

Functional Components and Development

Dr Harshad Prajapati
1 Dec 2023

1

React Functional Component

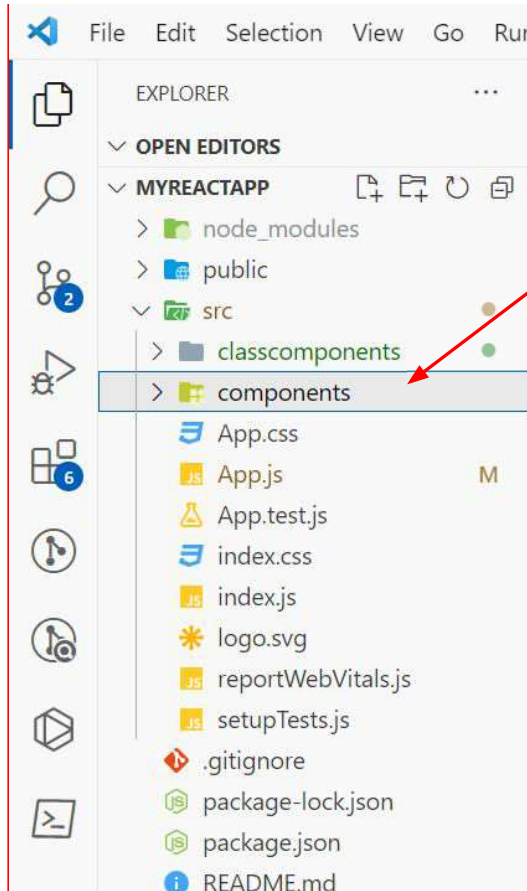
- **Component** is one of the **core concepts** of **React**.
- We build user interfaces (**UI**) using component.
- React component is a **JavaScript function** that we can sprinkle with **markup**.
 - React **recommends** use of **functional components**, though it started with class components.
- Functional **component** (React component) **returns markup**.

2

Use Existing App: **myreactapp**

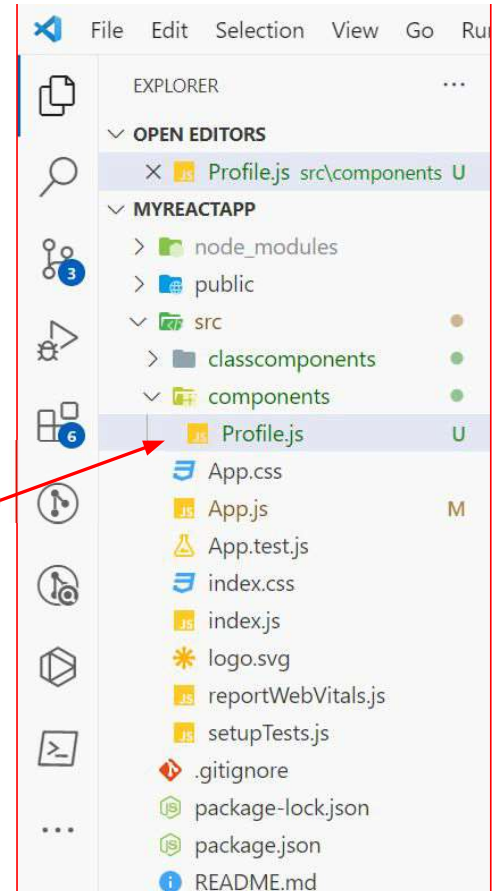
GitHub Repo: myreactapp/fun-components

3



Add new folder named **components** under **src**

Add new file named **Profile.js** under **components**



4

Profile.js 2, U ● Profile.js

src > components > Profile.js

1 rafce

rafce reactArrowFunctionExportComponent
ReadableStreamDefaultController

Creates a React Arrow Function Component with ES7 module system (ES7+ React/Redux/React-Native snippets)

```
import React from 'react'

const = () => {
  return (
    <div></div>
  )
}

export default
```

We use code snippet rafce.

Template code of snippet rafce.

5

Auto Generated Code of Profile.js

Profile.js U ●

src > components > Profile.js > ...

```
1 import React from 'react'
2
3 const Profile = () => {
4   return (
5     <div>Profile</div>
6   )
7 }
8
9 export default Profile
```

The code snippet generates this code.

We export component so that we can use it in another file (module).

6

Return Markup from Profile Component



The screenshot shows a code editor with a file named `Profile.js`. The breadcrumb path is `src > components > Profile.js > ...`. The code defines a `Profile` component that returns a JSX element. A red box highlights the path, and another red box highlights the return markup. Red arrows point from text annotations to these elements.

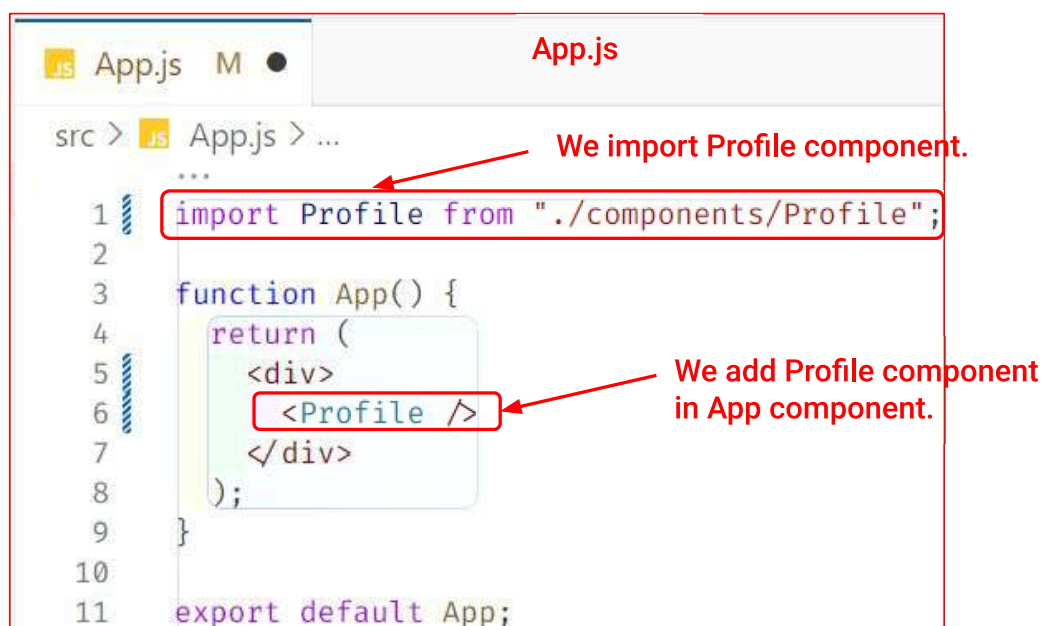
```
1 import React from 'react'
2
3 const Profile = () => {
4   return (
5     <div>
6       
7       <div>Katherine Johnson</div>
8     </div>
9   );
10 }
11
12 export default Profile
```

Annotations:

- Path from the src folder
- We return this markup

7

Add Profile Component in App Component



The screenshot shows a code editor with a file named `App.js`. The breadcrumb path is `src > App.js > ...`. The code defines an `App` component that imports the `Profile` component and renders it. Red boxes highlight the import statement and the `<Profile />` tag. Red arrows point from text annotations to these elements.

```
1 import Profile from "../components/Profile";
2
3 function App() {
4   return (
5     <div>
6       <Profile />
7     </div>
8   );
9 }
10
11 export default App;
```

Annotations:

- We import Profile component.
- We add Profile component in App component.

8

The index.js uses App Component

index.js

index.js

This code is generated by create-react-app

```
src > index.js > ...
  You, now | 1 author (You)
1  import React from 'react';
2  import ReactDOM from 'react-dom/client';
3  import './index.css';
4  import App from './App';
5
6  const root = ReactDOM.createRoot(document.getElementById('root'));
7  root.render(
8    <React.StrictMode>
9    <App />
10 </React.StrictMode>
11 );
```

The <div> with id=root is present in index.html file

We add App component in div with id=root.

9

index.html M

index.html

This code is generated by create-react-app

```
public > index.html > ...
  You, 1 second ago | 1 author (You)
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="utf-8" />
5      <link rel="icon" href="%PUBLIC_URL%/favicon.ico" />
6      <meta name="viewport" content="width=device-width, initial-scale=1" />
7      <meta name="theme-color" content="#000000" />
8      <meta
9        name="description"
10       content="Web site created using create-react-app"
11     />
12     <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
13     <!--
14       manifest.json provides metadata used when your web app is installed on a
15       user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
16     -->
17     <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
```

10

index.html

This code is generated by create-react-app

```
18 <!--
19 Notice the use of %PUBLIC_URL% in the tags above.
20 It will be replaced with the URL of the `public` folder.
21 Only files inside the `public` folder can be referenced from the HTML.
22
23 Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
24 work correctly both with client-side routing and a non-root public URL.
25 Learn how to configure a non-root public URL by running `npm run build`.
26 -->
27 <title>React App</title>
28 </head>
29 <body>
30   <noscript>You need to enable JavaScript to run this app.</noscript>
31   <div id="root"></div>
32   <!--
33     This HTML file is a template.
34     If you open it directly in the browser, you will see an empty page.
35
36     You can add webfonts, meta tags, or analytics to this file.
37     The build step will place the bundled scripts into the <body> tag.
38
39     To begin the development, run `npm start` or `yarn start`.
40     To create a production bundle, use `npm run build` or `yarn build`.
41   -->
42 </body>
43 </html>
```

In this <div> (with id=root), the App component and all other its child components are mounted.

11

Run App using npm start

```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS COMMENTS

Compiled successfully!

You can now view myreactapp in the browser.

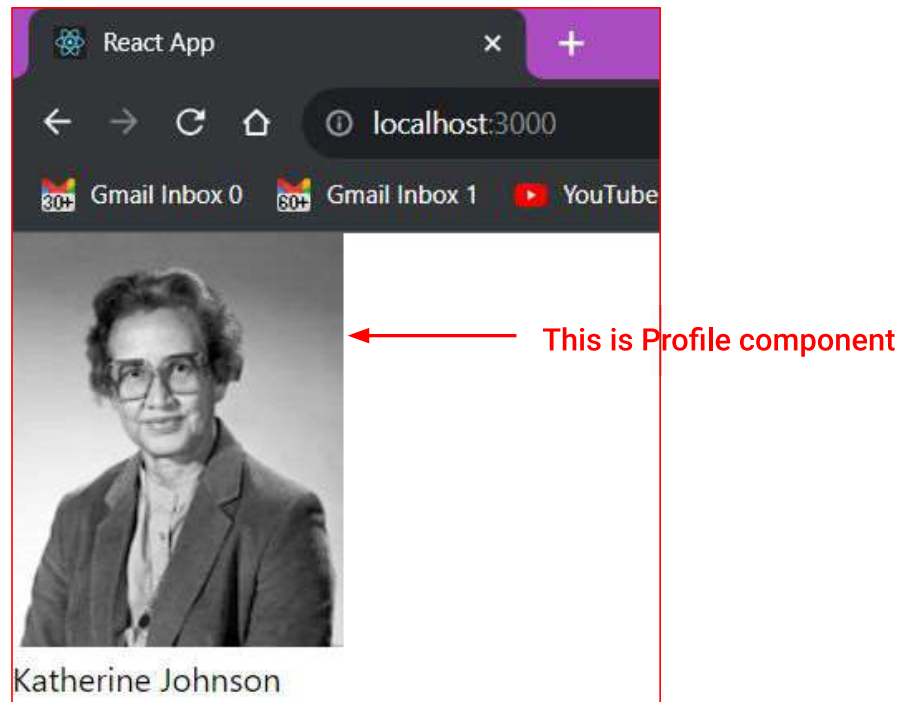
Local:      http://localhost:3000
On Your Network: http://192.168.29.194:3000

Note that the development build is not optimized.
To create a production build, use npm run build.

webpack compiled successfully
```

12

View App in Browser



13

View Page Source

View Page source in Chrome Browser

We do not see any Profile component in Page Source

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="utf-8" />
5     <link rel="icon" href="/favicon.ico" />
6     <meta name="viewport" content="width=device-width, initial-scale=1" />
7     <meta name="theme-color" content="#000000" />
8     <meta
9       name="description"
10      content="Web site created using create-react-app"
11    />
12    <link rel="apple-touch-icon" href="/logo192.png" />
13    <!--
14      manifest.json provides metadata used when your web app is installed on a
15      user's mobile device or desktop. See https://developers.google.com/web/f
16    -->
17    <link rel="manifest" href="/manifest.json" />
18    <!--
19      Notice the use of ` ` in the tags above.
20      It will be replaced with the URL of the `public` folder during the build.
21      Only files inside the `public` folder can be referenced from the HTML.
22
23      Unlike `/favicon.ico` or `favicon.ico`, `/favicon.ico` will
24      work correctly both with client-side routing and a non-root public URL.
25      Learn how to configure a non-root public URL by running `npm run build`.
26    -->
27    <title>React App</title>
28    <script defer src="/static/js/bundle.js"></script></head>
29    <body>
30      <noscript>You need to enable JavaScript to run this app.</noscript>
31      <div id="root"></div>
32    <!--
33      This HTML file is a template.
34      If you open it directly in the browser, you will see an empty page.
35
36      You can add webfonts, meta tags, or analytics to this file.
37      The build step will place the bundled scripts into the <body> tag.
38
39      To begin the development, run `npm start` or `yarn start`.
40      To create a production bundle, use `npm run build` or `yarn build`.
41    -->
42    </body>
43  </html>
```

14

View Page Source

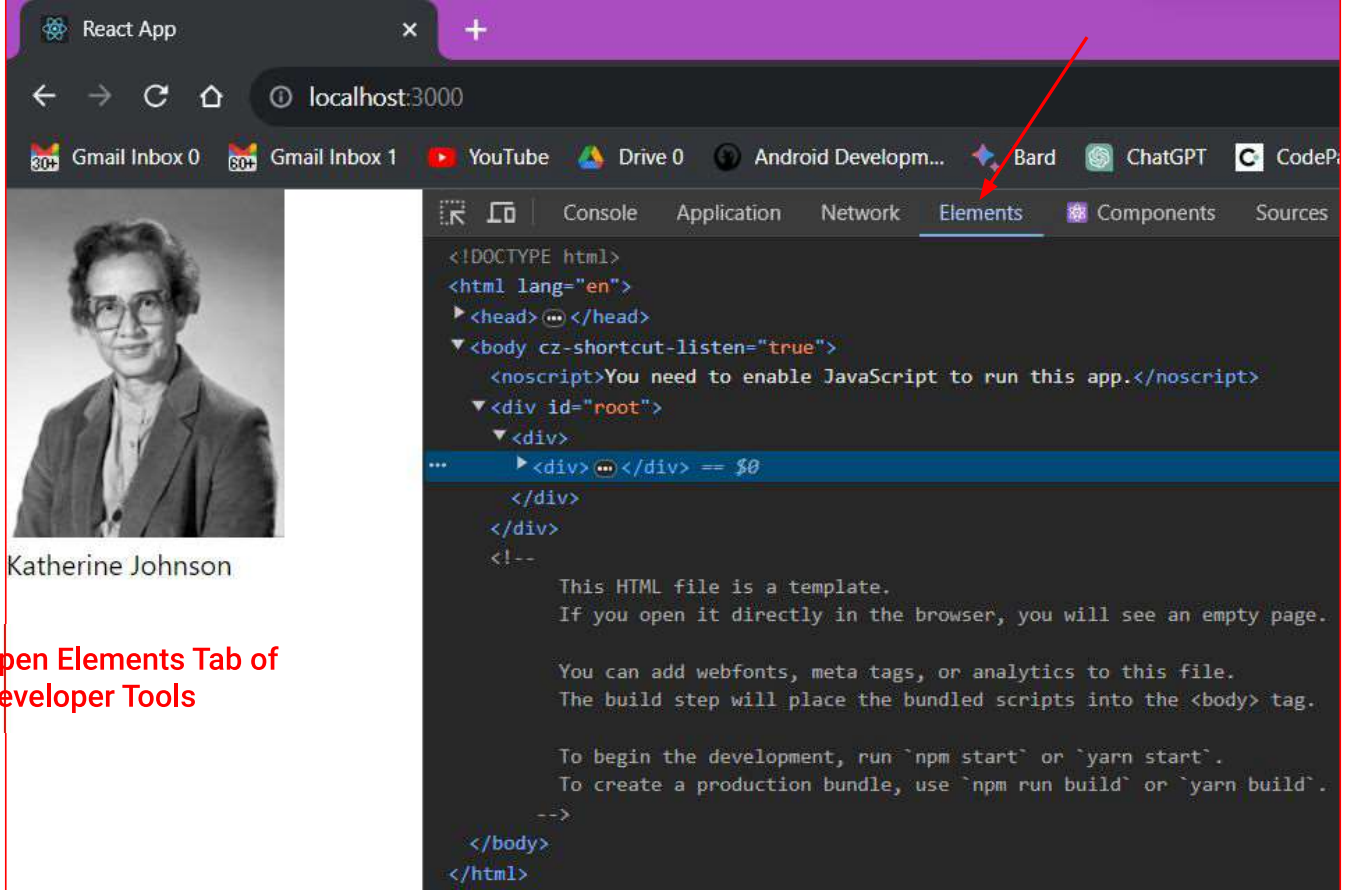
```
27 <title>React App</title>
28 <script defer src="/static/js/bundle.js"></script></head>
29 <body>
30   <noscript>You need to enable JavaScript to run this app.</noscript>
31   <div id="root"></div>
```

We do not see any Profile component in <body>

15

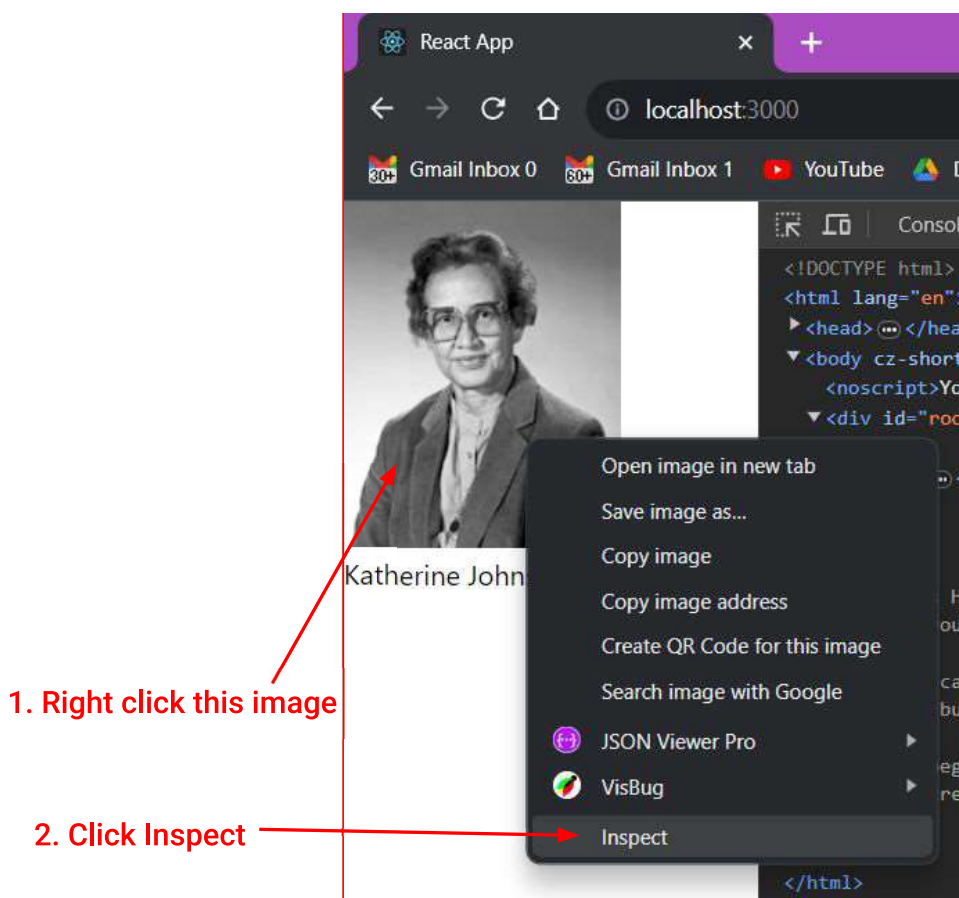
How to See Profile Component in Browser

16



Open Elements Tab of
Developer Tools

17



1. Right click this image

2. Click Inspect


18

React App

localhost:3000

Gmail Inbox 0 Gmail Inbox 1 YouTube Drive 0 Android Developm... Bard ChatGPT CodePath Andro

Console Application Network Elements Components Sources Perform

 Katherine Johnson

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <noscript>You need to enable JavaScript to run this app.</noscript>
  </head>
  <body cz-shortcut-listen="true">
    <div id="root">
      <div>
        <div>
          
          <div>Katherine Johnson</div>
        </div>
      </div>
    </div>
  </body>
</html>
```

This is markup of our Profile Component

19

Markups of Different Components

```
<div id="root"> == $0
  <div>
    <div>
      
      <div>Katherine Johnson</div>
    </div>
  </div>
</div>
```

This is markup of our Profile Component

This is markup of our App Component

This is markup present in index.html

20

View Page Source in browser (not actual source code in VS Code):

Location of React Code

JavaScript Code is bundled in this `bundle.js` file by bundler such as `webpack`.

```
27 <title>React App</title>
28 <script defer src="/static/js/bundle.js"></script></head>
29 <body>
30 <noscript>You need to enable JavaScript to run this app.</noscript>
31 <div id="root"></div>
```

Compiled successfully!

You can now view `myreactapp` in the browser.

Local: `http://localhost:3000`
On Your Network: `http://192.168.29.194:3000`

Note that the development build is not optimized.
To create a production build, use `npm run build`.

`webpack` compiled **successfully**

- On VSCode Terminal we can see working of `webpack`.
- If we do change and save any js file, `webpack` will recompile it automatically.
- This happens while app is running in development mode (`npm start`)

21

Open Network tab under Developer Tools

Name	Status	Type	Initiator	Size
localhost	200	document	Other	1.3 kB
bundle.js	200	script	(index):27	361 kB
MK3eW3Am.jpg	200	jpeg	react-dom.development.js:11038	7.0 kB
ws	101	websocket	WebSocketClient.js:13	0 B
manifest.json	200	manifest	Other	935 B
favicon.ico	200	x-icon	Other	3.9 kB
logo192.png	200	png	Other	5.7 kB

We may need to refresh the page to see this network activity

22

Network Tab under Developer Tools

This bundle.js is our React App.
We can see that this file gets downloaded to our browser when we visit React App.

Size of the React App

Name	Status	Type	Initiator	Size	Time	Waterfall
localhost	200	docum...	Other	1.3 kB	7 ms	
bundle.js	200	script	(index);27	361 kB	51 ms	
MK3eW3Am.jpg	200	jpeg	react-dom.de...	7.0 kB	449 ms	
ws	101	websoc...	WebSocketCli...	0 B	Pendi...	
manifest.json	200	manifest	Other	935 B	313 ms	
favicon.ico	200	x-icon	Other	3.9 kB	10 ms	
logo192.png	200	png	Other	5.7 kB	5 ms	

23

How to see React Components in Browser?

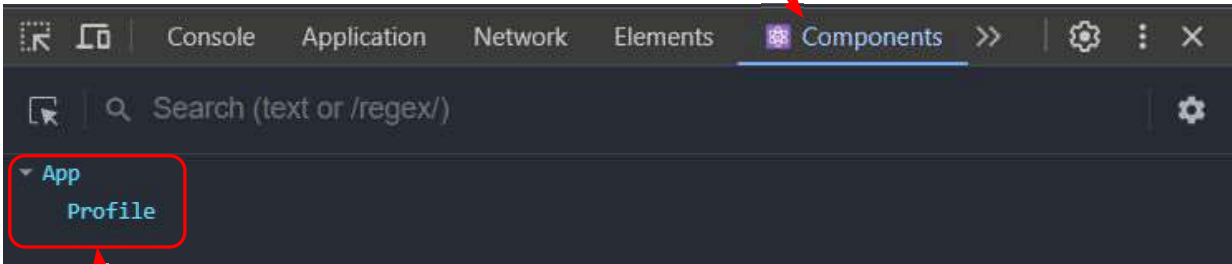
- In **Elements** tab, we can see **markups** related to our component.
- But, can we **see** our **React components**?
- We can see our React component using a **browser extension**:



24

React Developer Tools: Chrome Extension

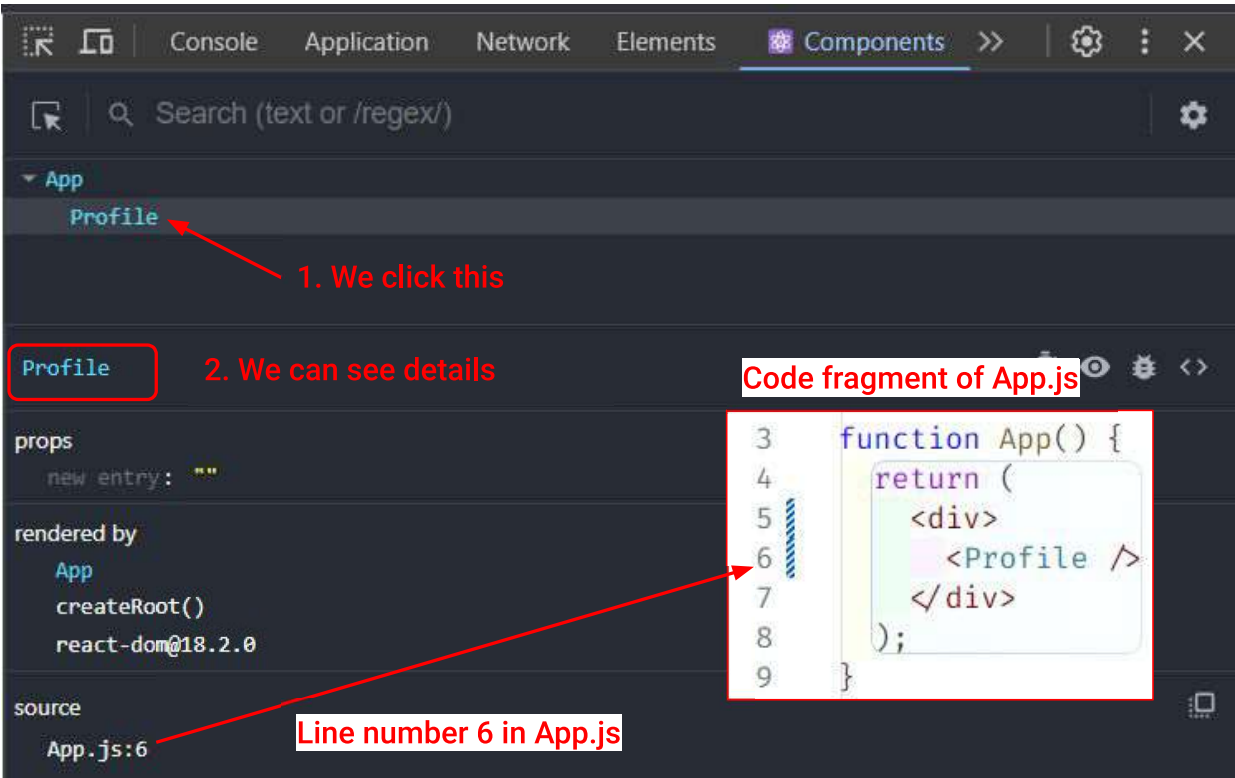
Once we install React Developer Tools chrome extension, a new tab named Components gets added in Chrome Developer Tools

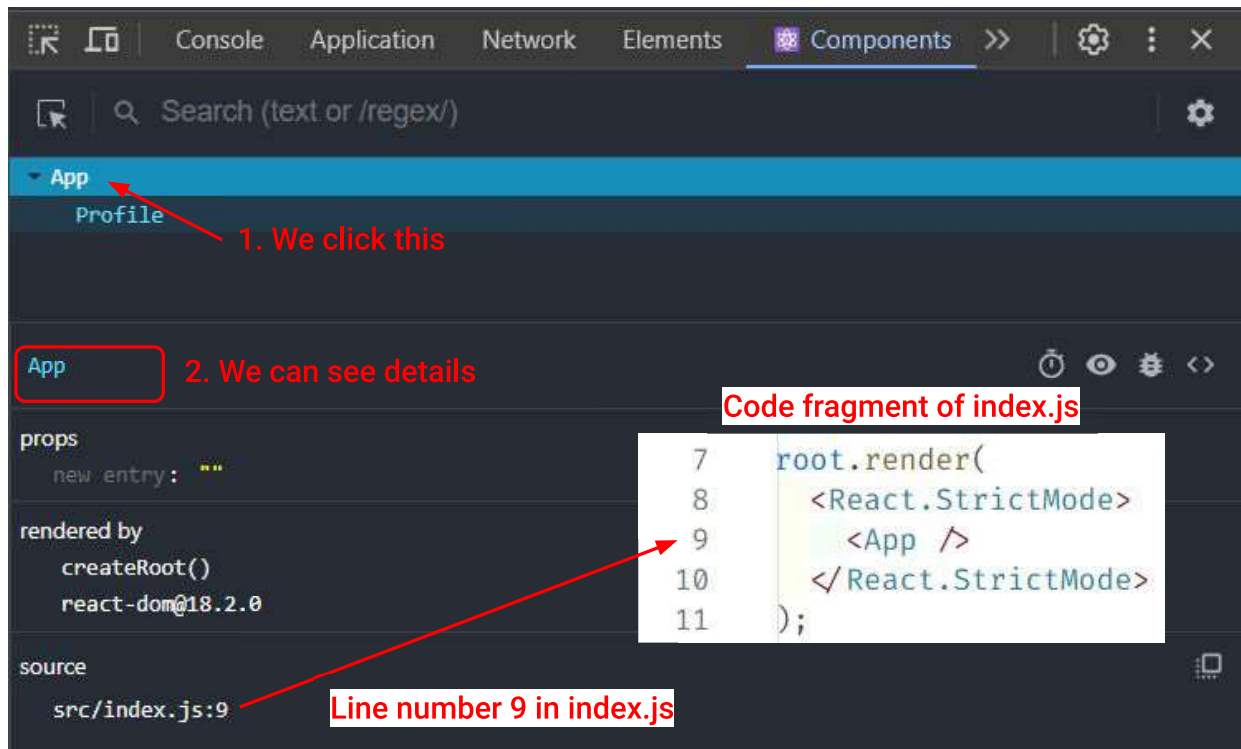


We can see our two components: Profile under App.

In source code, we have Profile component as a child component of App component.

```
3 function App() {
4   return (
5     <div>
6       <Profile />
7     </div>
8   );
9 }
```





27

Adding Styles to Profile Component

28

We add Styles to Profile Component:

- In HTML, style is added by using class attribute.
- In React, we add style by using className attribute.

```
Profile.js
src > components > Profile.js > ...
1  import React from 'react'
2
3  const Profile = () => {
4    return (
5      <div className="profile-card">
6        
11        <div className="profile-name">Katherine Johnson</div>
12      </div>
13    );
14  }
15
16  export default Profile
```

Style selector for profile card.

Style selector for profile photo.

Style selector for profile name.

29

```
index.css M X
Existing content of index.css
src > index.css > .profile-name
You, 3 seconds ago | 1 author (You)
1  body {
2    margin: 0;
3    font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto', 'Oxygen',
4      'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
5      sans-serif;
6    -webkit-font-smoothing: antialiased;
7    -moz-osx-font-smoothing: grayscale;
8  }
9
10 code {
11   font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
12     monospace;
13 }
```

30

We add Styles for the Three Selectors

```
14
15 .profile-card{
16   margin: 20px;
17   width: 200px;
18   display: flex;
19   flex-direction: column;
20   align-items: center;
21 }
22
23 .profile-photo {
24   border: 6px solid blueviolet;
25 }
26
27 .profile-name {
28   text-transform: uppercase;
29   font-size: 1.2rem;
30 }
```

We use flex module of css.

We provide border to the photo

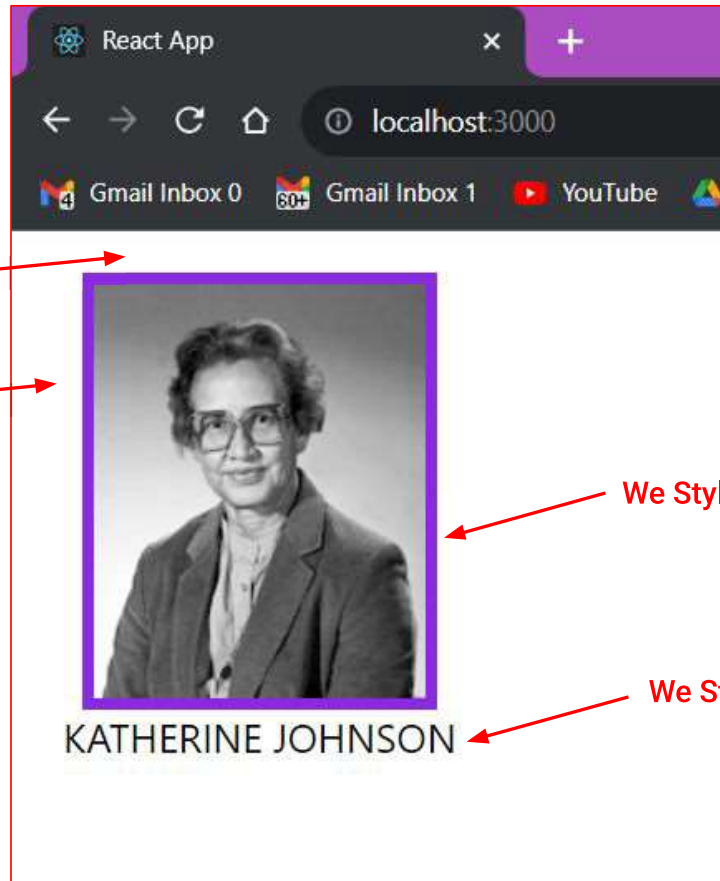
We transform text into uppercase and make it little bit big in size

Important note: flexbox (flex) is very useful. Explore it.

31

We Styled profile card

We can see these margins

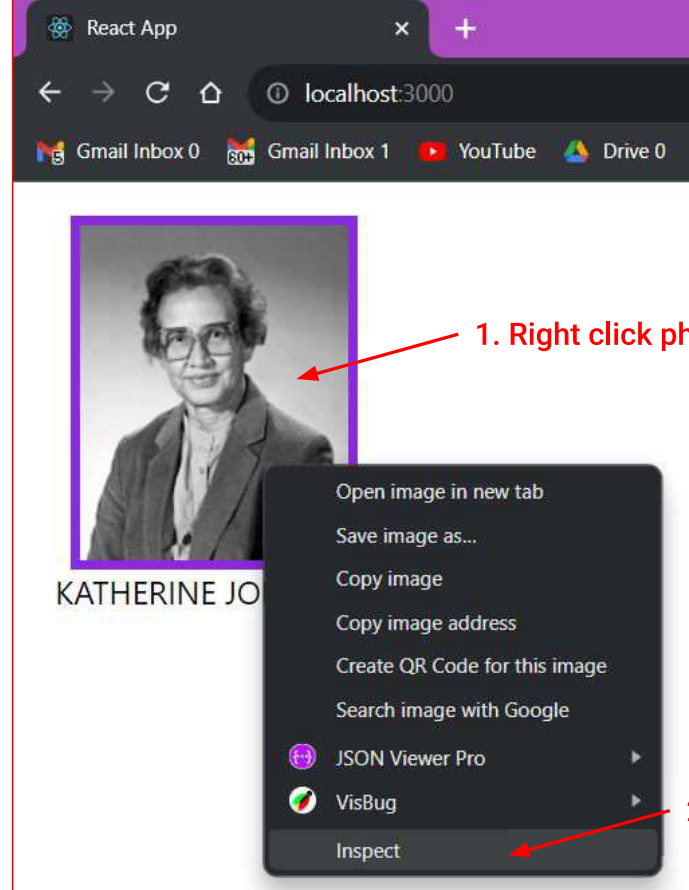


We Styled photo

We Styled name

32

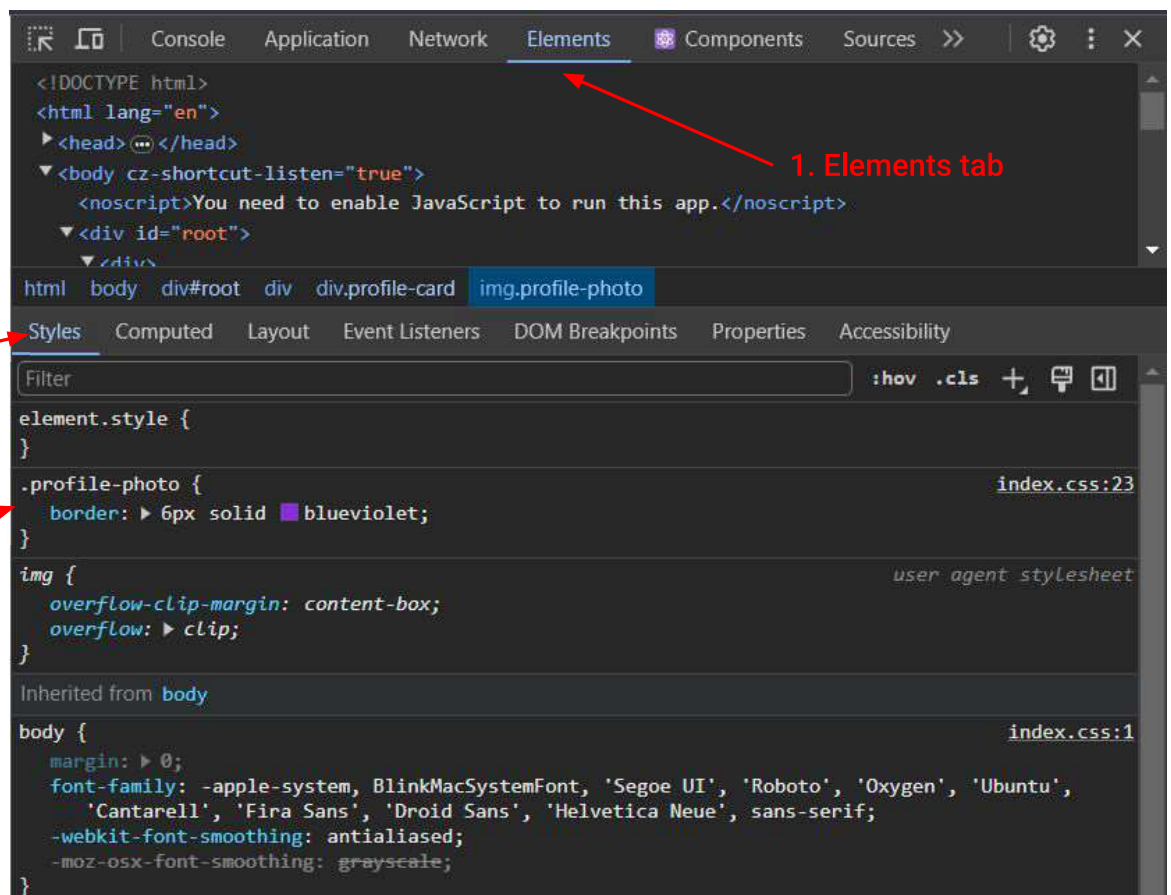
We can see styles applied to each element.



33

2. Styles tab

3. We can see the style for profile photo



34

Example: Multiple Instances of a Component

35

Example: Creating Multiple Instances of a Component

- Components are reusable.
 - We can use same component function or a component class to create multiple instances of it.
 - It is similar to instances of a class.
 - Each instance has its own copy of data members.
 - In component instances also, each instance of component has its own state, props, and presentation (DOM) on UI.

36

Create Two Instances of Profile Component

```
App.js
src > App.js > ...
You, 7 minutes ago | 1 author (You)
1 import Profile from "../components/Profile";
2
3 function App() {
4   return (
5     <div>
6       <Profile />
7       <Profile />
8     </div>
9   );
10 }
```

Create another instance of Profile component

37



Each instance has its own presentation (DOM) on UI

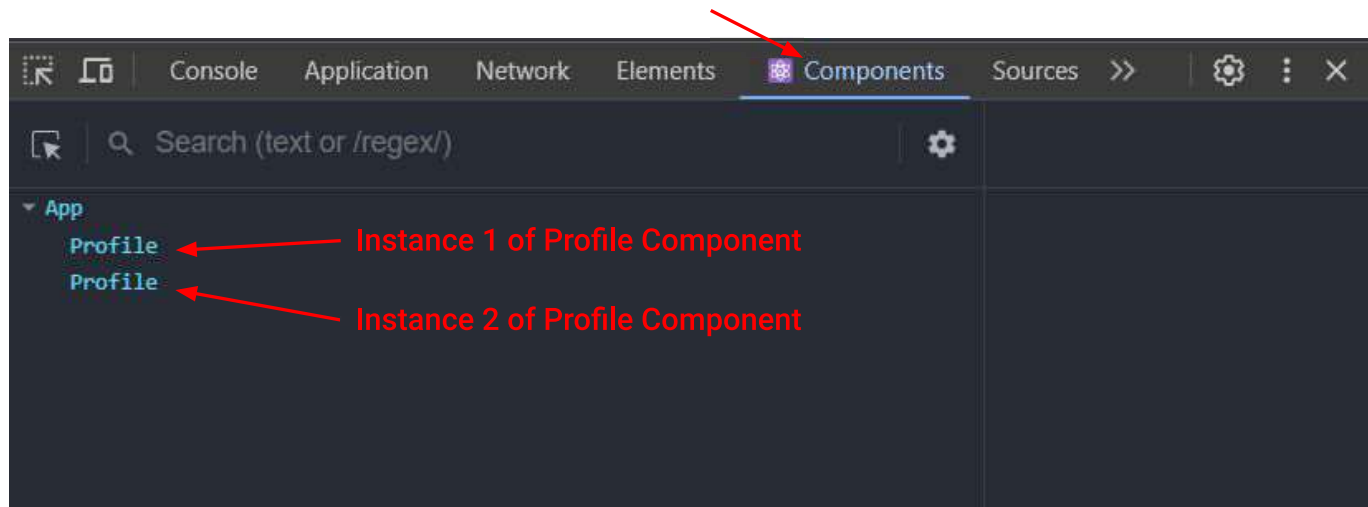
```
App.js
src > App.js > ...
You, 7 minutes ago | 1 author
1 import Profile from
2
3 function App() {
4   return (
5     <div>
6       <Profile />
7       <Profile />
8     </div>
9   );
10 }
```

Later, we will provide different data to different instances.

38

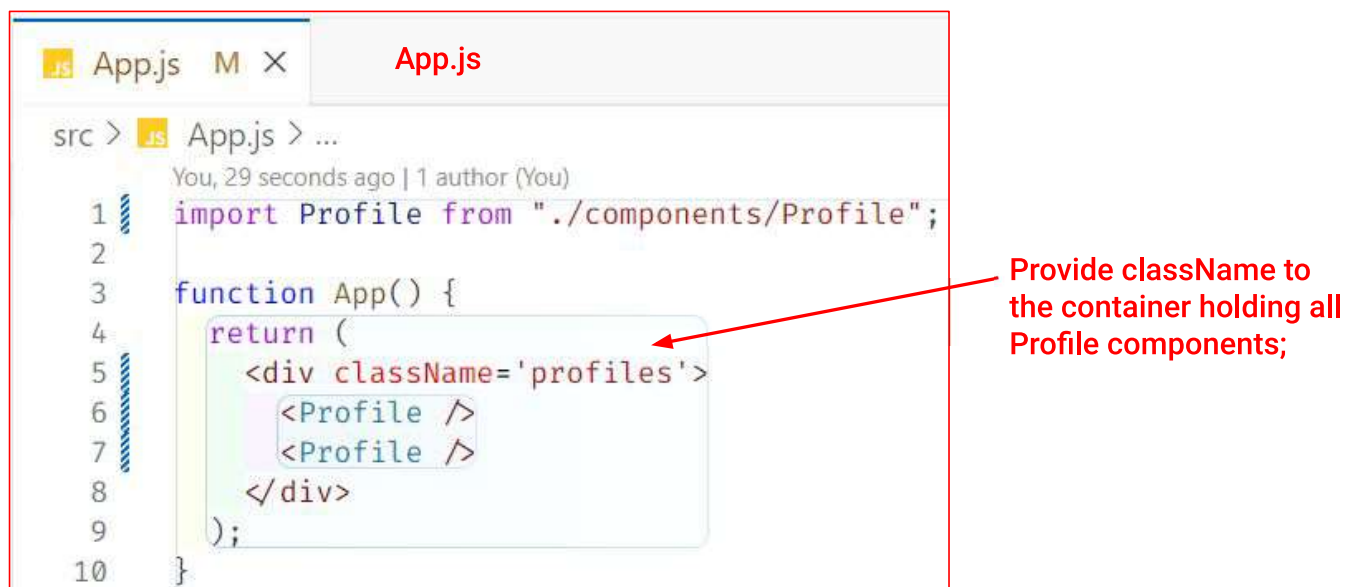
React Developer Tools

- We can also observe multiple instances in React Developer Tools.



39

Make All Profile Cards Horizontal



40

```

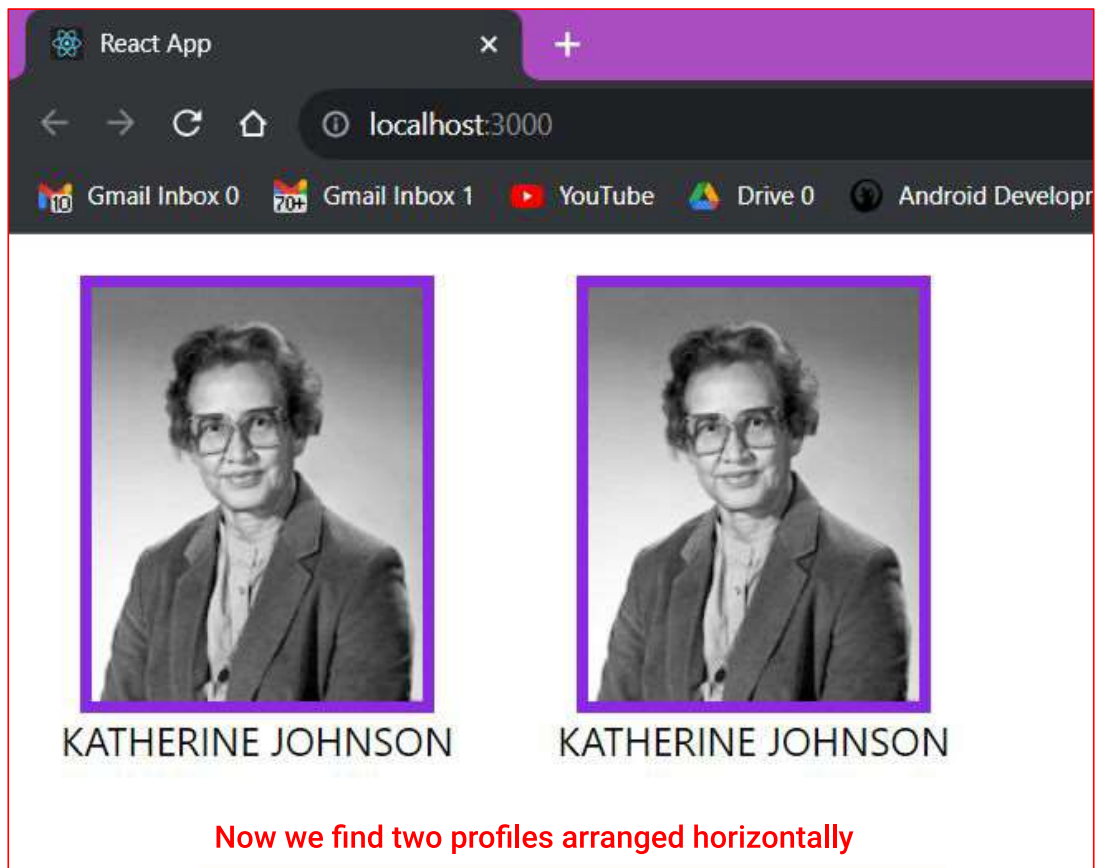
15 .profile-card{
16   margin: 20px;
17   width: 200px;
18   display: flex;
19   flex-direction: column;
20   align-items: center;
21 }
22
23 .profile-photo {
24   border: 6px solid blueviolet;
25 }
26
27 .profile-name {
28   text-transform: uppercase;
29   font-size: 1.2rem;
30 }
31
32 .profiles{
33   display: flex;
34   flex-direction: row;
35 }

```

index.css

We provide styles to
profiles class selector

41



42

References

- <https://react.dev/learn>