API Integration and React Routing- Practice Exercises

By now, we assume that you have a replica of the Apple website you created with React. All of the questions below will be completed using that replica as your starter code. For the second question, we assume that you have created a database for Apple products with all the 3 tables (Products, ProductsPrice and ProductDescription) in your MySQL class.

Notes:

- Whether your Apple replica components are class or function based, there will be no problem to call API or apply React routing

Question on React API

1. Apple has a YouTube channel with various videos available. Please display 9 random YouTube videos on your React based Apple website replica by fetching these videos from the YouTube data API that is publicly available. Please display your YouTube video component right below the component that has the last two images, i.e., right below the Arcade and Apple Card section. You will need to get an API key to access YouTube data API.

Question on React Routing

2. Create a dynamic page to display the different Apple products that you stored in your Database (during your MySQL class). To do that, you will need to create a functional component called "**Iphone.js**" and fetch the Iphone products data from your Database using API. Once you fetch the data, display them on your React based Apple website replica when the "iphone" section from your Navbar is clicked. Please watch the demo video here to see what your dynamic page should look like.
3. Create a dynamic page to display a single Apple product (that you stored in your Database) when the "**Learn more**" button from your "**Iphone.js**" page is clicked. Please watch the demo video here to see what your dynamic page should look like

Steps to Follow for Question 1

- In your “**Evangadi**” folder, create a folder called “**ReactAPIandRouting**”
- Copy your latest React based working Apple replica project folder into the “**ReactAPIandRouting**” folder
- Create a functional component called “**YoutubeVideos.js**” and add it to the folder where you have your components
    - Limit the number of YouTube videos to fetch to 8
    - You are responsible for styling your newly created “**YoutubeVideos.js**” component with CSS. For this, you will need to create a css file called “**YoutubeVideos..css**” and remember to import it in your“**YoutubeVideos.js**” component . For better understanding, look at the following designs by scrolling down to the YouTube video section:
        - For small screen sizes, the videos styling should look like this
        - For medium and larger screen sizes, the videos styling should look like this
- You will need to initiate a state to hold your YouTube videos and an updater function
- You will need to import useState and useEffect from React into your “**YoutubeVideos.js**” component
- Before proceeding with Mapping on each video and returning the videos, make sure you have fetched all the 8 videos. To do that, just log your current state into console and see if an array of 8 videos is printed in your console
- By mapping on the array containing the fetched YouTube videos, please show the “**Thumbnail**”, “**Video Title**”, “**Video description**” and “**Publishing date**”
- Steps to creating a YouTube API key. **Note**: The steps to creating a YouTube API key change from time to time as Google changes its website features every now and then. Therefore, you might need to research online for current information. However, here is a general step on how to create a YouTube API key:
    - If you do not have a Gmail account, create one and stay signed in
    - Click here, the Google Developer Console link and create a project called “**Apple API Project**”. Wait until Google finishes creating your project

  - Select the "**Apple API Project**" project you created, you will see all the information about the project you created
  - Now, find "**API & SErvices**" and click on it. Then, click on "**Library**"
  - You will see Google's API library page with a search bar to look for the various API services provided by Google. Just type "YouTube" in the search bar
  - Click on "**YouTube Data API v3**" and click on the "**Enable**" button. Wait until Google finishes enabling it
  - You are now taken to the "**YouTube Data API v3**" page. Find "**credentials**"
  - Find "**Create credentials**" and click on it
  - Select the "**YouTube Data API v3**" option
  - Select "**public data**" to answer the question about what data you will need to access
  - Click "**Next**". Excellent!!! You will now see your API key created
  - If you copy and paste your API key at the end of the link provided below, you will see the YouTube videos's data in a JSON form
    - https://www.googleapis.com/youtube/v3/search?key=

Steps to follow for question 2

- Create a functional component called "**Iphone.js**" where you will fetch your Apple products that you stored in your Database
- Make sure your Database is running
- You are creating a React based Apple website that has multiple routes. Therefore, make sure you create a new route for your "**Iphone**" page as well
- You are responsible for styling your "**Iphone**" component

Steps to Follow for Question 3

- Create a functional component called "**SingleAppleProduct.js**" where you will fetch your Apple products that you stored in your Database
- You will need to specify the unique identifier of each product you have in your database. You used "pid" as a unique ID when you created your tables in your database.
  - You will pass this identification as a parameter when you provide the route on your App.js. Meaning, when iphone/:id is entered as a URL, your

“**SingleAppleProduct.js**” will load to show the single product you fetched from your Database using API

- Make sure your Database is running
- You are responsible for styling your “**SingleAppleProduct.js**” component

Happy coding 😀