**Nike Store – E-Commerce Project**

**Name:** Tanishka Chandrakant Patil

**Class :** TE Div B

**Roll No:** T52019

**Tools / Software / Platforms Used**

**Frontend**

* HTML5 & CSS3
* JavaScript (Vanilla JS)

**Backend**

* Node.js
* Express.js

**Database**

 **MongoDB (Local) with MongoDB Compass** – Used for storing product details, user orders, and customer data.

 MongoDB Compass provides a GUI for managing and visualizing the database.

**Installation & Setup**

**1. Clone Repository**

git clone <your-repo-link>

cd nike-store

**2. Install Backend Packages**

Run the following commands inside the root folder (backend directory):

npm init -y

npm install express mongoose cors body-parser nodemon

**To Run Backend:**

npm run dev

(Default server: http://localhost:5000/)

**3. Setup MongoDB Database**

**a) Install MongoDB**

* Download and install **MongoDB Community Edition** from the [official website](https://www.mongodb.com/try/download/community).
* During installation, select **MongoDB Compass** (GUI).

**b) Start MongoDB Server**

* Usually, it runs automatically as a service.
* If not, run this command:

mongod

(Default port: **27017**)

**c) Connect via Compass**

* Open **MongoDB Compass**.
* Connection string for local database:

mongodb://127.0.0.1:27017

* Click **Connect**.

**d) Create Database**

* Database Name: nikeStore
* Collections:
  + orders
  + users

**e) Update Connection in Backend**

In your app.js (backend):

const mongoose = require("mongoose");

mongoose.connect("mongodb://127.0.0.1:27017/nikeStore", {

useNewUrlParser: true,

useUnifiedTopology: true

})

.then(() => console.log("MongoDB Connected"))

.catch((err) => console.log(err));

**4. Run Frontend**

* All frontend files are inside the /public folder.
* Open login.html directly in your browser or use VS Code Live Server Extension.

**🚀 Features**

1. Product Catalog (Nike Shoes Collection)
2. Select Size & Color
3. Checkout and Place Order
4. Order stored in MongoDB Database
5. Order Success Page showing details
6. Responsive UI with slider

**Project Workflow**

1. User selects a product → Chooses **size & color** → Proceeds to checkout.
2. User enters details → Order stored in **MongoDB Atlas**.
3. On success → Redirected to confirmation page showing **order details**.

**Scripts :** npm run dev # Run backend with nodemon