**Steps to Host MongoDB on Atlas**

**Step 1: Create an Account on MongoDB Atlas**

1. Go to [MongoDB Atlas](https://www.mongodb.com/atlas/database).
2. Click on "Sign Up" and create an account using your email or GitHub/Google account.
3. Verify your email and log in to your MongoDB Atlas account.

**Step 2: Create a New Cluster**

1. After logging in, click on "Create a New Project" and give it a name.
2. Inside the project, click "Build a Cluster."
3. Choose a free shared cluster (M0) or select a paid option.
4. Select a cloud provider (AWS, Azure, or GCP) and region.
5. Click "Create Cluster" and wait for the cluster to be provisioned.

**Step 3: Configure Security Settings**

1. Navigate to "Database Access" in the left sidebar.
2. Click "Add New Database User."
3. Set authentication method (Username & Password).
4. Grant access roles (e.g., Read and Write permissions).
5. Click "Add User."
6. Navigate to "Network Access" and click "Add IP Address."
7. Choose "Allow Access from Anywhere" (0.0.0.0/0) or specify IP addresses.
8. Click "Confirm."

**Step 4: Connect to the Database**

1. Go to the "Clusters" tab.
2. Click on "Connect" next to your cluster.
3. Select "Connect Your Application" or "Connect with MongoDB Compass."
4. Copy the connection string (replace <username> and <password> with actual credentials).

**Step 5: View Database in MongoDB Atlas**

1. Click on "Browse Collections" in your cluster.
2. You can create new databases and collections or insert documents here.

**Step 6: Connect Using MongoDB Compass if not**

1. Download and install [MongoDB Compass](https://www.mongodb.com/products/compass).
2. Open Compass and click "New Connection."
3. Paste the connection string from Atlas.
4. Click "Connect" to access your database.

**Step 7: Verify and Manage Data**

1. After connecting, explore your collections in Compass.
2. Run queries, insert documents, and manage indexes.

Your MongoDB is now successfully hosted on Atlas and accessible via Compass or any application using the provided connection string.

Hosted our Database on MongoDB Atlas and URL is -

**mongodb+srv://rajvardhant563:LqiFS8i6SUuT4gWK@cluster0.bv5cc3v.mongodb.net/JobPortalDB**

//we can run this string in our local MongoDB(MongoDB Compass) and can see our hosted database locally

**Steps to Deploy a Node.js Application on Render**

**Step 1: Create an Account on Render**

1. Go to [Render](https://render.com/).
2. Click on "Sign Up" and create an account using GitHub, GitLab, or email.
3. Verify your email and log in to your Render account.

**Step 2: Link Your Repository**

1. Click on "New" and select "Web Service."
2. Connect your GitHub/GitLab account to Render.
3. Choose the repository containing your Node.js application.

**Step 3: Configure the Deployment**

1. Select the branch to deploy (e.g., main or master).
2. Set the runtime as Node.js.
3. Provide a name for your service.
4. Specify the build command, typically: npm install
5. Specify the start command, such as: node server.js

**Step 4: Set Environment Variables**

1. Navigate to the "Environment" section.
2. Add required variables like:
   * PORT: Set to 5000 or as per your app.
   * Database credentials (if required).
3. Click "Save Changes."

**Step 5: Deploy the Application**

1. Click "Create Web Service."
2. Render will automatically build and deploy your application.
3. Wait for the deployment to complete.

**Step 6: Verify and Test**

1. Once deployed, Render provides a live URL.
2. Visit the URL to check if the application is running correctly.
3. Monitor logs in the "Logs" section for debugging.

**Step 7: Enable Automatic Deployments**

1. In the settings, enable "Automatic Deploys."
2. Render will redeploy the app whenever you push new changes to the repository.

Your Node.js application is now successfully deployed on Render and accessible via the provided live URL.

Hosted our Backend on Render and URL is –

**https://job-portal-backend-f8nm.onrender.com**

//We can use this command in our frontend and can make requests to backend

//for example –

For creating seeker’s profile -

<https://job-portal-backend-f8nm.onrender.com/api/profile/seeker> **(POST)**