

Lesson 04: Enhance Grocery Frontend Development with Generative AI

Overview:

In this project, we delve into the practical application of generative AI prompts to expedite the frontend development process for a grocery e-commerce website.

By leveraging ChatGPT, we aim to streamline the creation of essential web pages such as contact, shop, and cart pages. The goal is to ensure seamless site navigation, all facilitated by XAMPP as the local server environment and PHP for backend scripting.

Instructions:

1. Navigate to ChatGPT and initiate the development process by providing specific prompts to create web pages
2. Follow the generated codebase to integrate the created pages into your website
3. Link the newly created pages with the homepage using prompts
4. Update the CSS by generating a new script through ChatGPT to fix the header and footer to the top and bottom of the screen, respectively

Tasks:

1. Create a contact page for the grocery website using ChatGPT
 - i. Provide a prompt to generate the contact page
 - ii. Link the contact page with the home page
2. Generate a cart page for the grocery website through ChatGPT
 - i. Use prompt to generate the cart page
3. Generate CSS script to fix header and footer
 - i. Ask ChatGPT to generate a CSS script to fix header and footer

Tools required:

1. ChatGPT
2. XAMPP

Guided Practice Solution

Disclaimer: Please note that all the GenAI tools used in this exercise can produce varied outputs even when presented with similar prompts. Thus, you may get different output for the same prompt.

Note: Refer to Lesson 04 Demo 06 for generating Grocery e-commerce web pages, including directory structure, style.css, index.php, header.php, and footer.php

Task 1: Create a Contact page for the Grocery website using ChatGPT

Step 1: Provide a prompt to generate the contact page

- 1.1 Run the following prompt in ChatGPT to generate a contact page:
Let's add a contact page for the Grocery project

P You
Let's add a contact page for the Grocery project

It generates two PHP scripts: contact page and contact_process page, as follows:

Create files **contact.php** and **contact_process.php** in the **../xampp/htdocs/myGrocery/** folder and then copy and paste the generated script into those files:

```
php Copy code

<?php include 'includes/header.php'; ?>

<main>
    <section class="contact-form">
        <h2>Contact Us</h2>
        <p>Have a question or feedback? Send us a message using the form below:</p>
        <form action="contact_process.php" method="post">
            <label for="name">Your Name:</label>
            <input type="text" id="name" name="name" required>

            <label for="email">Your Email:</label>
            <input type="email" id="email" name="email" required>

            <label for="message">Message:</label>
            <textarea id="message" name="message" rows="5" required></textarea>

            <button type="submit">Send Message</button>
        </form>
    </section>
</main>

<?php include 'footer.php'; ?>
```

../xampp/htdocs/myGrocery/contact.php

```
<?php include 'includes/header.php'; ?>
```

```
<main>
```

```
    <section class="contact-form">
```

```
        <h2>Contact Us</h2>
```

```
        <p>Have a question or feedback? Send us a message using the form below:</p>
```

```
        <form action="contact_process.php" method="post">
```

```
            <label for="name">Your Name:</label>
```

```
            <input type="text" id="name" name="name" required>
```

```
            <label for="email">Your Email:</label>
```

```
            <input type="email" id="email" name="email" required>
```

```
            <label for="message">Message:</label>
```

```
            <textarea id="message" name="message" rows="5" required></textarea>
```

```
            <button type="submit">Send Message</button>
```

```
        </form>
```

```
    </section>
```

</main>

<?php include 'footer.php'; ?>

php

Copy code

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve form data
    $name = $_POST['name'];
    $email = $_POST['email'];
    $message = $_POST['message'];

    // Here you can add the logic to process the form data,
    // such as sending an email to the store's contact address
    // or storing the message in a database.

    // For now, we'll simply print the submitted data
    echo "<h2>Thank you, $name!</h2>";
    echo "<p>Your message has been received. We'll get back to you soon.</p>";
}
?>
```

This code provides a contact page (Contact.php) where users can fill out a form to send a message. Upon submission, the form data is sent to contact_process.php for further processing. In contact_process.php, the form data can be handled accordingly, such as sending an email or storing in a database.

↓

../xampp/htdocs/myGrocery/contact_process.php

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Retrieve form data
    $name = $_POST['name'];
    $email = $_POST['email'];
    $message = $_POST['message'];

    // Here you can add the logic to process the form data,
    // such as sending an email to the store's contact address
    // or storing the message in a database.
```

```
// For now, we'll simply print the submitted data
echo "<h2>Thank you, $name!</h2>";
echo "<p>Your message has been received. We'll get back to you soon.</p>";
}
?>
```

Step 2: Link the contact page with the home page

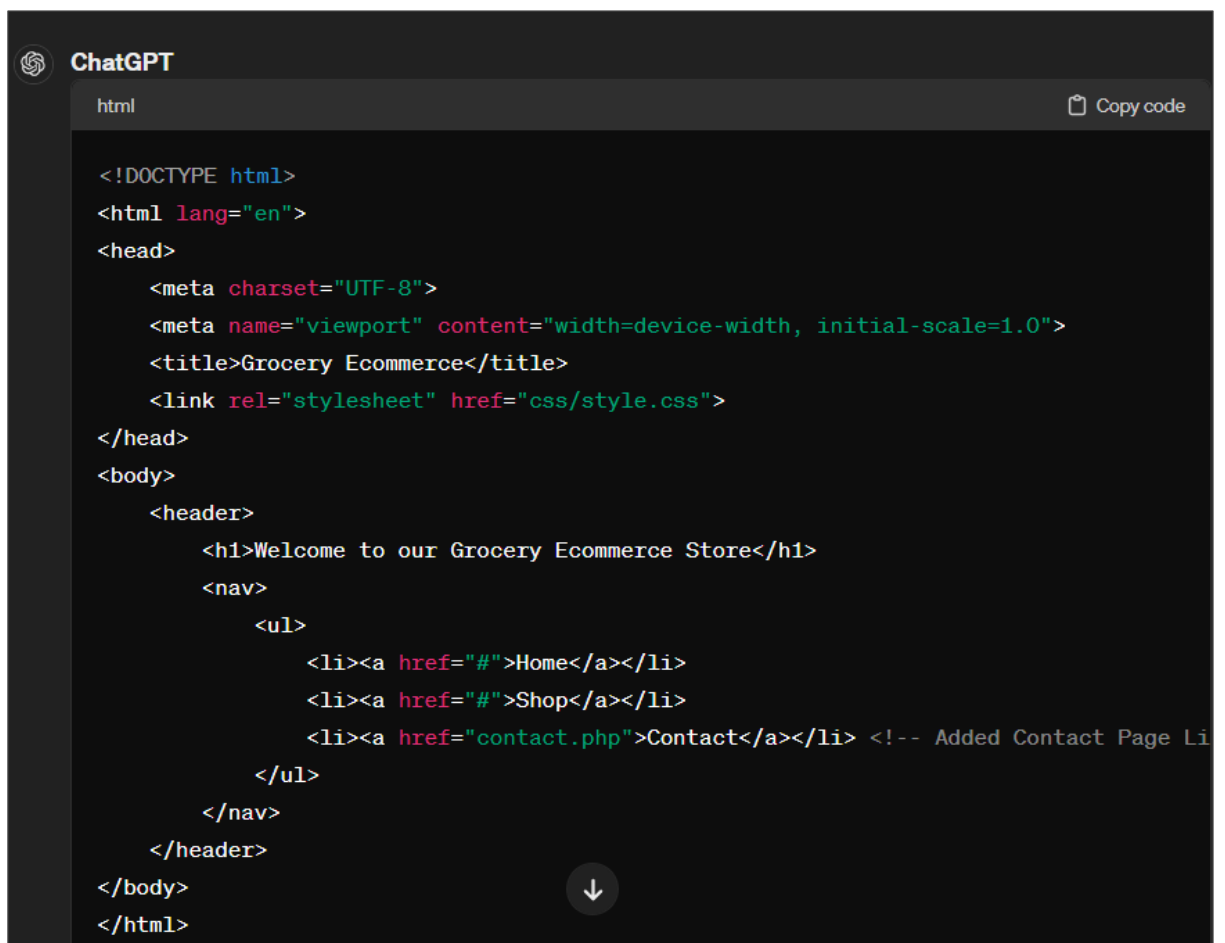
2.1 Run the following prompt to link the generated contact page with the home page:

Let's add contact with the homepage

P You
Let's add contact with the homepage

This prompt generates the following HTML:

Copy this content and replace it with the existing content in the `../xampp/htdocs/myGrocery/includes/header.php` file



The screenshot shows a ChatGPT interface with a dark theme. The chat window displays a prompt from the user and a response from the AI. The response is a complete HTML header for a website. The code is as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Ecommerce</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">Shop</a></li>
        <li><a href="contact.php">Contact</a></li> <!-- Added Contact Page Li
      </ul>
    </nav>
  </header>
</body>
</html>
```

The interface includes a "Copy code" button in the top right corner of the chat window and a downward arrow button at the bottom of the code block.

`../xampp/myGrocery/htdocs/includes/header.php`

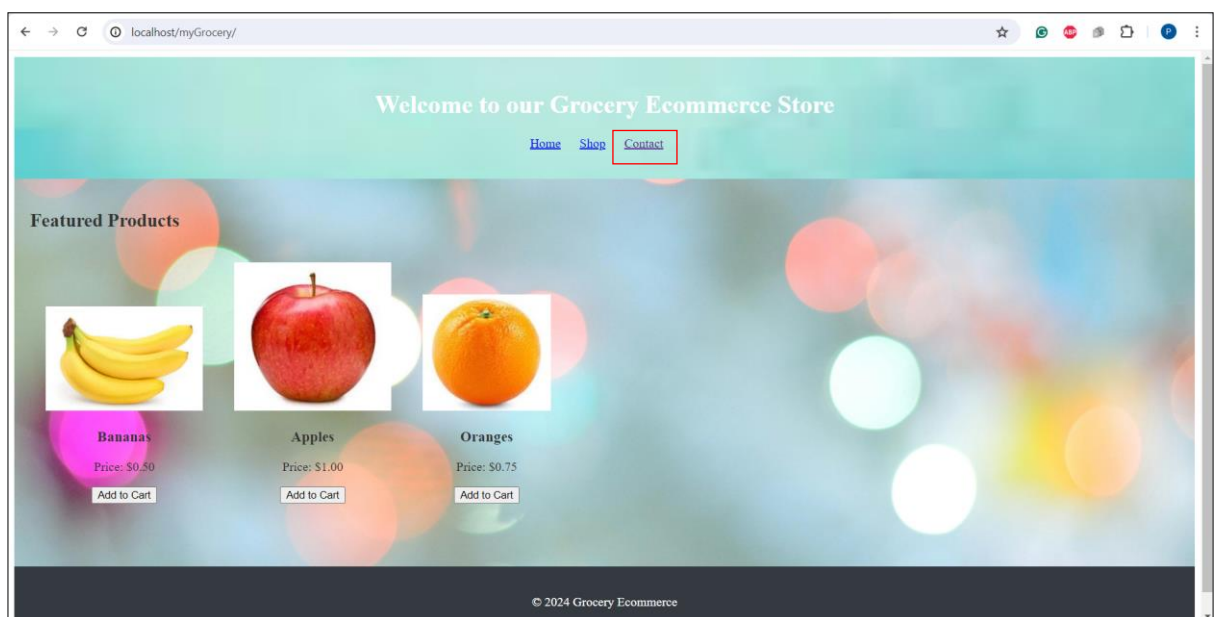
```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Ecommerce</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="#">Home</a></li>
        <li><a href="#">Shop</a></li>
        <li><a href="contact.php">Contact</a></li> <!-- Added Contact Page Link -->
      </ul>
    </nav>
  </header>
</body>
</html>

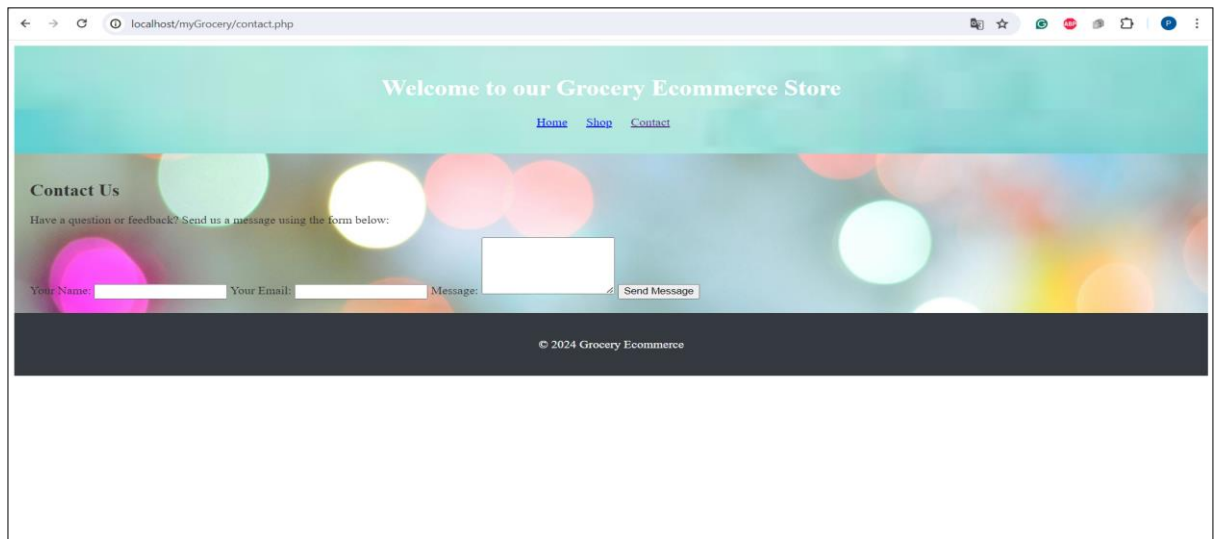
```

Note: Refer to Lesson 04 Demo 06 for running XAMPP services and executing the myGrocery project.

2.2 Open the XAMPP control panel and start the Tomcat service. In the browser, provide the following link: **http://localhost/myGrocery**. The output appears as follows:

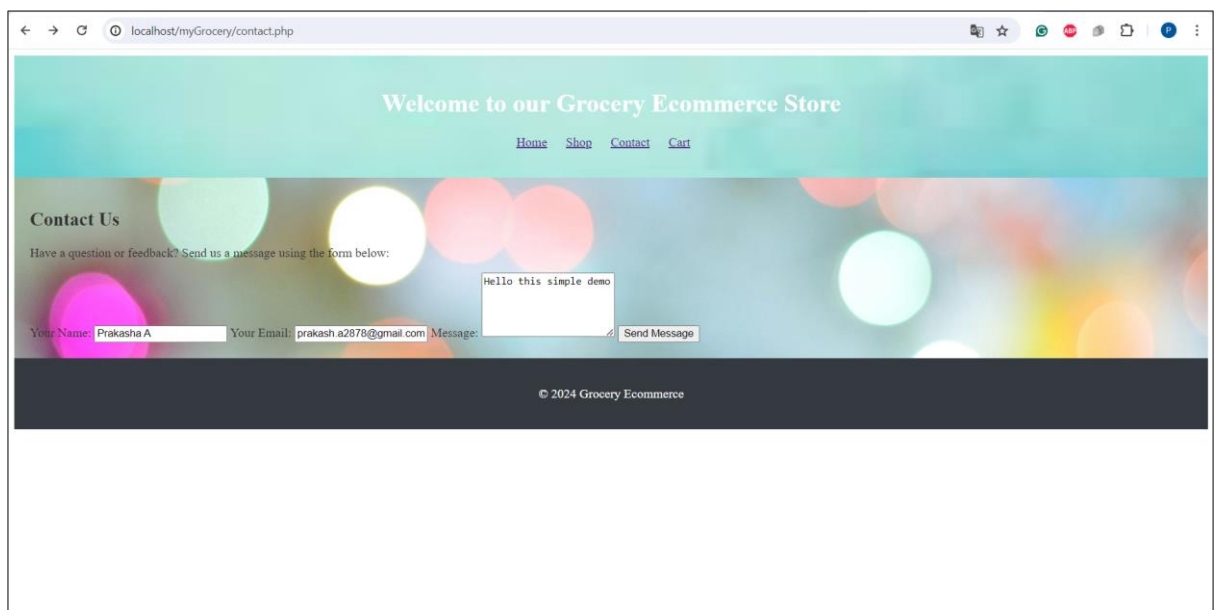


2.3 Click on **Contact** to navigate to the contact page



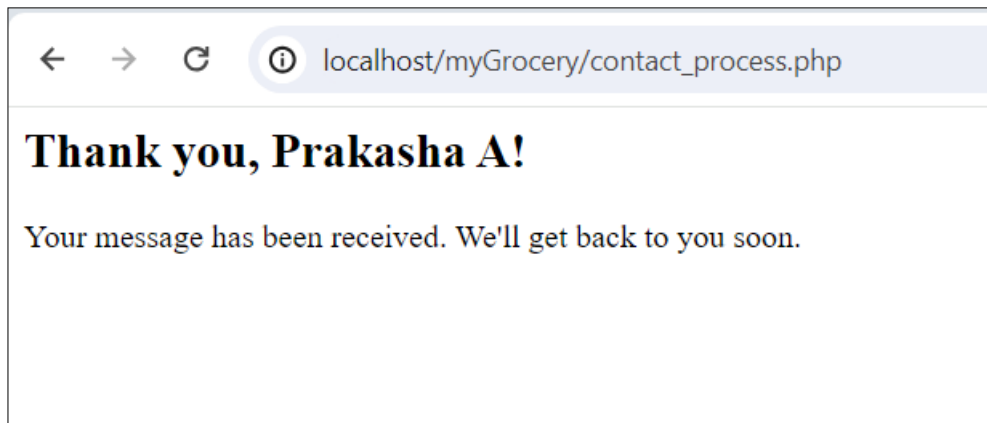
A screenshot of a web browser displaying the contact page of a grocery e-commerce store. The browser's address bar shows 'localhost/myGrocery/contact.php'. The page has a teal header with the text 'Welcome to our Grocery Ecommerce Store' and navigation links for 'Home', 'Shop', and 'Contact'. Below the header, the section is titled 'Contact Us' with the instruction 'Have a question or feedback? Send us a message using the form below:'. The form consists of three input fields: 'Your Name:', 'Your Email:', and 'Message:'. The 'Send Message' button is to the right of the message field. The footer is a dark grey bar with the text '© 2024 Grocery Ecommerce'.

2.4 Provide name, email, and message and then click on the **Send Message** button to send the information, as shown in the screenshot below:



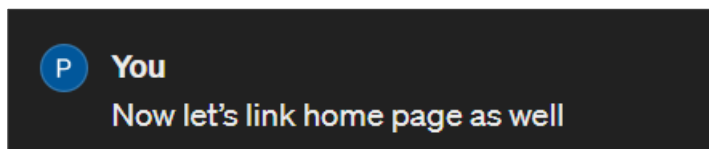
A screenshot of the same contact page, but with the form fields filled out. The 'Your Name:' field contains 'Prakasha A', the 'Your Email:' field contains 'prakash.a2878@gmail.com', and the 'Message:' field contains 'Hello this simple demo'. The 'Send Message' button remains to the right of the message field. The rest of the page layout, including the header and footer, is identical to the previous screenshot.

The following page shows the successful received message:



2.5 Use the following prompt to link the `../xampp/htdocs/myGrocery/index.php` file with the home page:

Now let's link home page as well



This generates the following script:

Update header.php file in `../xampp/htdocs/myGrocery/includes/header.php` by linking the home page by adding index.php

ChatGPT

html

Copy code

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Ecommerce</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="index.php">Home</a></li> <!-- Modified Home Link -->
        <li><a href="#">Shop</a></li>
        <li><a href="contact.php">Contact</a></li>
      </ul>
    </nav>
  </header>
</body>
</html>
```

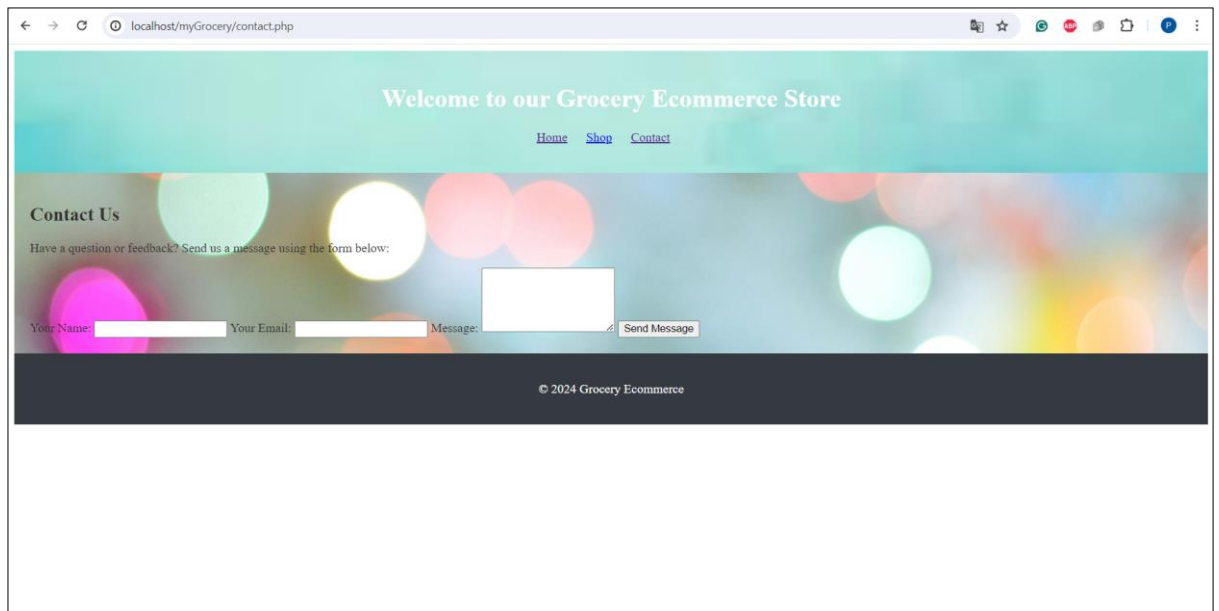
↓

../xampp/htdocs/myGrocery/includes/header.php

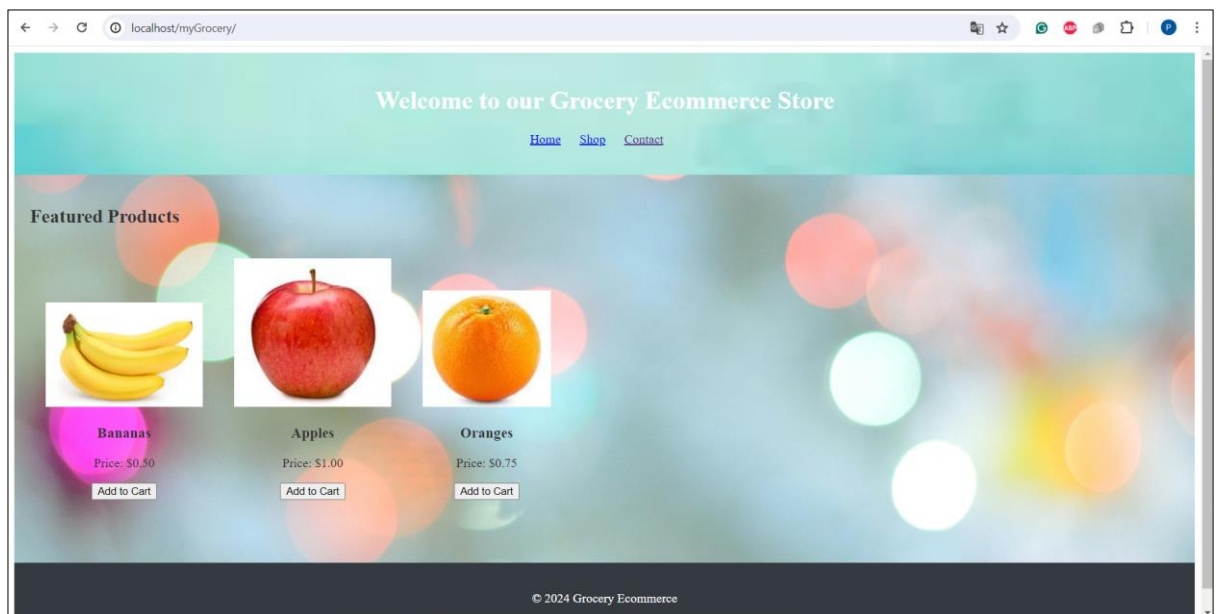
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Ecommerce</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="index.php">Home</a></li> <!-- Modified Home Link -->
        <li><a href="#">Shop</a></li>
        <li><a href="contact.php">Contact</a></li>
      </ul>
    </nav>
```

```
</header>
</body>
</html>
```

The output shows that clicking on **Home** on the contact page; it will navigate back to the **Home** page.



The home page after navigated from the Contact page will look as follows:



Task 2: Generate Cart page for the Grocery website through ChatGPT

Step 1: Use prompt to generate the cart page

- 1.1 Run the following prompt in ChatGPT to generate a cart page that shows the items in the cart:

Let's try adding another page for the cart and link it with another necessary page

You
Let's try adding another page for the cart and link it with another necessary page.

This generates the following PHP file. Create a new file called **cart.php** in the **../xampp/htdocs/myGrocery/** and copy and paste the generated file into **cart.php**

```
php Copy code

<?php include 'includes/header.php'; ?>

<main>
    <section class="cart">
        <h2>Your Cart</h2>
        <div class="cart-items">
            <!-- Display cart items here -->
            <p>No items in your cart.</p>
        </div>
        <div class="cart-summary">
            <!-- Display subtotal, checkout button, etc. here -->
            <p>Subtotal: $0.00</p>
            <button>Checkout</button>
        </div>
    </section>
</main>

<?php include 'footer.php'; ?>
```

../xampp/htdpes/ myGrocery/cart.php

```
<?php include 'includes/header.php'; ?>
```

```
<main>
    <section class="cart">
        <h2>Your Cart</h2>
        <div class="cart-items">
```

```

        <!-- Display cart items here -->
        <p>No items in your cart.</p>
    </div>
    <div class="cart-summary">
        <!-- Display subtotal, checkout button, etc. here -->
        <p>Subtotal: $0.00</p>
        <button>Checkout</button>
    </div>
</section>
</main>

<?php include 'footer.php'; ?>

```

Link this generated cart page with the home page by updating the header.php file.
 Replace the following script in the
../xampp/htdocs/myGrocery/includes/header.php file

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Grocery Ecommerce</title>
    <link rel="stylesheet" href="css/style.css">
</head>
<body>
    <header>
        <h1>Welcome to our Grocery Ecommerce Store</h1>
        <nav>
            <ul>
                <li><a href="index.php">Home</a></li>
                <li><a href="#">Shop</a></li> <!-- Changed Shop Page Link to # -->
                <li><a href="contact.php">Contact</a></li>
                <li><a href="cart.php">Cart</a></li>
            </ul>
        </nav>
    </header>
</body>
</html>

```

../xampp/htdocs/myGrocery/cart.php

```

<!DOCTYPE html>
<html lang="en">
<head>

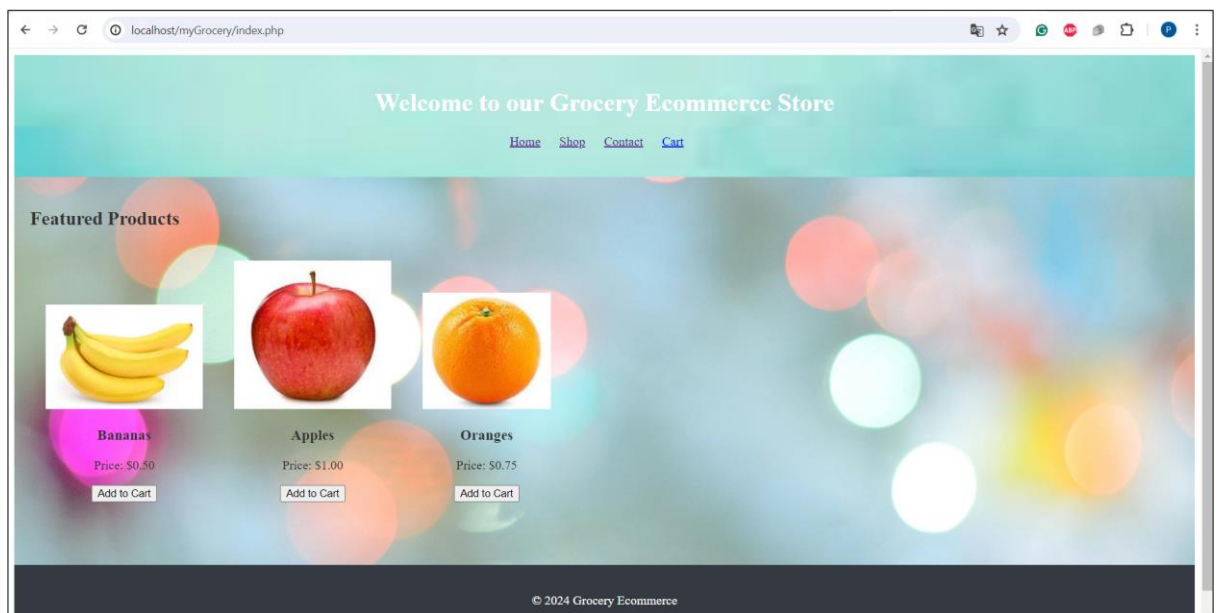
```

```

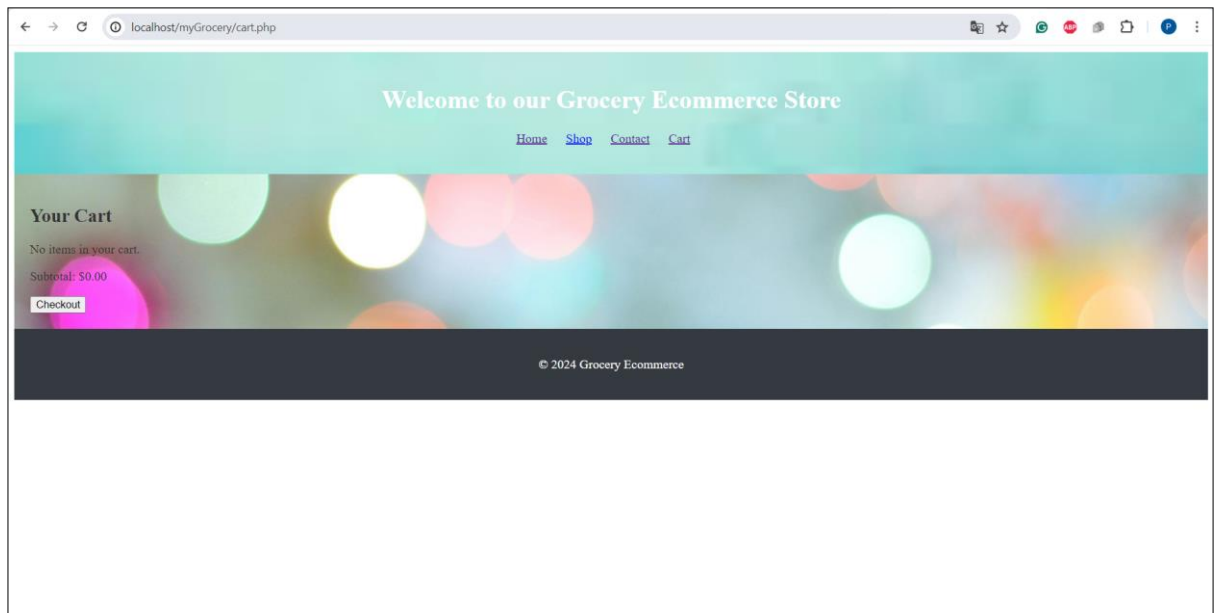
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Grocery Ecommerce</title>
<link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="index.php">Home</a></li>
        <li><a href="#">Shop</a></li> <!-- Changed Shop Page Link to # -->
        <li><a href="contact.php">Contact</a></li>
        <li><a href="cart.php">Cart</a></li>
      </ul>
    </nav>
  </header>
</body>
</html>

```

The page shows that when clicking on **Cart** in the Home page, it will navigate to the Cart page as follows:



The cart page appears as follows:



1.2 Input the following prompt to link the shop page to the home page:

Let's try to add a shop page by linking it to index.php

P You

Let's try to add a shop page by linking it to index.php

It generates the following script:

Copy and paste the generated script into the existing file in
../xampp/htdocs/myGrocery/includes/header.php

```
html Copy code

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Ecommerce</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="index.php">Home</a></li> <!-- Modified Home Link -->
        <li><a href="index.php">Shop</a></li> <!-- Changed Shop Page Link to -->
        <li><a href="contact.php">Contact</a></li>
        <li><a href="cart.php">Cart</a></li> <!-- Added Cart Page Link -->
      </ul>
    </nav>
  </header>
</body>
</html>
```

../xampp/htdocs/myGrocery/includes/header.php

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grocery Ecommerce</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <header>
    <h1>Welcome to our Grocery Ecommerce Store</h1>
    <nav>
      <ul>
        <li><a href="index.php">Home</a></li> <!-- Modified Home Link -->
```

```
        <li><a href="index.php">Shop</a></li> <!-- Changed Shop Page Link to
index.php -->
        <li><a href="contact.php">Contact</a></li>
        <li><a href="cart.php">Cart</a></li> <!-- Added Cart Page Link -->
    </ul>
</nav>
</header>
</body>
</html>
```

Task 3: Generate CSS script to fix header and footer

Step 1: Ask ChatGPT to generate a CSS script to fix header and footer

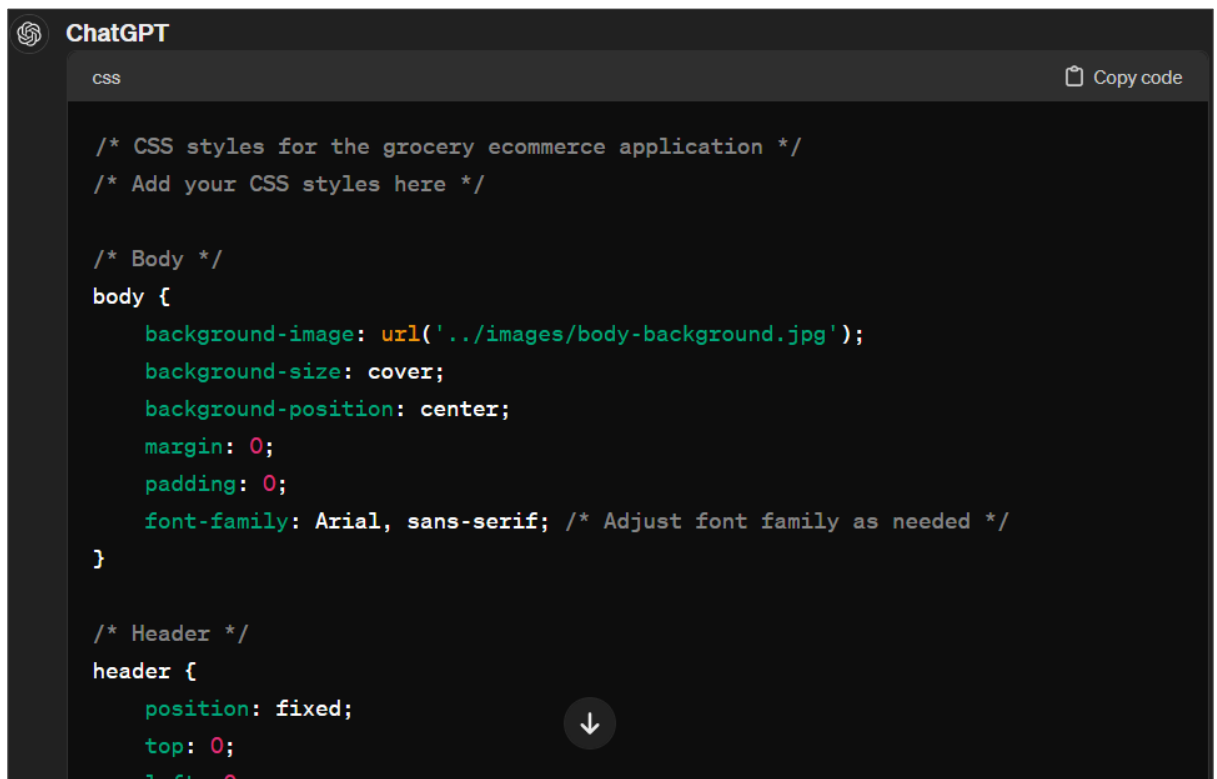
- 1.1 Input the following prompt to fix the header and footer to the extreme top and bottom, respectively

Let's fix the header and footer to extreme top and extreme bottom respectively

P You

Let's fix the header and footer to extreme top and extreme bottom respectively

This generates the following CSS script:
Copy the generated script and replace it with the existing one in the
../xampp/htdocs/myGrocery/css/style.css file

A screenshot of a ChatGPT chat window. The header shows the ChatGPT logo and name. Below the header is a dark grey bar with the text 'CSS' on the left and a 'Copy code' button on the right. The main area contains a code block with CSS code for a grocery e-commerce application. The code includes comments and styles for the body and header elements. A circular button with a downward arrow is visible at the bottom right of the code area.

```
/* CSS styles for the grocery ecommerce application */
/* Add your CSS styles here */

/* Body */
body {
    background-image: url('../images/body-background.jpg');
    background-size: cover;
    background-position: center;
    margin: 0;
    padding: 0;
    font-family: Arial, sans-serif; /* Adjust font family as needed */
}

/* Header */
header {
    position: fixed;
    top: 0;
    left: 0;
```

```
    left: 0;
    width: 100%;
    background-color: rgba(0, 0, 0, 0.8); /* Add transparency to make header text mor
    padding: 20px;
    text-align: center;
    color: #ffffff; /* Adjust text color for better readability */
    z-index: 1000; /* Ensure header is above other content */
}

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

nav ul li {
    display: inline;
    margin-right: 20px;
}

/* Main */
main {
    padding: 20px;
    margin-top: 80px; /* Ensure content starts below the fixed header */
    margin-bottom: 80px; /* Ensure content ends above the fixed footer */
    background-color: rgba(0, 0, 0, 0.8); /* Add transparency to make content =
```


```
margin-top: 80px; /* Ensure content starts below the fixed header */
margin-bottom: 80px; /* Ensure content ends above the fixed footer */
background-color: rgba(255, 255, 255, 0.8); /* Add transparency to make content m
}

.featured-products {
  margin-bottom: 40px;
}

.product {
  display: inline-block;
  margin: 20px;
  text-align: center;
}

.product img {
  width: 100%;
  max-width: 200px;
  height: auto;
}

/* Footer */
footer {
  position: fixed;
```



```

    bottom: 0;
    left: 0;
    width: 100%;
    background-color: rgba(0, 0, 0, 0.8); /* Add transparency to make footer text mor
    color: #ffffff;
    padding: 20px;
    text-align: center;
    z-index: 1000; /* Ensure footer is above other content */
}

/* Responsive Styles */
@media screen and (max-width: 768px) {
    nav ul li {
        display: block;
        margin-bottom: 10px;
    }

    .product {
        width: 45%;
        margin: 2.5%;
    }
}

```

../xampp/htdocs/myGrocery/css/style.css

```

/* CSS styles for the grocery ecommerce application */
/* Add your CSS styles here */

/* Body */
body {
    background-image: url('../images/body-background.jpg');
    background-size: cover;
    background-position: center;
    margin: 0;
    padding: 0;
    font-family: Arial, sans-serif; /* Adjust font family as needed */
}

/* Header */
header {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;

```

```
        background-color: rgba(0, 0, 0, 0.8); /* Add transparency to make header text
more readable */
        padding: 20px;
        text-align: center;
        color: #ffffff; /* Adjust text color for better readability */
        z-index: 1000; /* Ensure header is above other content */
    }

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

    nav ul li {
        display: inline;
        margin-right: 20px;
    }

    /* Main */
    main {
        padding: 20px;
        margin-top: 80px; /* Ensure content starts below the fixed header */
        margin-bottom: 80px; /* Ensure content ends above the fixed footer */
        background-color: rgba(255, 255, 255, 0.8); /* Add transparency to make content
more readable */
    }

    .featured-products {
        margin-bottom: 40px;
    }

    .product {
        display: inline-block;
        margin: 20px;
        text-align: center;
    }

    .product img {
        width: 100%;
        max-width: 200px;
        height: auto;
    }

    /* Footer */
    footer {
        position: fixed;
```

```

bottom: 0;
left: 0;
width: 100%;
background-color: rgba(0, 0, 0, 0.8); /* Add transparency to make footer text more
readable */
color: #ffffff;
padding: 20px;
text-align: center;
z-index: 1000; /* Ensure footer is above other content */
}

/* Responsive Styles */
@media screen and (max-width: 768px) {
  nav ul li {
    display: block;
    margin-bottom: 10px;
  }

  .product {
    width: 45%;
    margin: 2.5%;
  }
}

```

The output after updating the CSS script is as follows:

