

problem-8

Problem

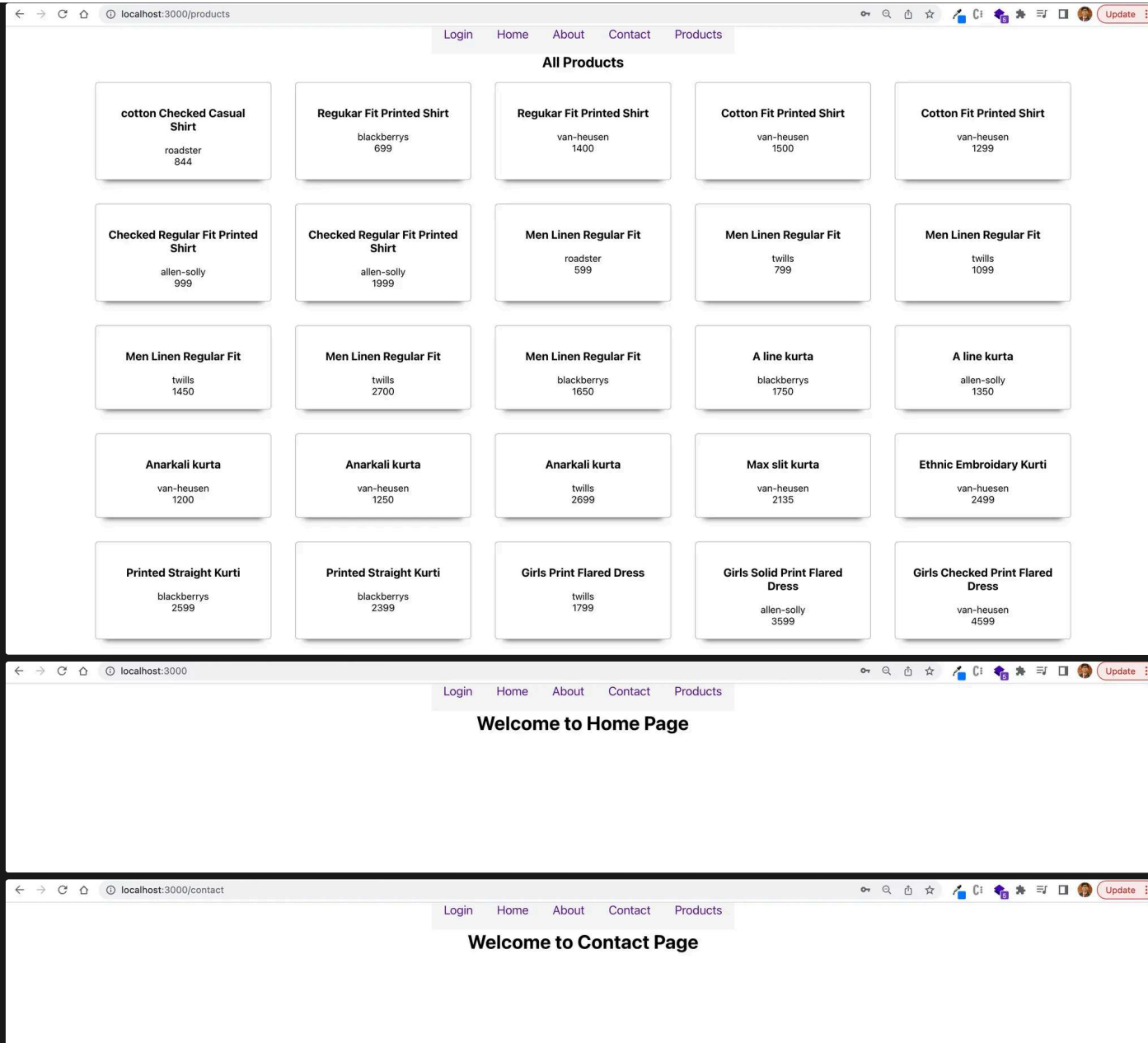
Understanding Component Structure

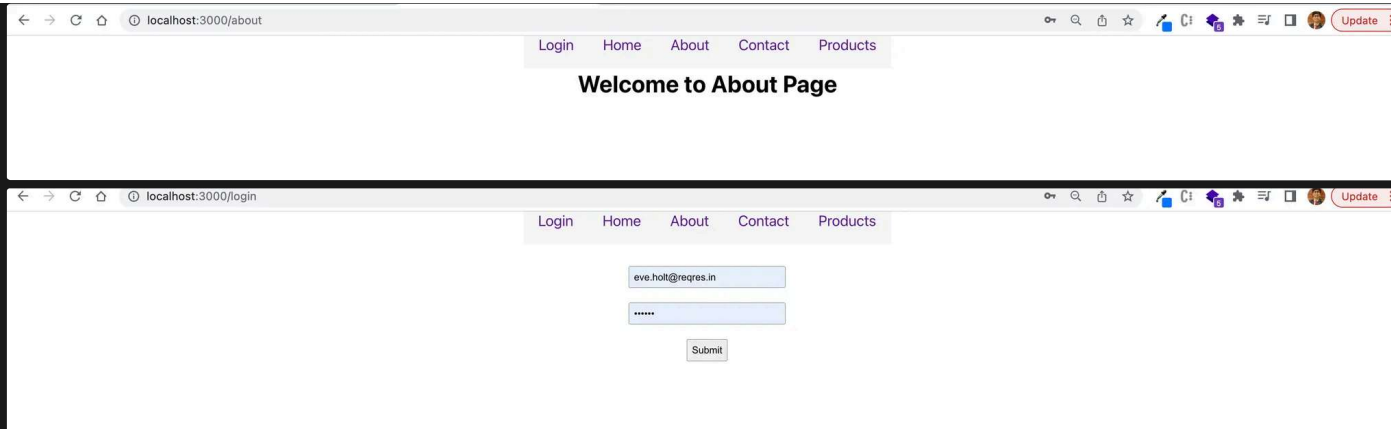
- App
 - Login (should be rendered on route `"/login"`)
 - Home (should be rendered on default route `"/"`)
 - About (should be rendered on route `"/about"`)
 - Contact (should be rendered on route `"/contact"`)
 - Products (should be rendered on route `"/products"`)
- Context
 - AuthContextProvider
- Routes
 - AllRoutes
 - PrivateRoute

Note - Make sure you use only the given components and dont create new files and folders. Changing component name, structures might result in giving you zero marks.

Features to build

1. Create a navbar which has the following links (strictly should have the same order as below)
 - Login
 - Home
 - About
 - Contact
 - Products
2. Authentication will be done using this API : <https://reqres.in/api/login> (POST request).
 - Use fetch to make the API request.
 - Read the API docs to find the credentials for login: <https://reqres.in/>
1. Clicking on the navbar link should redirect the user to that component and update url at the same time
2. Navbar component should be visible on every page
3. Products page should only be accessible only after login
4. Use this api to fetch the products data: <https://dbioz2ek0e.execute-api.ap-south-1.amazonaws.com/mockapi/get-products>
5. redirect the user to login page if the user tries to access product page without logging in
6. User should always be redirected to home when the login form is successfully submitted
7. Login page should be accessible after logging in, rather redirect the user to "/" page (Use Private Route component to achieve this)
8. Use context api to maintain state in the application:
 - Your initial state should have `isAuth` as false





General Instructions (*IMPORTANT*)

1. Do not use Global CSS, instead use `<componentName>.module.css` convention for Css in that file.