MERN Stack Employees Flow

Employees Flow (Frontend -> Backend -> Database -> Frontend)

1. Frontend Navigation
- frontend/src/App.jsx
  - React Router loads <Employees /> page when user navigates to /employees.

2. Frontend Page
- frontend/src/pages/Employees.jsx
  - Calls employeeAPI.getAll() (Axios) to fetch employees.

3. API Layer
- frontend/src/api/api.js
  - employeeAPI.getAll() -> Axios request to GET http://localhost:5000/api/employees

4. Backend Entry Point
- backend/server.js
  - Route mounted: app.use("/api/employees", require("./routes/core/employees"));

5. Backend Route
- backend/routes/core/employees.js
  - Handles GET / -> calls MongoDB via Employee model.
  - Returns JSON: { success: true, count: X, data: [...] }

6. Database Layer
- backend/models/Employee.js
  - Mongoose schema for Employee collection.
  - .find() retrieves all employee documents.

7. Response Back to Frontend
- Express sends JSON -> Axios receives it -> React updates state.

8. Frontend Render
- frontend/src/pages/Employees.jsx
  - Maps over employees and displays them in a list/table.

Diagram (with file names)

```
[User opens /employees in browser]
        |
        v
frontend/src/App.jsx  (Router -> <Employees />)
```

```
       |
       v
frontend/src/pages/Employees.jsx
  (calls employeeAPI.getAll())
       |
       v
frontend/src/api/api.js
  (Axios -> GET /api/employees)
       |
       v
backend/server.js
  (app.use("/api/employees", ./routes/core/employees))
       |
       v
backend/routes/core/employees.js
  (router.get("/") -> Employee.find())
       |
       v
backend/models/Employee.js (Mongoose -> MongoDB)
       |
       v
MongoDB (employees collection)
       |
       v
backend/routes/core/employees.js
  (responds with { success, count, data })
       |
       v
frontend/src/pages/Employees.jsx
  (state updated, UI renders employee list)
```