Ex. No: 13	FREELANCER EXPENSE TRACKER

AIM:

To create an efficient Freelancer Expense Tracker System and perform CRUD operations using Node.js and MySQL Workbench.

FREELANCER EXPENSE TRACKER SYSTEM:

- It is a web-based application built using **Node.js, MySQL, HTML, CSS, and JavaScript**, designed to help freelancers efficiently track and manage their daily expenses.
- The system offers features such as expense entry, editing and deletion, transaction history viewing, and an update log to monitor changes — all accessible through a clean and user-friendly interface.
- This project aims to demonstrate the integration of backend services, relational database connectivity, and responsive UI design to solve real-world financial tracking challenges faced by freelancers.

CREATING TABLE:

```
CREATE TABLE IF NOT EXISTS expenses (
   id INT AUTO_INCREMENT PRIMARY KEY,
   date DATE NOT NULL,
   category VARCHAR(100) NOT NULL,
   description TEXT,
   amount DECIMAL(10, 2) NOT NULL,
   payment_method VARCHAR(50),
   created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Table Created.

FRONT-END CODING:

index.html (Home page)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Freelancer Expense Tracker</title>
  <!-- Bootstrap CSS -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css"
rel="stylesheet">
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <!-- Title Section with Separate Background -->
  <div class="title-section bg-dark text-white text-center py-5">
    <div class="container" >

      <h1 class="display-4"> <img src="logo.jpg" alt="No Image" width="100" height="100" class="rounded-
circle">    Freelancer Expense Tracker</h1>
      <p class="lead">Track your earnings, expenses, and profits in one place</p>
      <button class="btn btn-primary" onclick="window.location.href='login.html'">Login</button>
      <button class="btn btn-secondary" onclick="window.location.href='signup.html'">Sign Up</button>
    </div>
  </div>

  <!-- Information Section -->
  <div class="container my-5">
    <div class="row justify-content-center">
      <div class="col-md-10 ">
        <div class="card shadow-lg card bg-secondary text-white text-center p-4">
          <img src="data:image/png;base64" class="card-img-top" alt="About Freelancer Expense Tracker"
width="100%" height="250">
          <div class="card-body text-center">
            <h2 class="card-title">About Freelancer Expense Tracker</h2>
            <p class="card-text">Our platform helps freelancers manage their finances efficiently. Track
your income, expenses, and profits with ease. Stay organized and make informed financial decisions with our
user-friendly tracker.</p>
            <p class="card-text">With intuitive tools, you can visualize your financial health, plan budgets,
and ensure financial security. Our secure system protects your data and offers detailed insights to maximize
your earnings.</p>
            <p class="card-text">Join us today and take control of your financial future!</p>
          </div>
        </div>
      </div>
    </div>
  </div>
  <!-- Feature Cards Section -->
  <div class="container my-5">
    <div class="row">
      <div class="col-md-4">
        <div class="card shadow-lg">
```

```
          <img src="data:image/jpeg;base64class="card-img-top" alt="Track Income" width="100%"
height="200">

         <div class="card-body text-center">
            <h5 class="card-title">Track Income</h5>
            <p class="card-text">Easily record and monitor your earnings from multiple sources.</p>
          </div>
        </div>
      </div>
      <div class="col-md-4">
        <div class="card shadow-lg">
          <img src="https://encrypted-
tbn0.gstatic.com/images?q=tbn:ANd9GcS7vlOM7PzbNycYTca8j6nyS3VY-5DK5PiQZQ&s" class="card-img-
top" alt="Manage Expenses" width="100%" height="200">
          <div class="card-body text-center">
            <h5 class="card-title">Manage Expenses</h5>
            <p class="card-text">Keep track of your spending and categorize your expenses efficiently.</p>
          </div>
        </div>
      </div>
      <div class="col-md-4">
        <div class="card shadow-lg">
          <img src="data:image/jpeg;base64
class="card-img-top" alt="Secure Transactions" width="100%" height="200">
        <div class="card-body text-center">
          <h5 class="card-title">Secure Transactions</h5>
          <p class="card-text">Ensure the security of your financial data with our encrypted system.</p>
        </div>
      </div>
    </div>
  </div>
</div>
<div class="container my-5">
  <div class="card bg-dark text-white text-center p-4">
    <h2 class="card-title">User Feedback</h2>
    <div id="feedbackCarousel" class="carousel slide" data-bs-ride="carousel">
      <div class="carousel-inner">
        <div class="carousel-item active">
          <div class="card bg-secondary text-white text-center p-4">
            <h5 class="card-title">Alex</h5>
            <p class="card-text">"This tracker has transformed the way I manage my finances as a
freelancer. The reports and budgeting tools are amazing!"</p>
          </div>
        </div>
        <div class="carousel-item">
          <div class="card bg-secondary text-white text-center p-4">
            <h5 class="card-title">Priya</h5>
            <p class="card-text">"I love how simple and intuitive the interface is. It helps me track my
expenses without any hassle."</p>
          </div>
        </div>
        <div class="carousel-item">
          <div class="card bg-secondary text-white text-center p-4">
```

```
                        <h5 class="card-title">John</h5>
                        <p class="card-text">"Security is a top priority for me, and this platform ensures my financial data is safe."</p>
                    </div>
                </div>

                <div class="carousel-item">
                    <div class="card bg-secondary text-white text-center p-4">
                        <h5 class="card-title">Emily</h5>
                        <p class="card-text">"An excellent tool for freelancers! It helps me track both my income and expenses effortlessly."</p>
                    </div>
                </div>
                <div class="carousel-item">
                    <div class="card bg-secondary text-white text-center p-4">
                        <h5 class="card-title">Michael</h5>
                        <p class="card-text">"I was struggling with budgeting before, but this platform has made it so easy!"</p>
                    </div>
                </div>
                <div class="carousel-item">
                    <div class="card bg-secondary text-white text-center p-4">
                        <h5 class="card-title">Sophia</h5>
                        <p class="card-text">"A must-have for freelancers! Simple, effective, and very secure."</p>
                    </div>
                </div>
            </div>
            <button class="carousel-control-prev" type="button" data-bs-target="#feedbackCarousel" data-bs-slide="prev">
                <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                <span class="visually-hidden">Previous</span>
            </button>
            <button class="carousel-control-next" type="button" data-bs-target="#feedbackCarousel" data-bs-slide="next">
                <span class="carousel-control-next-icon" aria-hidden="true"></span>
                <span class="visually-hidden">Next</span>
            </button>
        </div>
    </div>
</div>
    <!-- Bootstrap JS -->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
</body>
<!-- Footer Section -->
<footer class="bg-dark text-white text-center py-4">
    <div class="container">
        <p>&copy; 2025 Freelancer Expense Tracker. All Rights Reserved.</p>
        <div>
            <p>Contact Us:</p>
            <p>Email: <a href="mailto:boomadevis.23it@kongu.edu" class="text-white">boomadevis.23it@kongu.edu</a></p>
            <p>Phone: <a href="tel:8122764998" class="text-white">8122764998</a></p>
    </div>
    <div>
      <p>Follow Us:</p>
      <a href="https://www.facebook.com" >
        <img
alt="whatsapp" width="30" height="30">
      </a>
    </div></footer></html>
```

Dashboard.html

```
  <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Dashboard</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="bg-light">

  <!-- Navbar -->
  <nav class="navbar navbar-expand-lg navbar-light bg-white shadow-sm">
    <div class="container-fluid">
      <a class="navbar-brand" href="#">Expense Tracker</a>
      <div class="collapse navbar-collapse justify-content-end">
        <ul class="navbar-nav">
          <li class="nav-item">
            <a class="nav-link active" href="dashboard.html">Dashboard</a>
          </li>
          <li class="nav-item">
            <a class="nav-link text-danger" href="#" onclick="logout()">Logout</a>
          </li>
        </ul>
      </div>
    </div>
  </nav>

  <!-- Main Content -->
  <div class="container mt-4">
    <h2>Add Expense</h2>
    <form id="expenseForm" class="mb-4 p-3 bg-white rounded shadow-sm">
      <div class="row g-2">
        <div class="col-md-3">
          <input class="form-control" type="text" id="title" placeholder="Title" required>
        </div>
        <div class="col-md-2">
          <input class="form-control" type="number" id="amount" placeholder="Amount" required>
        </div>
        <div class="col-md-3">
          <input class="form-control" type="text" id="category" placeholder="Category" required>
        </div>
        <div class="col-md-3">
          <input class="form-control" type="date" id="date" required>
        </div>
        <div class="col-md-1">
```

```
     <button class="btn btn-success w-100" type="submit">Add</button>
    </div>
   </div>

</form>

  <h3 class="mb-3">My Expenses</h3>
  <ul id="expenseList" class="list-group mb-5 shadow-sm"></ul>
 </div>
 <!-- Edit Modal -->
 <div class="modal fade" id="editModal" tabindex="-1" aria-hidden="true">
  <div class="modal-dialog">

   <form class="modal-content" id="editForm">
    <div class="modal-header">
     <h5 class="modal-title">Edit Expense</h5>
     <button type="button" class="btn-close" data-bs-dismiss="modal"></button>
    </div>
    <div class="modal-body">
     <input type="hidden" id="editId">
     <div class="mb-3">
      <label>Title</label>
      <input type="text" id="editTitle" class="form-control" required>
     </div>
     <div class="mb-3">
      <label>Amount</label>
      <input type="number" id="editAmount" class="form-control" required>
     </div>
     <div class="mb-3">
      <label>Category</label>
      <input type="text" id="editCategory" class="form-control" required>
     </div>
     <div class="mb-3">
      <label>Date</label>
      <input type="date" id="editDate" class="form-control" required>
     </div>
    </div>
    <div class="modal-footer">
     <button class="btn btn-primary" type="submit">Update</button>
     <button class="btn btn-secondary" data-bs-dismiss="modal" type="button">Cancel</button>
    </div>
   </form>
  </div>
 </div>

 <!-- Logs Table -->
 <div class="container mt-5">
  <h4 class="mb-3">Previous Expense Updates</h4>
  <table class="table table-bordered" id="logsTable">
   <thead class="table-dark">
    <tr>
     <th>Old Title</th>
     <th>Old Amount</th>
     <th>Old Category</th>
     <th>Old Date</th>
     <th>Updated At</th>
     </tr>
    </thead>
    <tbody></tbody>
  </table>
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
<script>
  const userId = localStorage.getItem("userId");

  function logout() {
    if (confirm("Are you sure you want to log out?")) {
      localStorage.removeItem("userId");
      window.location.href = "index.html";


  async function fetchExpenseUpdateLogs() {
    if (!userId) return;

    try {
      const res = await fetch(`http://localhost:3000/api/expense-update-logs/${userId}`);
      const logs = await res.json();
      const tbody = document.querySelector("#logsTable tbody");
      tbody.innerHTML = "";

      if (logs.length === 0) {
        tbody.innerHTML = `<tr><td colspan="5" class="text-center">No logs found</td></tr>`;
        return;
      }

      logs.forEach(log => {
        const tr = document.createElement("tr");
        tr.innerHTML = `
          <td>${log.old_title}</td>
          <td>${log.old_amount}</td>
          <td>${log.old_category}</td>
          <td>${log.old_date}</td>
          <td>${new Date(log.updated_at).toLocaleString()}</td>
        `;
        tbody.appendChild(tr);
      });
    } catch (err) {
      console.error("Failed to fetch update logs:", err);
    }
  }

  document.getElementById("expenseForm").onsubmit = async function (e) {
    e.preventDefault();

    const title = document.getElementById("title").value;
    const amount = document.getElementById("amount").value;
    const category = document.getElementById("category").value;
    const date = document.getElementById("date").value;

    try {
      const response = await fetch("http://localhost:3000/api/expense", {
          method: "POST",
          headers: { "Content-Type": "application/json" },
          body: JSON.stringify({ userId, title, amount, category, date })
        });

        const data = await response.json();
        if (response.status === 201) {
          alert("Expense added successfully!");
          document.getElementById("expenseForm").reset();
          loadExpenses();
        } else {
          alert("Error adding expense: " + data.message);
        }
      } catch (error) {
        console.error("Error adding expense:", error);
        alert("Something went wrong. Please try again.");
      }
    };

    async function loadExpenses() {
      try {
        const res = await fetch(`http://localhost:3000/api/expenses/${userId}`);
        const expenses = await res.json();
        const list = document.getElementById("expenseList");
        list.innerHTML = "";

        if (expenses.length === 0) {
          list.innerHTML = `<li class="list-group-item text-muted">No expenses found.</li>`;
          return;
        }

        expenses.forEach(exp => {
          const formattedDate = new Date(exp.date).toLocaleDateString();
          list.innerHTML += `
            <li class="list-group-item d-flex justify-content-between align-items-center">
              <div>
                <strong>${exp.title}</strong> (${exp.category})<br>
                <small>${formattedDate}</small>
              </div>
              <div>
                <span class="badge bg-primary rounded-pill me-2">₹${parseFloat(exp.amount).toFixed(2)}</span>
                <button class="btn btn-sm btn-outline-secondary me-1"
onclick='openEditModal(${JSON.stringify(exp)})'>Edit</button>
                <button class="btn btn-sm btn-outline-danger" onclick="deleteExpense(${exp.id})">Delete</button>
              </div>
            </li>
          `;
        });
      } catch (error) {
        console.error("Error loading expenses:", error);
        alert("Failed to load expenses.");
      }
    }

    async function deleteExpense(id) {
      if (!confirm("Are you sure you want to delete this expense?")) return;
```

```
  try {
    const res = await fetch(`http://localhost:3000/api/expense/${id}`, {
      method: "DELETE"
    });

    const data = await res.json();
    if (res.ok) {
      alert("Expense deleted successfully.");
      loadExpenses();
    } else {
      alert("Failed to delete: " + data.message);
    }
  } catch (err) {
    console.error("Error deleting:", err);
    alert("Server error.");
  }
}

function openEditModal(expense) {


  document.getElementById("editId").value = expense.id;
  document.getElementById("editTitle").value = expense.title;
  document.getElementById("editAmount").value = expense.amount;
  document.getElementById("editCategory").value = expense.category;
  document.getElementById("editDate").value = expense.date.split("T")[0];
  new bootstrap.Modal(document.getElementById("editModal")).show();
}

document.getElementById("editForm").onsubmit = async function (e) {
  e.preventDefault();

  const id = document.getElementById("editId").value;
  const title = document.getElementById("editTitle").value;
  const amount = document.getElementById("editAmount").value;
  const category = document.getElementById("editCategory").value;
  const date = document.getElementById("editDate").value;

  try {
    const res = await fetch(`http://localhost:3000/api/expense/${id}`, {
      method: "PUT",
      headers: { "Content-Type": "application/json" },
      body: JSON.stringify({ title, amount, category, date })
    });

    const data = await res.json();
    if (res.ok) {
      alert("Expense updated successfully.");
      bootstrap.Modal.getInstance(document.getElementById("editModal")).hide();
      loadExpenses();
      fetchExpenseUpdateLogs();
    } else {
      alert("Update failed: " + data.message);
    }
  } catch (err) {
    console.error("Error updating:", err);
```

```
        alert("Server error.");
      }
    };

    // Load data on page load
    window.onload = () => {
      loadExpenses();
      fetchExpenseUpdateLogs();
    };
  </script>
</body>
</html>
```

Admin.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Admin Dashboard</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="p-4">
  <div class="container">
    <div class="d-flex justify-content-between align-items-center mb-4">
      <h2>Admin Panel - User Expenses</h2>
      <button class="btn btn-danger" onclick="logout()">Logout</button>
    </div>
    <table class="table table-bordered" id="joinTable">
      <thead class="table-dark">
        <tr>
          <th>User Name</th>
          <th>Email</th>
          <th>Title</th>
          <th>Amount</th>
          <th>Category</th>
          <th>Date</th>
        </tr>
      </thead>
      <tbody></tbody>
    </table>
  </div>
  <script>
    async function fetchJoinedData() {
      try {
        const res = await fetch("http://localhost:3000/api/admin/join");
        const data = await res.json();
        const tbody = document.querySelector("#joinTable tbody");
        data.forEach(row => {
          const tr = document.createElement("tr");
          tr.innerHTML = `
            <td>${row.name}</td>
            <td>${row.email}</td>
```

```
 <td>${row.title}</td>
          <td>${row.amount}</td>
          <td>${row.category}</td>
          <td>${row.date}</td> `;
       tbody.appendChild(tr) })}} catch (err) {
      console.error("Error loading data", err);
    }
  }
  function logout() {
    // You can also clear sessionStorage/localStorage here if you use them
    window.location.href = "login.html"; // Redirect to login page
  }

  fetchJoinedData();
</script></body></html>
```

Signup.html

```
          <!DOCTYPE html>
<html>
<head>
  <title>Signup</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body class="bg-light">
  <!-- Navigation Bar -->
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="index.html">Freelancer Expense Tracker</a>
    </div>
  </nav>

  <div class="container mt-5 d-flex justify-content-center">
    <div class="card shadow-sm p-4" style="width: 100%; max-width: 400px;">
      <h3 class="text-center mb-4">Sign Up</h3>
      <form id="signupForm">
        <input class="form-control mb-3" type="text" id="name" placeholder="Full Name" required>
        <input class="form-control mb-3" type="email" id="email" placeholder="Email" required>
        <input class="form-control mb-3" type="password" id="password" placeholder="Password" required>
        <button class="btn btn-primary w-100" type="submit">Create Account</button>
        <div class="text-center mt-3">
          <a href="login.html" class="text-decoration-none">Already have an account? Log in</a>
        </div>
      </form>
    </div>
  </div>

  <script>
    document.getElementById("signupForm").onsubmit = async function (e) {
      e.preventDefault();

      const name = document.getElementById("name").value;
      const email = document.getElementById("email").value;
      const password = document.getElementById("password").value;
      const res = await fetch("http://localhost:3000/api/signup", {
        method: "POST",
        headers: { "Content-Type": "application/json" },
        body: JSON.stringify({ name, email, password })
      });

      if (res.ok) {
        alert("Signup successful");
        window.location.href = "login.html";
      } else {
        alert("Signup failed. Try again.");
      }
    };
  </script>
</body>
</html>
```

Login.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Login</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
  <body class="bg-light">
  <!-- Navigation Bar -->
  <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <div class="container-fluid">
      <a class="navbar-brand" href="index.html">Freelancer Expense Tracker</a>
    </div>
  </nav>
  <div class="container mt-5" style="max-width: 400px;">
    <div class="card shadow-sm">
      <div class="card-body">
        <h3 class="card-title text-center mb-4">Login</h3>
        <form id="loginForm">
          <input class="form-control mb-3" type="email" id="email" placeholder="Email" required>
          <input class="form-control mb-3" type="password" id="password" placeholder="Password" required>
          <button class="btn btn-success w-100" type="submit">Login</button>
          <div class="text-center mt-3">
            <a href="signup.html" class="text-decoration-none">Don't have an account? Sign up</a>
          </div>
        </form>
        <div id="errorMsg" class="text-danger mt-3 text-center" style="display: none;"></div>
      </div>
    </div>
  </div>
  <script>
    document.getElementById("loginForm").onsubmit = async function (e) {
      e.preventDefault();
      const email = document.getElementById("email").value;
      const password = document.getElementById("password").value;
      const errorMsg = document.getElementById("errorMsg");
      try {
        const res = await fetch("http://localhost:3000/api/login", {
          method: "POST",
          headers: { "Content-Type": "application/json" },
          body: JSON.stringify({ email, password })
        });

        const data = await res.json();
  </script>
</body>
</html>
```

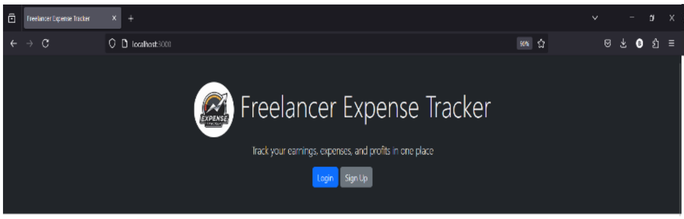
BACKEND CODING:

Server.js

```
const express = require('express');
const mysql = require('mysql2');
const cors = require('cors');
const bodyParser = require('body-parser');
const path = require('path');

const app = express();
app.use(cors());
app.use(bodyParser.json());
app.use(express.static(path.join(__dirname, '../frontend')));

app.get('/', (req, res) => {
  res.sendFile(path.join(__dirname, '../frontend/index.html'));
});

const connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: '19-Apr-06',
  database: 'freelancer_expense',
  multipleStatements: true
});

connection.connect(err => {
  if (err) {
    console.error('Database connection failed:', err);
    return;
  }
  console.log('Connected to MySQL database.');

  // Create user_logs table
  const createUserLogsTable = `
    CREATE TABLE IF NOT EXISTS user_logs (
      log_id INT AUTO_INCREMENT PRIMARY KEY,
      user_id INT,
      name VARCHAR(100),
      email VARCHAR(100),
      created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
    )
  `;
  connection.query(createUserLogsTable, err => {
    if (err) console.error("Error creating user_logs table:", err);
    else console.log("user_logs table ready");
  });

  // Create trigger (workaround without DELIMITER)
  const dropTrigger = `DROP TRIGGER IF EXISTS after_user_signup;`;
  const createTrigger = `
    CREATE TRIGGER after_user_signup
    AFTER INSERT ON users
    FOR EACH ROW
    INSERT INTO user_logs (user_id, name, email)
    VALUES (NEW.id, NEW.name, NEW.email);
  `;

  connection.query(dropTrigger + createTrigger, err => {
    if (err) console.error("Error creating signup trigger:", err.sqlMessage || err);
    else console.log("Signup trigger created");
  });
});

// Signup
app.post('/api/signup', (req, res) => {
  const { name, email, password } = req.body;
  const sql = 'INSERT INTO users (name, email, password) VALUES (?, ?, ?)';
  connection.query(sql, [name, email, password], (err, result) => {
    if (err) return res.status(500).json({ message: 'Signup error' });
    res.status(201).json({ message: 'Signup successful', userId: result.insertId });
  });
});

// Login
app.post('/api/login', (req, res) => {
  const { email, password } = req.body;

  if (email.trim() === 'admin@gmail.com' && password === 'admin123') {
    return res.status(200).json({
      message: 'Admin login successful',
      userId: 0,
      role: 'admin'
    });
  }

  const sql = 'SELECT * FROM users WHERE email = ? AND password = ?';
  connection.query(sql, [email, password], (err, results) => {
    if (err) {
      console.error("Database error:", err);
      return res.status(500).json({ message: 'Internal server error' });
    }

    if (results.length === 0) {
      return res.status(401).json({ message: 'Invalid credentials' });
    }

    const user = results[0];
    res.status(200).json({
      message: 'Login successful',
      userId: user.id,
      role: 'user'
    });
  });
});

// Add Expense
app.post('/api/expense', (req, res) => {
  const { userId, title, amount, category, date } = req.body;
  const sql = 'INSERT INTO expenses (user_id, title, amount, category, date) VALUES (?, ?, ?, ?, ?)';
  connection.query(sql, [userId, title, amount, category, date], (err, result) => {
    if (err) return res.status(500).json({ message: 'Failed to add expense' });
    res.status(201).json({ message: 'Expense added successfully' });
  });
});

// Get Expenses
app.get('/api/expenses/:userId', (req, res) => {
  const userId = req.params.userId;
  const sql = 'SELECT * FROM expenses WHERE user_id = ? ORDER BY date DESC';
  connection.query(sql, [userId], (err, results) => {
    if (err) return res.status(500).json({ message: 'Failed to fetch expenses' });
    res.json(results);
  });
});

// Delete Expense
app.delete('/api/expense/:id', (req, res) => {
  const sql = 'DELETE FROM expenses WHERE id = ?';
  connection.query(sql, [req.params.id], (err, result) => {
    if (err) return res.status(500).json({ message: 'Delete failed' });
    res.json({ message: 'Expense deleted' });
  });
});

// Update Expense
app.put('/api/expense/:id', (req, res) => {
  const { title, amount, category, date } = req.body;
  const sql = 'UPDATE expenses SET title = ?, amount = ?, category = ?, date = ? WHERE id = ?';
  connection.query(sql, [title, amount, category, date, req.params.id], (err, result) => {
    if (err) return res.status(500).json({ message: 'Update failed' });
    res.json({ message: 'Expense updated' });
  });
});

// Show expense update logs for a specific user
app.get("/api/expense-update-logs/:userId", (req, res) => {
  const userId = req.params.userId;
  const sql = "SELECT * FROM expense_update_logs WHERE user_id = ?";
  connection.query(sql, [userId], (err, result) => {
    if (err) {
      console.error("Failed to fetch update logs:", err);
      return res.status(500).json({ message: "Error fetching logs" });

    }
    res.json(result);
  });
});

// Admin join: users + expenses
app.get('/api/admin/join', (req, res) => {
  const sql = `
    SELECT u.name, u.email, e.title, e.amount, e.category, e.date
    FROM users u
    JOIN expenses e ON u.id = e.user_id
    ORDER BY e.date DESC
  `;
  connection.query(sql, (err, results) => {
    if (err) return res.status(500).json({ message: 'Join failed' });
    res.json(results);
  });
});

// Start server
app.listen(3000, () => {
  console.log('Server running on http://localhost:3000');
});
```

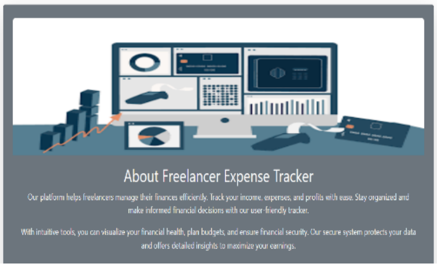
OUTPUT

INDEX PAGE



Freelancer Expense Tracker

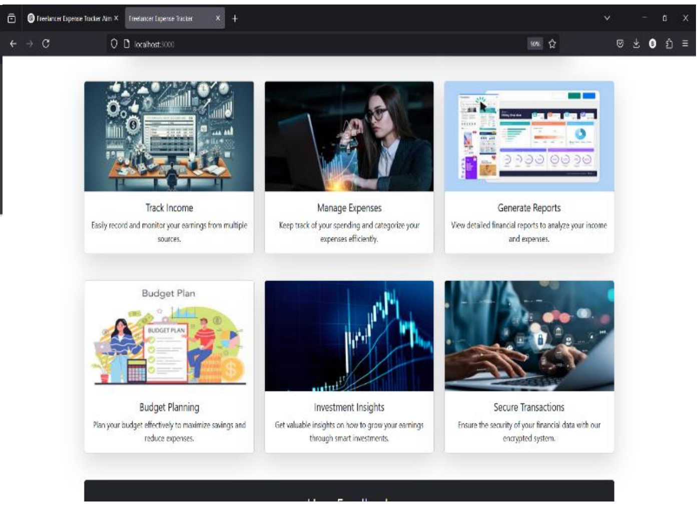
Track your earnings, expenses, and profits in one place



About Freelancer Expense Tracker

Our platform helps freelancers manage their finances efficiently. Track your income, expenses, and profits with ease. Stay organized and make informed financial decisions with our user-friendly tracker.

With intuitive tools, you can visualize your financial health, plan budgets, and ensure financial security. Our secure system protects your data and offers detailed insights to maximize your earnings.



Track Income

Easily record and monitor your earnings from multiple sources.

Manage Expenses

Keep track of your spending and categorize your expenses efficiently.

Generate Reports

View detailed financial reports to analyze your income and expenses.

Budget Planning

Plan your budget effectively to maximize savings and reduce expenses.

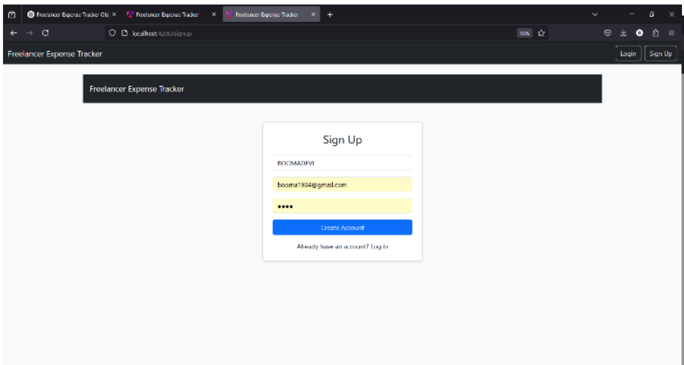
Investment Insights

Get valuable insights on how to grow your earnings through smart investments.

Secure Transactions

Ensure the security of your financial data with our encrypted system.

SIGNUP PAGE


Freelancer Expense Tracker Obj
Freelancer Expense Tracker
Freelancer Expense Tracker
localhost:4200/signup
90%
Freelancer Expense Tracker
Login
Sign Up

Freelancer Expense Tracker

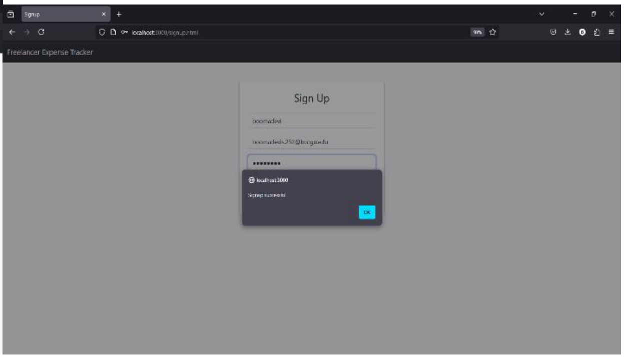
Sign Up

BOOMADEVI

booma1904@gmail.com

••••

Already have an account? Log in



Sign Up

boomadevi

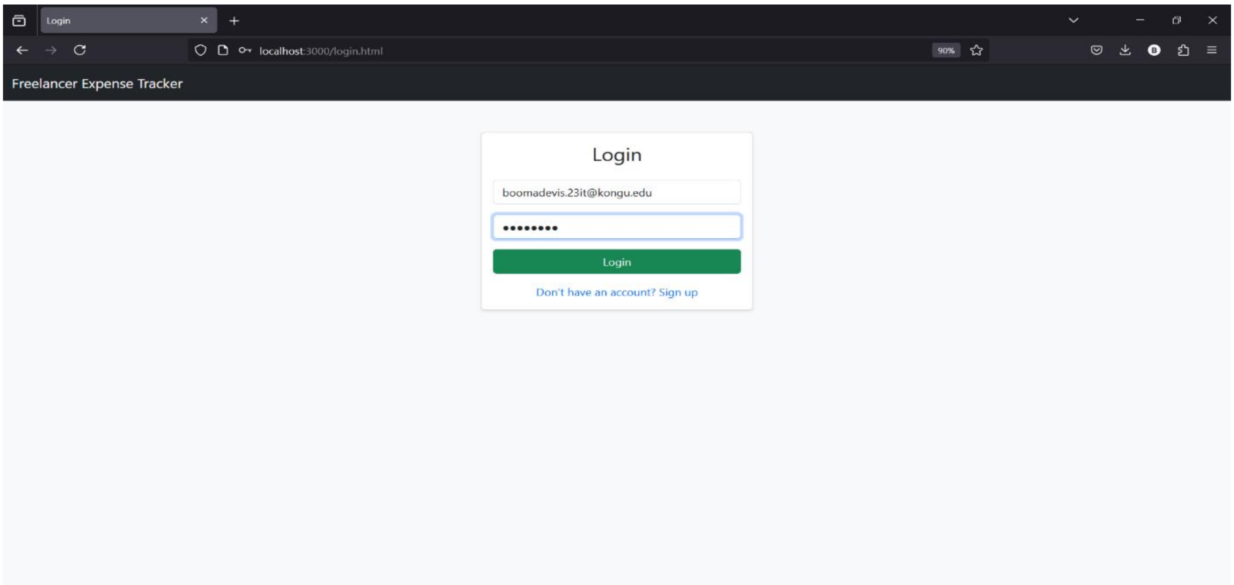
boomadevis.23it@kongu.edu

•••••••••

localhost:3000

Signup successful

LOGIN PAGE



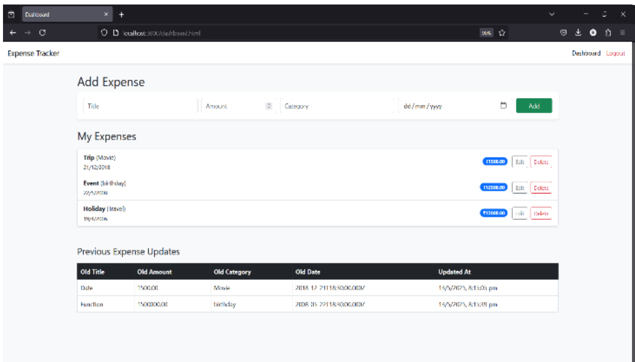
Login

boomadevis.23it@kongu.edu

•••••••••

Don't have an account? Sign up

DASHBOARD



Add Expense

My Expenses

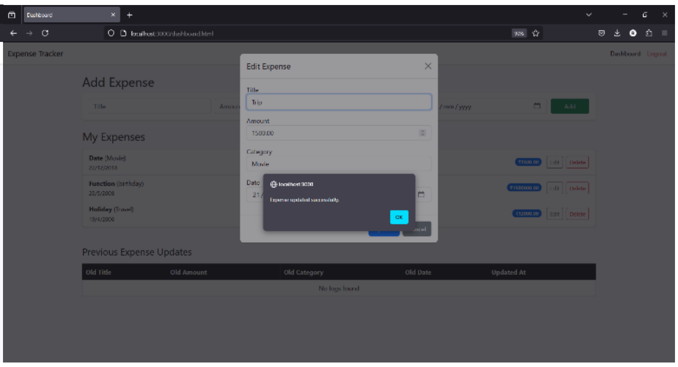
Trip (Movie)
21/12/2018
₹1500.00 Edit Delete

Event (birthday)
22/5/2008
₹12500.00 Edit Delete

Holiday (Travel)
19/4/2006
₹12000.00 Edit Delete

Previous Expense Updates

Old Title	Old Amount	Old Category	Old Date	Updated At
Date	1500.00	Movie	2018-12-21T18:30:00.000Z	16/5/2025, 8:15:06 pm
Function	1500000.00	birthday	2008-05-22T18:30:00.000Z	16/5/2025, 8:15:39 pm



Edit Expense

Title: Trip

Amount: 1500.00

Category: Movie

Date: 21/

localhost:3000

Expense updated successfully.

My Expenses

Date (Movie)
22/12/2018
₹1500.00 Edit Delete

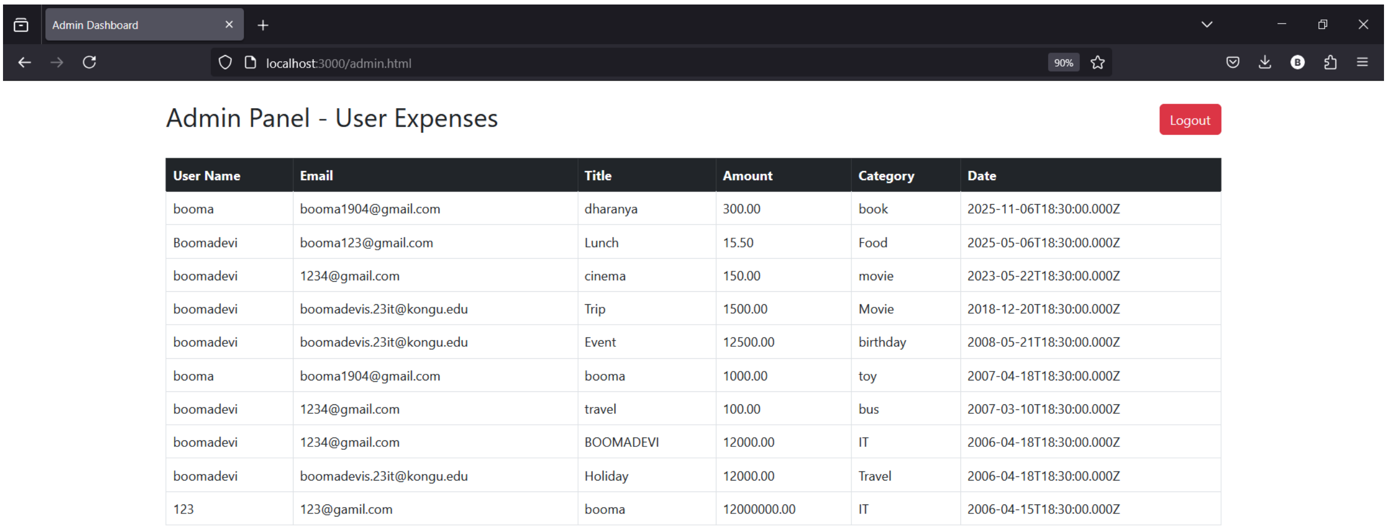
Function (birthday)
23/5/2008
₹1500000.00 Edit Delete

Holiday (Travel)
19/4/2006
₹12000.00 Edit Delete

Previous Expense Updates

Old Title	Old Amount	Old Category	Old Date	Updated At
No logs found				

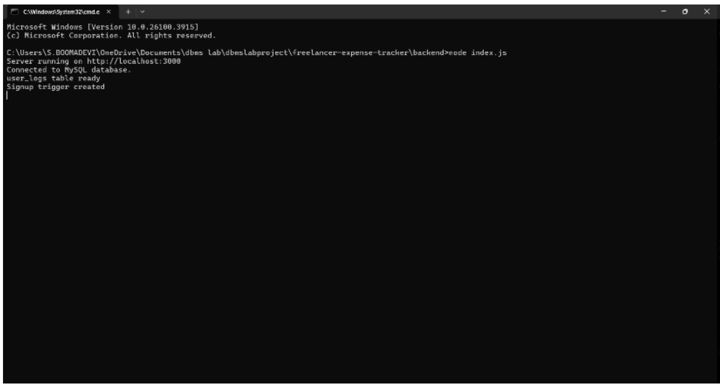
ADMIN PAGE



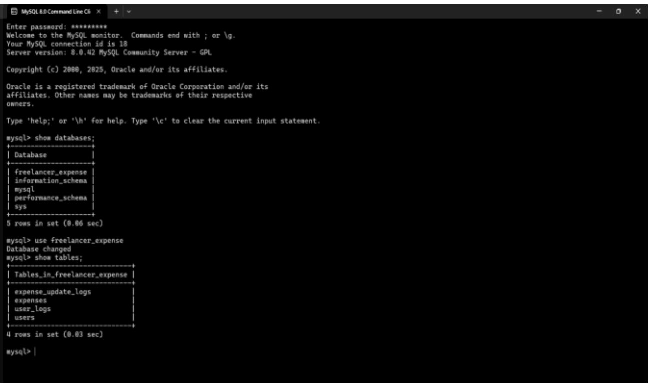
Admin Panel - User Expenses

User Name	Email	Title	Amount	Category	Date
booma	booma1904@gmail.com	dharanya	300.00	book	2025-11-06T18:30:00.000Z
Boomadevi	booma123@gmail.com	Lunch	15.50	Food	2025-05-06T18:30:00.000Z
boomadevi	1234@gmail.com	cinema	150.00	movie	2023-05-22T18:30:00.000Z
boomadevi	boomadevis.23it@kongu.edu	Trip	1500.00	Movie	2018-12-20T18:30:00.000Z
boomadevi	boomadevis.23it@kongu.edu	Event	12500.00	birthday	2008-05-21T18:30:00.000Z
booma	booma1904@gmail.com	booma	1000.00	toy	2007-04-18T18:30:00.000Z
boomadevi	1234@gmail.com	travel	100.00	bus	2007-03-10T18:30:00.000Z
boomadevi	1234@gmail.com	BOOMADEVI	12000.00	IT	2006-04-18T18:30:00.000Z
boomadevi	boomadevis.23it@kongu.edu	Holiday	12000.00	Travel	2006-04-18T18:30:00.000Z
123	123@gamil.com	booma	12000000.00	IT	2006-04-15T18:30:00.000Z

BACKEND

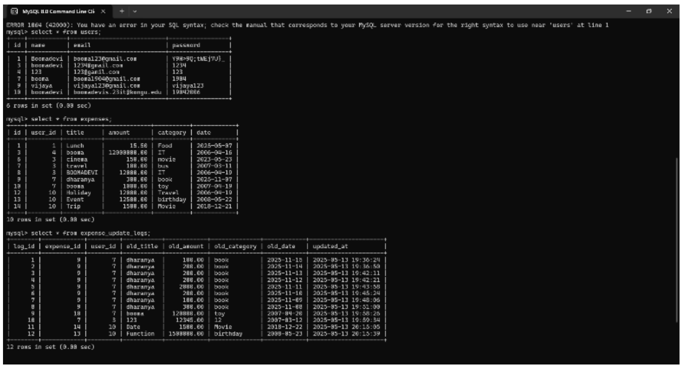


```
Microsoft Windows [Version 10.0.26100.3915]
(c) Microsoft Corporation. All rights reserved.

C:\Users\S.BOOMADEVI\OneDrive\Documents\dbms lab\dbmslabproject\freelancer-expense-tracker\backend>node index.js
Server running on http://localhost:3000
Connected to MySQL database.
user_logs table ready
Signup trigger created
```

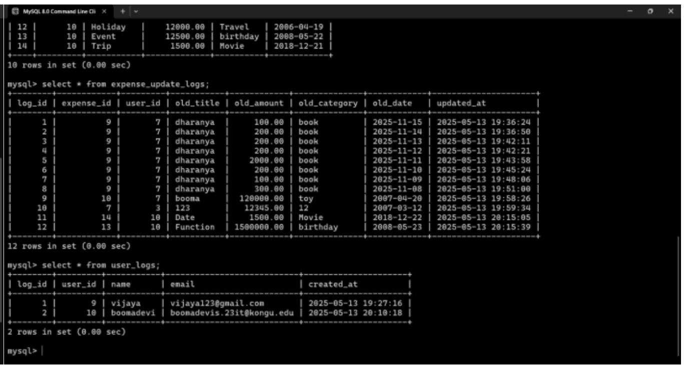


```
Enter password: *********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 8.0.42 MySQL Community Server - GPL

Copyright (c) 2000, 2025, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| freelancer_expense |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.06 sec)

mysql> use freelancer_expense
Database changed
mysql> show tables;
+------------------------------+
| Tables_in_freelancer_expense |
+------------------------------+
| expense_update_logs          |
| expenses                     |
| user_logs                    |
| users                        |
+------------------------------+
4 rows in set (0.03 sec)

mysql>
```

TABLE



```
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'users' at line 1
mysql> select * from users;
+----+-----------+---------------------------+-----------------+
| id | name      | email                     | password        |
+----+-----------+---------------------------+-----------------+
|  1 | Boomadevi | booma123@gmail.com        | Y9H>9Q;tWEj7U}_ |
|  3 | boomadevi | 1234@gmail.com            | 1234            |
|  4 | 123       | 123@gamil.com             | 123             |
|  7 | booma     | booma1904@gmail.com       | 1904            |
|  9 | vijaya    | vijaya123@gmail.com       | vijaya123       |
| 10 | boomadevi | boomadevis.23it@kongu.edu | 19042006        |
+----+-----------+---------------------------+-----------------+
6 rows in set (0.00 sec)

mysql> select * from expenses;
+----+---------+-----------+-------------+----------+------------+
| id | user_id | title     | amount      | category | date       |
+----+---------+-----------+-------------+----------+------------+
|  1 |       1 | Lunch     |       15.50 | Food     | 2025-05-07 |
|  3 |       4 | booma     | 12000000.00 | IT       | 2006-04-16 |
|  6 |       3 | cinema    |      150.00 | movie    | 2023-05-23 |
|  7 |       3 | travel    |      100.00 | bus      | 2007-03-11 |
|  8 |       3 | BOOMADEVI |    12000.00 | IT       | 2006-04-19 |
|  9 |       7 | dharanya  |      300.00 | book     | 2025-11-07 |
| 10 |       7 | booma     |     1000.00 | toy      | 2007-04-19 |
| 12 |      10 | Holiday   |    12000.00 | Travel   | 2006-04-19 |
| 13 |      10 | Event     |    12500.00 | birthday | 2008-05-22 |
| 14 |      10 | Trip      |     1500.00 | Movie    | 2018-12-21 |
+----+---------+-----------+-------------+----------+------------+
10 rows in set (0.00 sec)

mysql> select * from expense_update_logs;
+--------+------------+---------+-----------+------------+--------------+------------+---------------------+
| log_id | expense_id | user_id | old_title | old_amount | old_category | old_date   | updated_at          |
+--------+------------+---------+-----------+------------+--------------+------------+---------------------+
|      1 |          9 |       7 | dharanya  |     100.00 | book         | 2025-11-15 | 2025-05-13 19:36:24 |
|      2 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-14 | 2025-05-13 19:36:50 |
|      3 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-13 | 2025-05-13 19:42:11 |
|      4 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-12 | 2025-05-13 19:42:21 |
|      5 |          9 |       7 | dharanya  |    2000.00 | book         | 2025-11-11 | 2025-05-13 19:43:58 |
|      6 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-10 | 2025-05-13 19:45:24 |
|      7 |          9 |       7 | dharanya  |     100.00 | book         | 2025-11-09 | 2025-05-13 19:48:06 |
|      8 |          9 |       7 | dharanya  |     300.00 | book         | 2025-11-08 | 2025-05-13 19:51:00 |
|      9 |         10 |       7 | booma     |  120000.00 | toy          | 2007-04-20 | 2025-05-13 19:58:26 |
|     10 |          7 |       3 | 123       |   12345.00 | 12           | 2007-03-12 | 2025-05-13 19:59:34 |
|     11 |         14 |      10 | Date      |    1500.00 | Movie        | 2018-12-22 | 2025-05-13 20:15:05 |
|     12 |         13 |      10 | Function  | 1500000.00 | birthday     | 2008-05-23 | 2025-05-13 20:15:39 |
+--------+------------+---------+-----------+------------+--------------+------------+---------------------+
12 rows in set (0.00 sec)
```



```
| 12 |      10 | Holiday   |    12000.00 | Travel   | 2006-04-19 |
| 13 |      10 | Event     |    12500.00 | birthday | 2008-05-22 |
| 14 |      10 | Trip      |     1500.00 | Movie    | 2018-12-21 |
+----+---------+-----------+-------------+----------+------------+
10 rows in set (0.00 sec)

mysql> select * from expense_update_logs;
+--------+------------+---------+-----------+------------+--------------+------------+---------------------+
| log_id | expense_id | user_id | old_title | old_amount | old_category | old_date   | updated_at          |
+--------+------------+---------+-----------+------------+--------------+------------+---------------------+
|      1 |          9 |       7 | dharanya  |     100.00 | book         | 2025-11-15 | 2025-05-13 19:36:24 |
|      2 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-14 | 2025-05-13 19:36:50 |
|      3 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-13 | 2025-05-13 19:42:11 |
|      4 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-12 | 2025-05-13 19:42:21 |
|      5 |          9 |       7 | dharanya  |    2000.00 | book         | 2025-11-11 | 2025-05-13 19:43:58 |
|      6 |          9 |       7 | dharanya  |     200.00 | book         | 2025-11-10 | 2025-05-13 19:45:24 |
|      7 |          9 |       7 | dharanya  |     100.00 | book         | 2025-11-09 | 2025-05-13 19:48:06 |
|      8 |          9 |       7 | dharanya  |     300.00 | book         | 2025-11-08 | 2025-05-13 19:51:00 |
|      9 |         10 |       7 | booma     |  120000.00 | toy          | 2007-04-20 | 2025-05-13 19:58:26 |
|     10 |          7 |       3 | 123       |   12345.00 | 12           | 2007-03-12 | 2025-05-13 19:59:34 |
|     11 |         14 |      10 | Date      |    1500.00 | Movie        | 2018-12-22 | 2025-05-13 20:15:05 |
|     12 |         13 |      10 | Function  | 1500000.00 | birthday     | 2008-05-23 | 2025-05-13 20:15:39 |
+--------+------------+---------+-----------+------------+--------------+------------+---------------------+
12 rows in set (0.00 sec)

mysql> select * from user_logs;
+--------+---------+-----------+---------------------------+---------------------+
| log_id | user_id | name      | email                     | created_at          |
+--------+---------+-----------+---------------------------+---------------------+
|      1 |       9 | vijaya    | vijaya123@gmail.com       | 2025-05-13 19:27:16 |
|      2 |      10 | boomadevi | boomadevis.23it@kongu.edu | 2025-05-13 20:10:18 |
+--------+---------+-----------+---------------------------+---------------------+
2 rows in set (0.00 sec)

mysql>
```

CONTENTS	MARKS ALLOTED	MARKS OBTAINED
Aim, Algorithm, SQL, PL/SQL	30	
Execution and Result	20	
Viva	10	
Total	60	

RESULT:

The Freelancer Expense Tracker System was created using Node.js, MySQL, HTML, CSS, and JavaScript.Users can add, view, edit, and delete their expense records easily.