card-body">
              <h3>Sales by region:</h3>
              <ul type="1">
                <li>Asia:</li>
                <li>Europe:</li>
                <li>South America:</li>
                <li>North America:</li>
              </ul>
            </div>
          </div>
          <div class="card">
            <div class="card-body">
              <h3>PRODUCT</h3>
              <ul type="1">
                <li>Vegetables:</li>
                <li>Fruits:</li>
                <li>Grains and cereals:</li>
                <li>Meat:</li>
                <li>Dairy:</li>
                <li>Grains and cereals:</li>
                <li>Meat:</li>
                <li>Dairy:</li>
              </ul>
            </div>
          </div>
          <div class="card">
            <div class="card-body">
              <h3>Wholesaler's Details:</h3>
              <ul type="1">
                <li>Name:</li>
                <li>Address:</li>
                <li>ID:</li>
                <li>Email:</li>
                <li>Phone number:</li>
                <li>Contact number:</li>
              </ul>
            </div>
          </div>
        </div>
      </section>
    </div>
  </body>
</html>

```

Distributor.css

```

body {
  font-family: Arial, sans-serif;
}

.nav {
  background-color: #f2f2f2;
  padding: 10px;
  text-align: center;
}

.nav a {
  color: black;
  text-decoration: none;
  margin: 0 10px;
}

.main {
  padding: 20px;
}

.main-top {
  text-align: center;
}

.main-bottom {
  display: flex;
  justify-content: space-around;
}

.card {
  border: 1px solid #ccc;
  padding: 10px;
  width: 250px;
  margin: 10px 0;
}

.card-body {
  text-align: center;
}

h3 {
  margin: 0;
}

ul {
  list-style-type: none;
  padding-left: 0;
}

li {
  margin: 5px 0;
}

```

Explanation of methodology of Distributor Dashboard Page: HTML Structure: The document begins with the `<!DOCTYPE html>` declaration, indicating that it is an HTML5 document. The `<html>` tag defines the root of the document, and the `lang` attribute specifies the language as English. The `<head>` section contains metadata such as character set, title, and links to external stylesheets (though the stylesheet file is missing). Styling: The styles for the page are expected to be defined in an external CSS file named "Distributor.css file.css," but the content of this file is not provided. Navigation: The navigation section (`<nav>`) contains links to different sections of the dashboard, such as Home, Profile, Settings, Help, and Logout. Main Content: The main content is divided into sections (`<section class="main">`). It includes a heading "DISTRIBUTOR'S DASHBOARD" and an icon representing user settings. The main skills section contains several cards with information and input fields. These cards include an overview of total sales, profit margin, customer satisfaction, on-time delivery rate, sales by region, product details (Vegetables, Fruits, Dairy, Grains and cereals), and wholesaler's details. Input Fields: The input fields are intended for users to enter specific data such as amounts in dollars, percentages, and details about products and wholesalers. Incomplete/Erroneous Parts: There are several issues in the code, such as incomplete input elements, missing CSS files, and redundant closing tags. These need to be addressed for the code to function properly. JavaScript (Not Present): The code doesn't include any JavaScript, which might be necessary for dynamic interactions or form validations. External Dependencies: The code references external fonts from Google Fonts, specifically "Hedvig Letters Sans," "Hedvig Letters Serif," and "Roboto."

Explanation of methodology of Distributor Profile Page: Document Declaration: The `<!DOCTYPE html>` declaration specifies that the document is an HTML5 document. HTML Structure: The `<html>` tag defines the root of the document, and the lang attribute is set to "en" for English. Metadata: The

`<head>` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. External Stylesheet: The code links an external stylesheet named "Distributor profile.css" to apply styles to the HTML elements. Body Content: The `<body>` section contains the main content of the web page. Profile Container: The main content is wrapped in a `<div>` with the class "profile-container." Profile Header: Within the "profile-container," there is a "profile-header" containing an image (presumably a profile picture) and the distributor's name. Profile Information: The "profile-info" section includes several paragraphs providing information about the distributor, such as: About Me: A brief description of the distributor's role and dedication to serving the company. Occupation: The distributor's job role. Location: The distributor's location (in this case, Dhaka). Contact: The distributor's email address. Styling: The styles for the page are expected to be defined in the external CSS file "Distributor profile.css." The actual content of this file is not provided.

Explanation of methodology of Distributor setting Page: Document Declaration: The `<!DOCTYPE html>` declaration specifies that the document is an HTML5 document. HTML Structure: The `<html>` tag defines the root of the document, and the lang attribute is set to "en" for English. Metadata: The `<head>` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. External Stylesheet: The code links an external stylesheet named "Distributor setting.css" to apply styles to the HTML elements. Body Content: The `<body>` section contains the main content of the settings page. Settings Container: The main content is wrapped in a `<div>` with the class "settings-container." Settings Heading: An `<h1>` heading displays "Settings." Settings List (Navigation): A navigation list (`<ul class="settings-list">`) provides links to different sections of the settings, such as Account, Notifications, and Privacy. Settings Content: The content is organized into a `<div>` with the class "settings-content." Account Settings: The first settings tab (`<div id="account" class="settings-tab">`) contains account-related settings. It includes a form with fields for email and password, along with a "Save Changes" button. Notification Settings: The second settings tab (`<div id="notifications" class="settings-tab">`) contains notification-related settings. It includes checkboxes for controlling email and push notifications. Privacy Settings: The third settings tab (`<div id="privacy" class="settings-tab">`) contains privacy-related settings. It includes a dropdown for profile visibility (Public/Private) and a checkbox for data sharing. Styling: The styles for the page are expected to be defined in the external CSS file "Distributor setting.css." The actual content of this file is not provided.

6 Wholesaler:

6.1 Screenshot:

The screenshot displays two windows side-by-side, both titled "WHOLESALER | By Code Info".

Wholesaler's Dashboard (Top Window):

- OVERVIEW:**
 - 1. Total Sales: Enter amount in \$
 - 2. Profit Margin in \$: Enter amount
 - 3. Customer satisfaction: Enter amount
- Sales By Country:**
 - 1. Bangladesh: Enter amount in \$
 - 2. India: Enter amount in \$
 - 3. Argentina: Enter amount in \$
- Available Products:**
 - A. Vegetables: Tomato: Enter amount
 - B. Potato: Enter amount
 - C. Cabbage: Enter amount
- Message From Distributor:** A red button labeled "See Message".

Wholesaler profile (Bottom Window):

- Profile Picture:** Indronil Dey Niloy
- About Me:** I am a countrywise Wholesaler. I have sold my product based on country.
- Occupation:** Wholesaler
- Location:** Dhaka
- Contact:** indronil4566@gmail.com



Settings

Account
Notifications
Privacy



Explanation of methodology of Wholesaler Dashboard Page: Document Declaration: The `<!DOCTYPE html>` declaration specifies that the document is an HTML5 document. HTML Structure: The `<html>` tag defines the root of the document, and the lang attribute is set to "en" for English. Metadata: The `<head>` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. External Stylesheet: The code links an external stylesheet named "wholesaler.css" to apply styles to the HTML elements. Font Imports: The code imports fonts from Google Fonts, including "Hedvig+Letters+Sans," "Hedvig+Letters+Serif," and "Roboto." Body Content: The `<body>` section contains the main content of the wholesaler's dashboard. Navigation: The navigation section (`<nav>`) contains links to different sections of the wholesaler's dashboard, such as Home, Profile, Settings, Help, Finance and Account, and Logout. Main Content: The main content is organized within a `<section class="main">` tag. It includes a top section with the wholesaler's name, an icon representing user settings, and a heading for the wholesaler's dashboard. Skills Section: The skills section (`<div class="main-skills">`) contains several cards with information and input fields. The cards include an overview of total sales, sales by country, available products (vegetables and fruits), a message from the distributor, and customer feedback. Input Fields and Buttons: Various input fields are provided for entering data, such as amounts in dollars and quantities for different products. "Add More" buttons suggest the ability to dynamically add more input fields for additional entries. Styling: The styles for the page are expected to be defined in the external CSS file "wholesaler.css." The actual content of this file is not provided. Incomplete/Erroneous Parts: There are some issues in the code, such as incomplete input elements, missing CSS files, and redundant closing tags. These need to be addressed for the code to function properly.

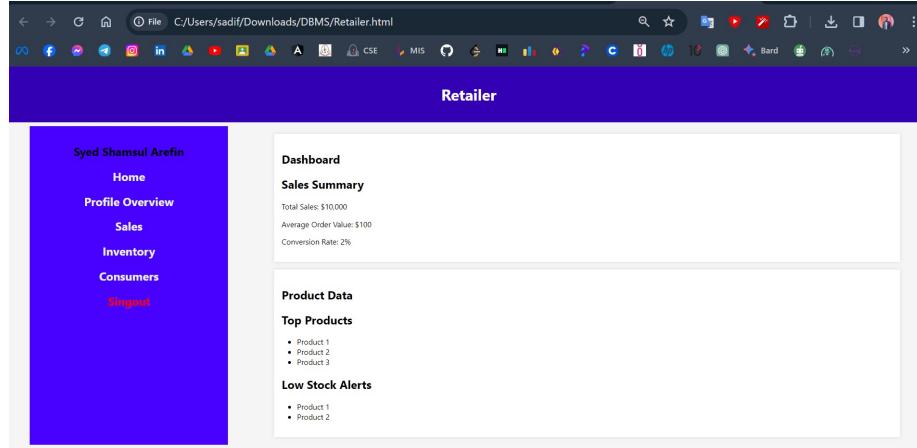
Explanation of methodology of Wholesaler Profile Page: Document Declaration: The `<!DOCTYPE html>` declaration specifies that the document is an

HTML5 document. HTML Structure: The `<html>` tag defines the root of the document, and the lang attribute is set to "en" for English. Metadata: The `<head>` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. External Stylesheet: The code links an external stylesheet named "Wholesaler profile.css" to apply styles to the HTML elements. Body Content: The `<body>` section contains the main content of the wholesaler's profile page. Profile Container: The main content is wrapped in a `<div>` with the class "profile-container." Profile Header: Within the "profile-container," there is a "profile-header" containing an image (presumably a profile picture) and the wholesaler's name. Profile Information: The "profile-info" section includes several paragraphs providing information about the wholesaler, such as: About Me: A brief description of the wholesaler's role and how they operate (country-wise wholesaler). Occupation: The wholesaler's job role. Location: The wholesaler's location (in this case, Dhaka). Contact: The wholesaler's email address. Styling: The styles for the page are expected to be defined in the external CSS file "Wholesaler profile.css." The actual content of this file is not provided.

Explanation of methodology of Wholesaler setting Page: Document Declaration: The `<!DOCTYPE html>` declaration specifies that the document is an HTML5 document. HTML Structure: The `<html>` tag defines the root of the document, and the lang attribute is set to "en" for English. Metadata: The `<head>` section contains metadata, including character set (UTF-8), viewport settings for responsiveness, and the title of the page. External Stylesheet: The code links an external stylesheet named "Wholesaler setting.css" to apply styles to the HTML elements. Body Content: Settings Container: The main content is wrapped in a `<div>` with the class "settings-container." Settings Heading: An `<h1>` heading displays "Settings." Settings List (Navigation): A navigation list (`<ul class="settings-list">`) provides links to different sections of the settings, such as Account, Notifications, and Privacy. Settings Content: The content is organized into a `<div>` with the class "settings-content." Account Settings Section: Account Settings Tab: The first settings tab (`<div id="account" class="settings-tab">`) contains account-related settings. It includes a form with fields for email and password, along with a "Save Changes" button. Notification Settings Section: Notification Settings Tab: The second settings tab (`<div id="notifications" class="settings-tab">`) contains notification-related settings. It includes checkboxes for controlling email and push notifications.

7 Retailer:

7.1 Screenshot:



HTML Structure: I started by creating an HTML document with the standard structure, including the doctype declaration and specifying the language as English. In the `<head>` section, I set the character encoding, viewport settings for responsiveness, and provided a title for the webpage, which is "Retailer." I linked an external stylesheet (Retailer.css) to keep the styling separate from the HTML for better maintainability. The `<body>` section contains the main content of the webpage.

Header Section (`<header>`): I created a header with an `<h1>` element displaying the title "Retailer." The header has a distinctive appearance with a purple background, white text, padding, and centered alignment.

Dashboard Container (`<div class="dashboard-container">`): To structure the layout, I used a container with the class dashboard-container. This container utilizes Flexbox properties to arrange its children (sidebar and main content) with space around them, providing a balanced layout.

Sidebar (`<div class="sidebar">`): Within the dashboard container, I created a sidebar with class sidebar. The sidebar has a purple background, white text, and padding for a visually appealing navigation panel. It contains links for Home, Profile Overview, Sales, Inventory, Consumers, and a Logout link.

Main Content (`<div class="main-content">`): The main content area takes up 70space for the dashboard's central information.

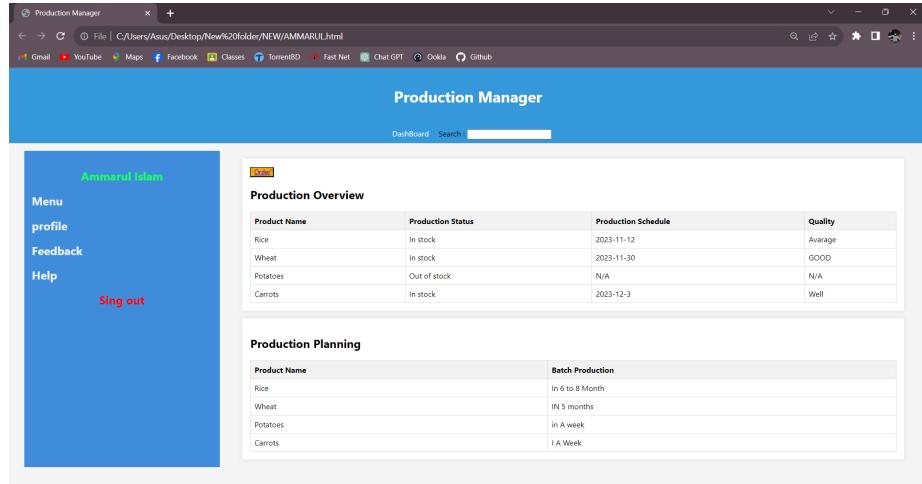
Sections (`<div class="section">`): I divided the main content into sections with the class section. Each section has a white background, a subtle box shadow for depth, padding for spacing, and margin for separation.

CSS Styling: The CSS styles enhance the visual presentation and layout of the webpage. I set the overall styling for the body, including the font family, background color, and removal of default margin and padding. The header has a distinctive purple background, white text, and padding for a visually appealing header. The sidebar has a purple background, white text, and padding, with a width of 20navigation panel. Flexbox is used to create a responsive and

balanced layout for the dashboard container and its children. Sections have a white background, a subtle box shadow for a border effect, padding for spacing, and margin for separation.

8 Retailer:

8.1 Screenshot:



HTML Structure: I started by creating an HTML document with the standard structure, including the doctype declaration and specifying the language as English. In the `<head>` section, I set the character encoding, viewport settings for responsiveness, and provided a title for the webpage, which is "Retailer." I linked an external stylesheet (Retailer.css) to keep the styling separate from the HTML for better maintainability. The `<body>` section contains the main content of the webpage. **Header Section (`<header>`):** I created a header with an `<h1>` element displaying the title "Retailer." The header has a distinctive appearance with a purple background, white text, padding, and centered alignment. **Dashboard Container (`<div class="dashboard-container">`):** To structure the layout, I used a container with the class dashboard-container. This container utilizes Flexbox properties to arrange its children (sidebar and main content) with space around them, providing a balanced layout. **Sidebar (`<div class="sidebar">`):** Within the dashboard container, I created a sidebar with class sidebar. The sidebar has a purple background, white text, and padding for a visually appealing navigation panel. It contains links for Home, Profile Overview, Sales, Inventory, Consumers, and a Logout link. **Main Content (`<div class="main-content">`):** The main content area takes up 70space for the dashboard's central information. **Sections (`<div class="section">`):** I divided the main content into sections with the class section. Each section has a white background, a subtle box shadow for depth, padding for spacing, and margin for separation.

CSS Styling: The CSS styles enhance the visual presentation and layout of the webpage. I set the overall styling for the body, including the font family, background color, and removal of default margin and padding. The header has a distinctive purple background, white text, and padding for a visually appealing header. The sidebar has a purple background, white text, and padding, with a width of 20navigation panel. Flexbox is used to create a responsive and balanced layout for the dashboard container and its children. Sections have a white background, a subtle box shadow for a border effect, padding for spacing, and margin for separation.