1. **What are various ways to add images into our app? Explain with code examples.**

* The HTML <img> tag is used to embed an image in a web page.
* The HTML <map> tag defines an image map. An image map is an image with clickable areas. The areas are defined with one or more <area> tags.
* CDN, external files, url
* To add a background image on an HTML element, use the HTML style attribute and the CSS background-image property
* The HTML <picture> element allows you to display different pictures for different devices or screen sizes.

1. **What would happen if we do console.log(useState())?**

* It returns an array with two values: 1. Undefined and 2. Function

1. **How will useEffect behave if we don’t add a dependency array?**

* If there is no dependency array, then the useeffect code will run after every render cycle.

1. **What is SPA?**

* An SPA (Single-page application) is a web app implementation that loads only a single web document, and then updates the body content of that single document via JavaScript APIs such as [XMLHttpRequest](https://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest) and [Fetch](https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API) when different content is to be shown.
* This therefore allows users to use websites without loading whole new pages from the server, which can result in performance gains and a more dynamic experience, with some tradeoff disadvantages such as SEO, more effort required to maintain state, implement navigation, and do meaningful performance monitoring.

1. **What is difference between Client Side Routing and Server Side Routing?**

* Server Side Routing -
  + When browsing, the adjustment of a URL can make a lot of things happen.
  + This will happen regularly by clicking on a link, which in turn will request a new page from the server.
  + This is what we call a server-side route. A whole new document is served to the user.
  + A server-side request causes the whole page to refresh.
  + This is because a new GET request is sent to the server which responds with a new document, completely discarding the old page altogether.
  + Pros –
    - A server-side route will only request the data that’s needed. No more, no less.
    - Because server-side routing has been the standard for a long time, search engines are optimized for webpages that come from the server.
  + Cons –
    - Every request results in a full-page refresh. That means that unnecessary data is being requested.
    - A header and a footer of a webpage often stays the same. This isn’t something you would want to request from the server again.
    - It can take a while for the page to be rendered. However, this is only the case when the document to be rendered is very large or when you have slow internet speed.
* Client Side Routing –
  + A client-side route happens when the route is handled internally by the JavaScript that is loaded on the page.
  + When a user clicks on a link, the URL changes but the request to the server is prevented. The adjustment to the URL will result in a changed state of the application.
  + The changed state will ultimately result in a different view of the webpage.
  + This could be the rendering of a new component, or even a request to a server for some data that the application will turn into some HTML elements
  + It is important to note that the whole page won’t refresh when using client-side routing. There are just some elements inside the application that will change.
  + Pros –
    - Because less data is processed, routing between views is generally faster.
    - Smooth transitions and animations between views are easier to implement.
  + Cons –
    - The whole website or web-application needs to be loaded on the first request. That’s why the initial loading time usually takes longer.
    - Because the whole website or web-application is loaded initially, there is a possibility that there is data downloaded for views you won’t even come across.
    - It requires more setup work or even a library. Because server-side is the standard, extra code must be written to make client-side routing possible.
    - Search engine crawling is less optimised. Google is making good progress on crawling single-paged-apps, but it isn’t nearly as efficient as server-side routed websites.