Digital Assessment

NAME – KARTIK BATLA

REG. NO – 23BBS0017

COURSE NAME – DATABASE SYSTEMS

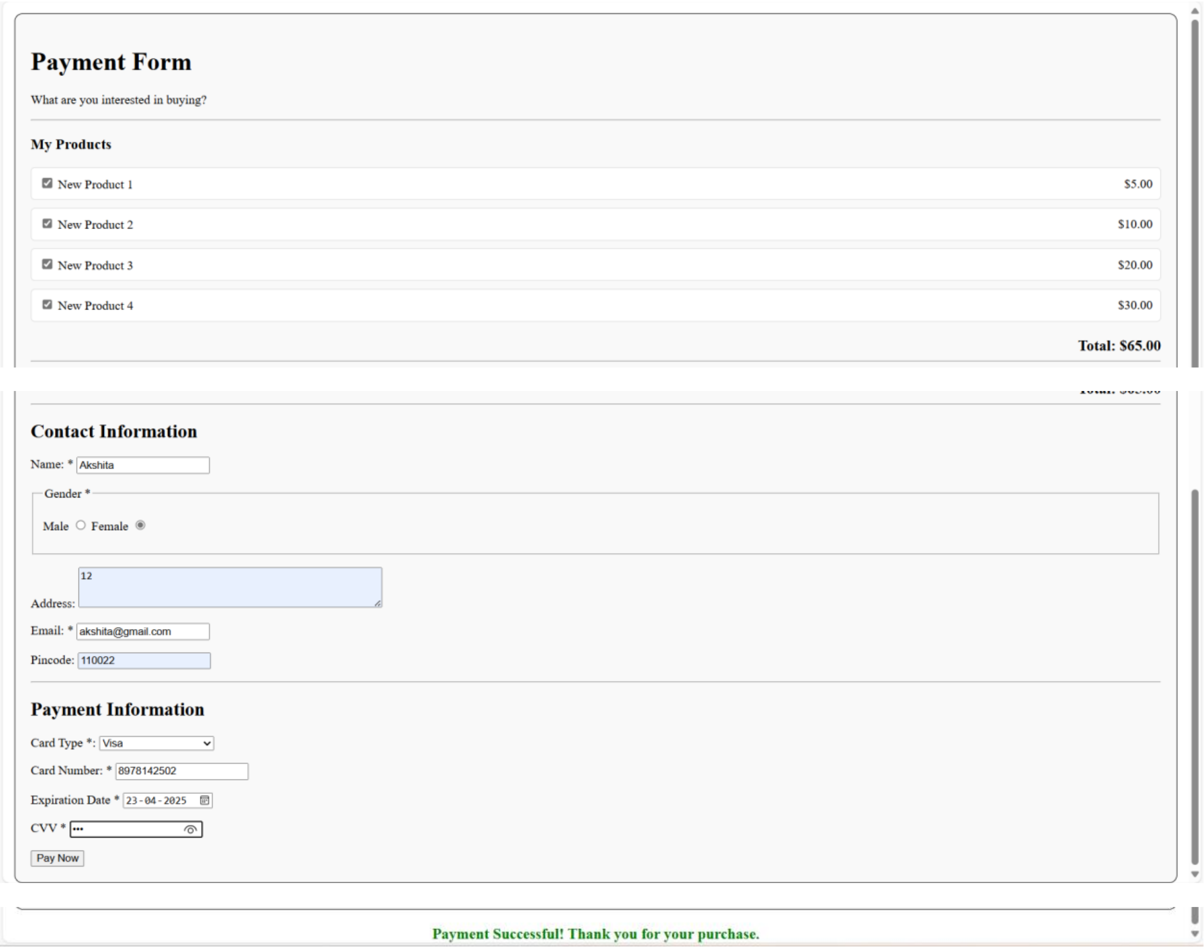
COURSE CODE - 1007

🖥️ Frontend (HTML, CSS, JavaScript)

The **frontend** is responsible for displaying the payment form and handling user interactions.

- **Structure:** Built using **HTML** for layout, **CSS** for styling, and **JavaScript** for functionality.
- **Features:**
    - A list of selectable products with **dynamic price calculation**.
    - A **real-time total price update** as users select/deselect products.
    - A form for **user details and card payment information**.
    - **Automatic saving of selected products and amount** using JavaScript & Fetch API.
    - **Form validation** to ensure correct data input before submission.

PROTOTYPE:



Payment Form

What are you interested in buying?

My Products

☑ New Product 1	$5.00
☑ New Product 2	$10.00
☑ New Product 3	$20.00
☑ New Product 4	$30.00

Total: $65.00

Contact Information

Name: * Akshita

Gender *

Male ○ Female ●

Address: 12

Email: * akshita@gmail.com

Pincode: 110022

Payment Information

Card Type *: Visa

Card Number: * 8978142502

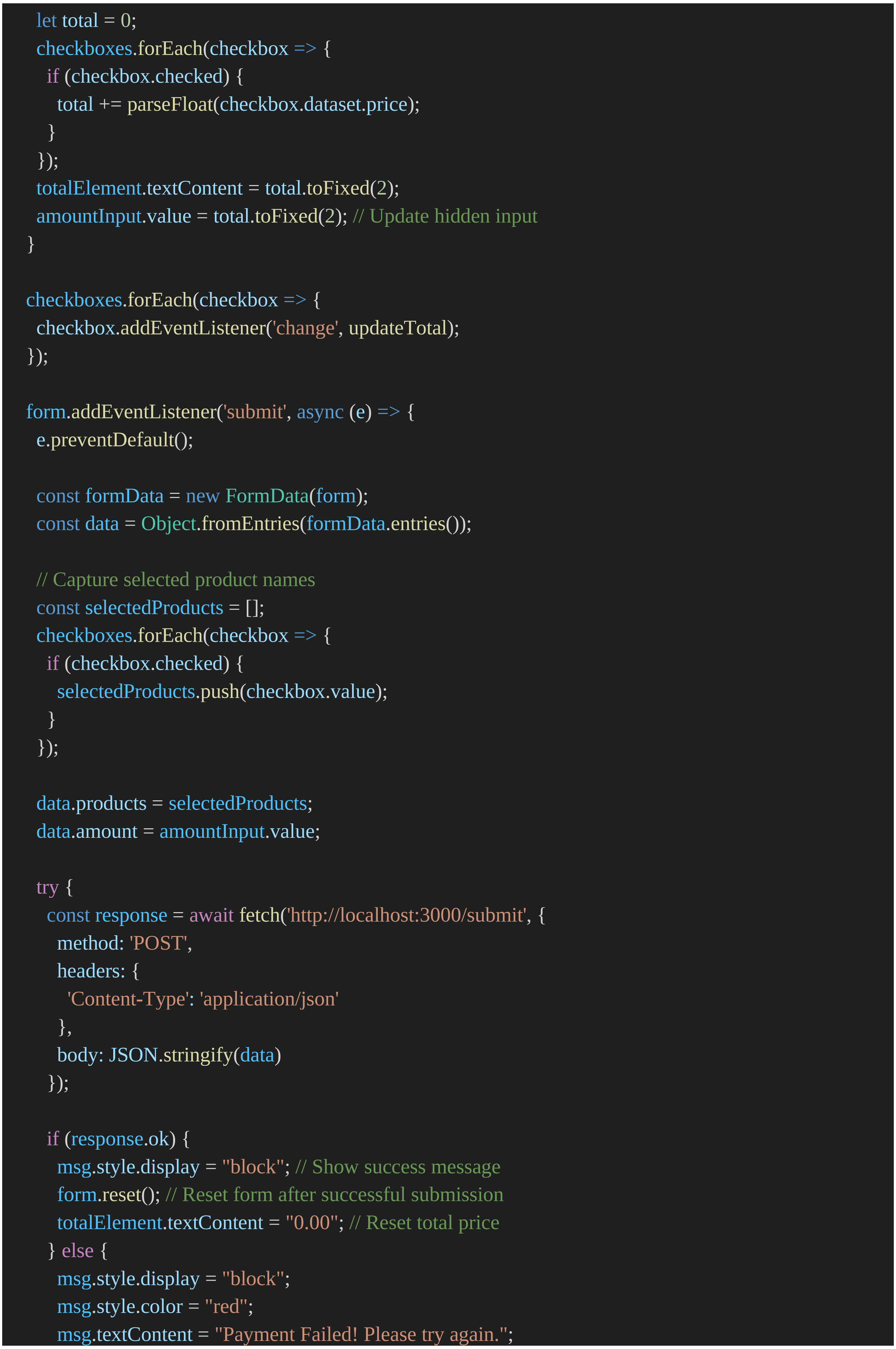
Expiration Date * 23-04-2025

CVV * •••

Payment Successful! Thank you for your purchase.

CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Payment Form</title>
  <style>
    body {
      font-family: "Times New Roman", Times, serif;
      background-color: white;
      margin: 0;
      padding: 15px;
      height: 100vh;
    }
    .container {
      max-width: 2000px;
      margin: auto;
      padding: 20px;
      border: 1px solid black;
      border-radius: 10px;
      background-color: #f9f9f9;
    }
    .product {
      display: flex;
      justify-content: space-between;
      align-items: center;
      padding: 10px;
      border: 1px solid #ddd;
      border-radius: 5px;
      margin-bottom: 10px;
      background: #fff;
    }
    .total {
      font-weight: bold;
      font-size: 1.2em;
      text-align: right;
      margin-top: 20px;
    }
    .success-message {
      display: none;
      color: green;
      font-size: 1.2em;
      font-weight: bold;
      text-align: center;
      margin-top: 20px;
    }
  </style>
</head>
<body>

  <div class="container">
    <form id="paymentForm">
      <h1>Payment Form</h1>
      <p>What are you interested in buying?</p>
      <hr>

      <!-- Product Selection -->
      <div id="products">
        <h3>My Products</h3>
        <div class="product">
          <label>
            <input type="checkbox" name="products" value="New Product 1" data-price="5"> New Product 1
          </label>
          <span>$5.00</span>
        </div>
        <div class="product">
          <label>
            <input type="checkbox" name="products" value="New Product 2" data-price="10"> New Product
2
          </label>
          <span>$10.00</span>
        </div>
        <div class="product">
          <label>
            <input type="checkbox" name="products" value="New Product 3" data-price="20"> New Product
3
          </label>
          <span>$20.00</span>
        </div>
        <div class="product">
          <label>
            <input type="checkbox" name="products" value="New Product 4" data-price="30"> New Product
4
          </label>
          <span>$30.00</span>
        </div>
      </div>

      <!-- Total Amount -->
      <div class="total">
        Total: $<span id="total">0.00</span>
      </div>
    <hr>
    <h2>Contact Information</h2>
    <p>Name: * <input type="text" name="full_name" required></p>
    <fieldset>
      <legend>Gender *</legend>
      <p>
        Male <input type="radio" name="gender" value="Male" required>
        Female <input type="radio" name="gender" value="Female" required>
      </p>
    </fieldset>
    <p>Address: <textarea name="address" cols="50" rows="3"></textarea></p>
    <p>Email: * <input type="email" name="email" required></p>
    <p>Pincode: <input type="number" name="pincode"></p>

    <hr>
    <h2>Payment Information</h2>
    <p>
      Card Type *:
      <select name="card_type" required>
        <option value="">--Select a card type--</option>
        <option value="Visa">Visa</option>
        <option value="MasterCard">MasterCard</option>
        <option value="Rupay">Rupay</option>
      </select>
    </p>
    <p>Card Number: * <input type="number" name="card_number" required></p>
    <p>Expiration Date * <input type="date" name="exp_date" required></p>
    <p>CVV * <input type="password" name="cvv" required></p>

    <!-- Hidden Input for Total Amount -->
    <input type="hidden" name="amount" id="amount">

    <input type="submit" value="Pay Now">
  </form>
</div>

<!-- Success Message -->
<p id="message" class="success-message">Payment Successful! Thank you for your purchase.</p>

<script>
  const checkboxes = document.querySelectorAll('#products input[type="checkbox"]');
  const totalElement = document.getElementById('total');
  const amountInput = document.getElementById('amount');
  const form = document.getElementById('paymentForm');
  const msg = document.getElementById('message');

  function updateTotal() {
```



```
  let total = 0;
  checkboxes.forEach(checkbox => {
    if (checkbox.checked) {
      total += parseFloat(checkbox.dataset.price);
    }
  });
  totalElement.textContent = total.toFixed(2);
  amountInput.value = total.toFixed(2); // Update hidden input
}

checkboxes.forEach(checkbox => {
  checkbox.addEventListener('change', updateTotal);
});

form.addEventListener('submit', async (e) => {
  e.preventDefault();

  const formData = new FormData(form);
  const data = Object.fromEntries(formData.entries());

  // Capture selected product names
  const selectedProducts = [];
  checkboxes.forEach(checkbox => {
    if (checkbox.checked) {
      selectedProducts.push(checkbox.value);
    }
  });

  data.products = selectedProducts;
  data.amount = amountInput.value;

  try {
    const response = await fetch('http://localhost:3000/submit', {
      method: 'POST',
      headers: {
        'Content-Type': 'application/json'
      },
      body: JSON.stringify(data)
    });

    if (response.ok) {
      msg.style.display = "block"; // Show success message
      form.reset(); // Reset form after successful submission
      totalElement.textContent = "0.00"; // Reset total price
    } else {
      msg.style.display = "block";
      msg.style.color = "red";
      msg.textContent = "Payment Failed! Please try again.";
```

```
        }
      } catch (error) {
        msg.style.display = "block";
        msg.style.color = "red";
        msg.textContent = "Error! Unable to process payment.";
      }
    });
  </script>

</body>
</html>
```

🔗 Backend (Node.js, Express)

The **backend** acts as a bridge between the frontend and the database.

- **Built using: Node.js** with **Express.js** for handling API requests.
- **Endpoints:**
    1. POST /save-selection → Saves selected products & amount (before form submission).
    2. POST /submit-payment → Saves **user & payment details** to MongoDB after form submission.
- **Data Processing:** Receives JSON data from the frontend, processes it, and stores it in the database.
- **Security Consideration:** Since this is a basic implementation, actual **payment processing** (like Stripe or Razorpay integration) is not included.

CODE:

```
const express = require('express');
const bodyParser = require('body-parser');
const mysql = require('mysql');
const cors = require('cors');

const app = express();
const PORT = 3000;

// Middleware
app.use(cors());
app.use(bodyParser.json());

// MySQL Connection
const db = mysql.createConnection({
  host: 'localhost',
  user: 'root',          // replace with your MySQL username
  password: '08021518',          // replace with your MySQL password
  database: 'payment_form'
});

db.connect(err => {
  if (err) throw err;
  console.log('Connected to MySQL Database');
});

// Route: Submit Payment Form
```

```
app.post('/submit', (req, res) => {
  const {
    product_name, amount, full_name, gender,
    address, email, pincode,
    card_type, card_number, exp_date, cvv
  } = req.body;

  const query = `INSERT INTO payments
    (product_name, amount, full_name, gender, address, email, pincode, card_type, card_number, exp_date, cvv)
    VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)`;

  const values = [product_name, amount, full_name, gender, address, email, pincode, card_type, card_number, exp_date, cvv];

  db.query(query, values, (err, result) => {
    if (err) {
      console.error(err);
      res.status(500).send('Failed to save payment!');
    } else {
      res.send('Payment submitted successfully!');
    }
  });
});

// Start server
app.listen(PORT, () => {
  console.log(`Server running at http://localhost:${PORT}`);
});
```

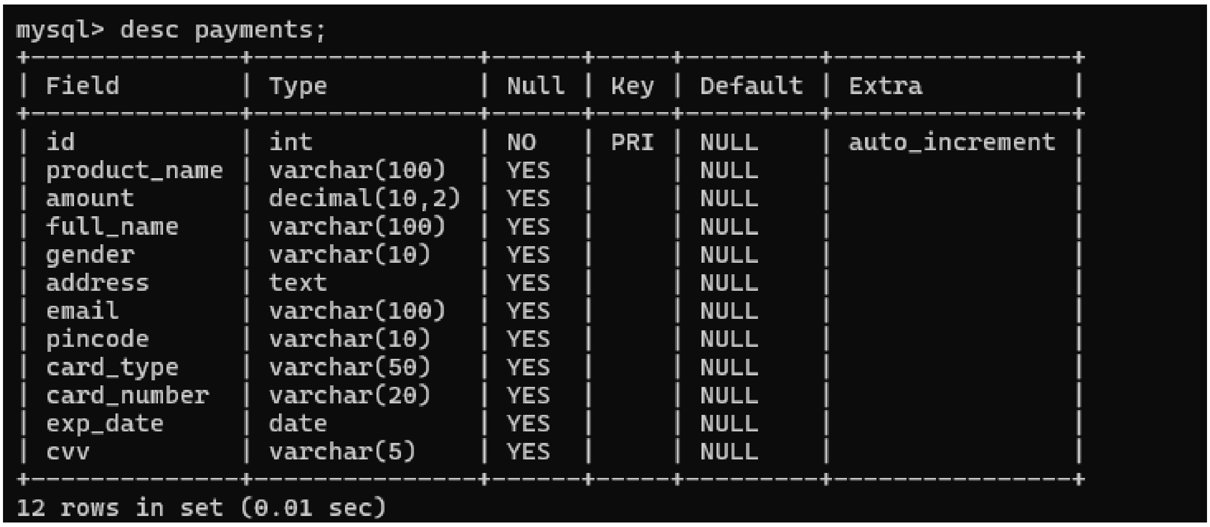
🗄️ Database (MySQL)

The database payment_form contains a single table: payments, which is designed to store payment details when a user completes a transaction.

CODE FOR TABLE CREATION:

```
CREATE DATABASE payment_form;
USE payment_form;

CREATE TABLE payments (
   id INT AUTO_INCREMENT PRIMARY KEY,
   product_name VARCHAR(100),
   amount DECIMAL(10,2),
   full_name VARCHAR(100),
   gender VARCHAR(10),
   address TEXT,
   email VARCHAR(100),
   pincode VARCHAR(10),
   card_type VARCHAR(50),
   card_number VARCHAR(20),
   exp_date DATE,
   cvv VARCHAR(5)
);
```

PAYMENTS TABLE:



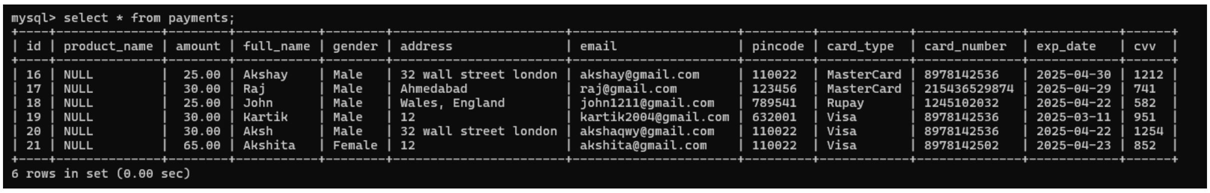
```
mysql> desc payments;
+--------------+---------------+------+-----+---------+----------------+
| Field        | Type          | Null | Key | Default | Extra          |
+--------------+---------------+------+-----+---------+----------------+
| id           | int           | NO   | PRI | NULL    | auto_increment |
| product_name | varchar(100)  | YES  |     | NULL    |                |
| amount       | decimal(10,2) | YES  |     | NULL    |                |
| full_name    | varchar(100)  | YES  |     | NULL    |                |
| gender       | varchar(10)   | YES  |     | NULL    |                |
| address      | text          | YES  |     | NULL    |                |
| email        | varchar(100)  | YES  |     | NULL    |                |
| pincode      | varchar(10)   | YES  |     | NULL    |                |
| card_type    | varchar(50)   | YES  |     | NULL    |                |
| card_number  | varchar(20)   | YES  |     | NULL    |                |
| exp_date     | date          | YES  |     | NULL    |                |
| cvv          | varchar(5)    | YES  |     | NULL    |                |
+--------------+---------------+------+-----+---------+----------------+
12 rows in set (0.01 sec)
```

The **payments** table in the database is designed to **store user-submitted payment details** from the payment form on the website. Whenever a user fills out the payment form and submits it, the entered values are **fetched by the host** and **stored in the database**.

Each row in the table represents a **separate payment transaction**, capturing details such as the **user's full name, gender, address, email, pincode, payment amount, card type, card number, expiry date, and CVV**. The **id** column is an **auto-incrementing primary key**, ensuring that each transaction is uniquely identified.

This stored data can later be used for **processing transactions, verifying payments, generating reports, or performing other necessary operations**.

Values stored in payments table:



```
mysql> select * from payments;
+----+--------------+--------+-----------+--------+-----------------------+----------------------+---------+------------+--------------+------------+------+
| id | product_name | amount | full_name | gender | address               | email                | pincode | card_type  | card_number  | exp_date   | cvv  |
+----+--------------+--------+-----------+--------+-----------------------+----------------------+---------+------------+--------------+------------+------+
| 16 | NULL         |  25.00 | Akshay    | Male   | 32 wall street london | akshay@gmail.com     | 110022  | MasterCard | 8978142536   | 2025-04-30 | 1212 |
| 17 | NULL         |  30.00 | Raj       | Male   | Ahmedabad             | raj@gmail.com        | 123456  | MasterCard | 215436529874 | 2025-04-29 | 741  |
| 18 | NULL         |  25.00 | John      | Male   | Wales, England        | john1211@gmail.com   | 789541  | Rupay      | 1245102032   | 2025-04-22 | 582  |
| 19 | NULL         |  30.00 | Kartik    | Male   | 12                    | kartik2004@gmail.com | 632001  | Visa       | 8978142536   | 2025-03-11 | 951  |
| 20 | NULL         |  30.00 | Aksh      | Male   | 32 wall street london | akshaqwy@gmail.com   | 110022  | Visa       | 8978142536   | 2025-04-22 | 1254 |
| 21 | NULL         |  65.00 | Akshita   | Female | 12                    | akshita@gmail.com    | 110022  | Visa       | 8978142502   | 2025-04-23 | 852  |
+----+--------------+--------+-----------+--------+-----------------------+----------------------+---------+------------+--------------+------------+------+
6 rows in set (0.00 sec)
```