THEORY QUESTIONS ASSIGNMENT
Full Stack Stream

1. React.js is an open source front-end JavaScript library for building Web applications. The main features of React use :

- is declarative : the programmer write the code with the desired output and React will interpret how to get the desired output
- React.js is component based (reac app contains components that represent specific screens/elements in the app)
- React.js supports server side (Server-side rendering (SSR) is when content on your webpage is rendered on the server and not on your browser using JavaScript.)
- React.js is extensive

The main advantages of React js are :

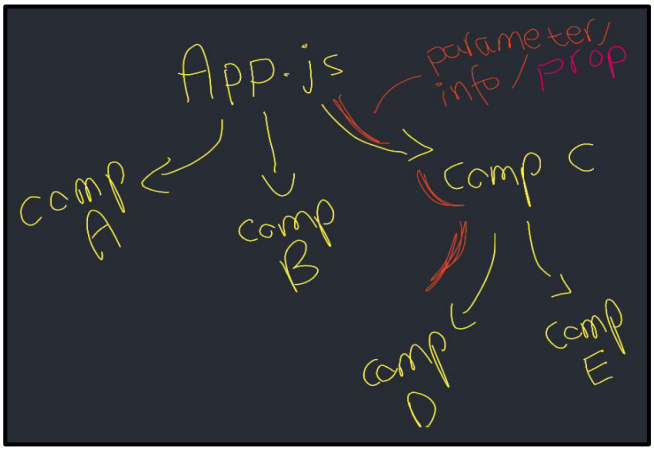
- it is easy to add to your website : React can be added partially to our websites (like HTML websites for example) or make the entire website be React-based
- easy to update : thanks to Virtual DOM (see the definition below), the local change on your laptop is loading on the website without reloading the page while HTML website requires to refresh/reload the website for each change
- React.js is simple and easy to learn :with a basic previous knowledge in HTML/CSS programming, you can easly understand React in contrast with Angular and Ember which are difficult to learn.
- Testability : ReactJS applications are super easy to test.
- Native Approach React can be used to create mobile applications
- Easy to debug
- reusable code : you code one specific customized component (as a button for example) which can be used multiples times on your website
- is more web dev customizability than standard HTML and CSS

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

DOM:stands for Document Object Model is a platform and language- neutral interface that allows programs and scripts to change the structure,style and content of a document.

Virtual DOM : In React, for every DOM object, there is a corresponding "virtual DOM object." A virtual DOM object is a representation of a DOM object, like a lightweight copy. Manipulating the DOM is slow while manipulating the virtual DOM is fast,

2. Props (short for properties) is a JS object which allows to pass data between React components (from Parents to child components).

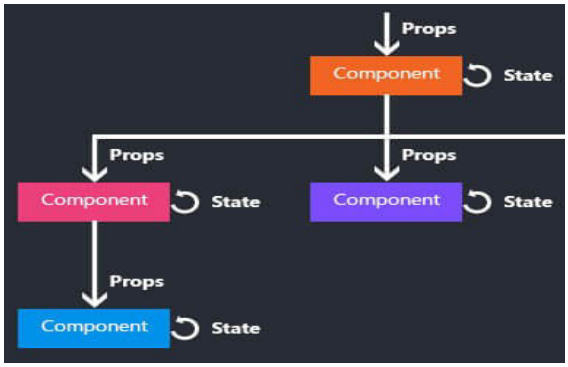


State is as well a JS Objects which contains information and attributes about the component and allows components to manage their own data.

Props vs State

- Props allows components to receive data from outside (child compoennt receive data from parent component) while State doesn't create any data link between parent and child components, State allows component to create and manage their own data
- Props are used to pass data, State is for creating/managing data
- State is changeable while Props is immutable :State allows to change the data inside the component while props doesn't allow it

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/



3. React hooks are reusable functions that let you hook into React State and lifecycle features from function components. It is a mechanism that allows us to access State and Lifecycle methods in functional components, hence not restricting our development to be exclusively being class components.

The main features of React hooks :

- Hooks can only be called inside React function components
- Hooks can only be called at the top level of a component
- Hooks cannot be conditional

Lifecycle Methods vs Hooks

Lifecycle methods are incorporated with class based components while Hooks are incorporated with Function based components

Construct and use and binding of 'this' are required for Lifecycle methods but not for Hooks

4. To define the perfect door, it is important to understand what would be the environment. For example, we need to define if the door is to enter a business building or a door to enter a

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

family house. If the door is to enter a business building, you will focus on the easy accessibility to building while for a family house, you will focus on the safe accessbility to the family house as you don't want anyone to have easy access to the family house except their occupants.

For my perfect door, i would like to focus on a business house access door.

As well, to define the perfect door, it is important to understand who will be the stakeholders and take in consideration the ability of everyone in or outside the house. For example, a safety system on the door should be implemented if the user is a child. The door system should be comprehensive and usable for everyone (people with disabilties for example).

Now, that we define our basic requirements for the perfect door, let's describe the design of this door.

The perfect door design :

- has an automatic sliding open system with presence detectors : to reduce the mental workloads of the user and make it natural. They don't have to think if they have to pull or push the door as the door is opening itself. It allows as well for blind people to not have to look for the door, for persons in wheelchair and for people carrying things or children to not have to struggle to open the door.

- Has a « welcome »sign where it is written «this is a automatic door, please show your face and wait for the door to open. If you are not working in the building, say «call reception» to reach the reception»:for standardisation

- has a « welcome » automatic message where it said« this is a automatic door, please show your face and wait for the door to open. If you are not working in the building, say « call reception » to reach the reception» for blind persons and as well for the users who didn't see the sign and try to open the door manually by mistake (design for error). The message will be repeated every 10/15 seconds.

- is plastic transparent : the users can see from inside/outside the building and see if someone is coming to the door

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

5. Angular vs React

- Angular is an open-sourced JavaScript framework built using Typescript (a more specialised JavaScript form) for web and mobile development. while React is a Javascript library and built using JSX.

- Angular can do everything as everything built in so can handle routing, nuanced operation etc without any additionnal libraries. React needs the assistance of additionnal libraries to handle everything.

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

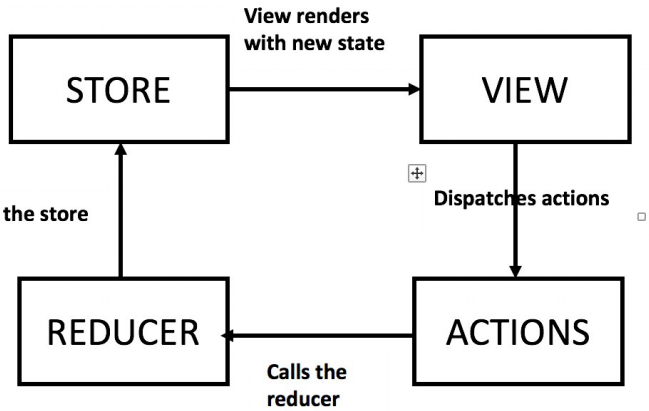
- If the programmer has basic knowledge in javascript, React is easy to learn while Angular is long and difficult to learn
- App structure is fixed and complex component-based Model/Controller/View with Angular while the app structure is flexible component-based View

- Data binding is bidirectionnal and data is mutable with Angular while data binding is unnidirectional and data is immutable with React

- Angular change the Reald DOM while React is changing Virtual DOM

6. Redux is a predictable state container for JavaScript apps. It can be described as an open-source JavaScript library which is used to manage application state. Information can be stored globally in a Redux 'store' which allows components to have a centralised bank of information that they can access anytime. Redux helps you write applications that behave consistently, run in different environments ,are easy to test and provides a great developer experience

   Redux set up :
   - The entirety of the app is surrounded / nested by a **'Provider'** component which makes the Redux stor available for all other descending components in the hierarchy
   - Components then utilise React hooks provided from Redux to interact with the store - **useSelector** links you to the store (e.g. to read information, subscribe for updates if any changes occur) whilst **useDispatch** is used for sending **action (where action** are effectively 'information changes')
   - then 'reducers' receive the action, and determine **how the Store should be modified** based on the action type



https://www.c-sharpcorner.com/arti
https://blog.logrocket.com/why-yo
https://skillcrush.com/blog/what-is
https://www.codecademy.com/artic
https://www.w3schools.com/js/js_h
%20document.%22
https://reactjs.org/docs/hooks-over
https://www.opcito.com/blogs/lifec
https://www.freecodecamp.org/new
https://www.educba.com/advantage
https://ubuntu.com/tutorials/install-
%20of%20a
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

Redux advantages are :

- making easier to share information between components
- Redux is maintainable
- Debugging/testing is easy in Redux

Redux disadvantages are :

- security issues because of no-encapsulation : component can access the data
- Restricted design
- excessive use of memory.: as state is immutable in redux, the reducer updates the state by returning a new state every time

How to install Redux ?

- Run npm install @reduxjs/toolkit in commands
- Create a react redux project by running npx create-react-app my-app --template redux in commands

7. Linux History
   UNIX was the historical simple operating system that allowed everyone to use computer easily compared to alternatives but was not open source. After a disagreement with a printer company XEROX refusing to share their code for diagnostic purposes for his

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

printer ,a MIT student called Richard Stallman started the Free Software Foundation and ('GNU's not UNIX') Project. His idealogical is that code should be free and open to all. GNU was supposed to be a open source replacement to UNIX. Unfortunately, they took too long to create the OS kernel (important component of an operating system, it allows the hardware to talk to the software) and another student started his one OS's kernel. Indeed, Linus Torvald succeed to create a complete free open source OS kernel for GNU called « Linux ». Rapidly, Linux became very popular and the main kernel for GNU.

Why someone will use Linux over Windows/MACOS ?

- being free
- being an open source
- being secured , antivirus is unnecessary but affordable compared to MACOS which is secured too but expensive
- having a dedicated community full of technology addicts and scientists who are ready to help in your issues and give you a full support
- easily being used and installed on older computer systems
- fast software updates
- customisation:features on linux can be added, deleted or customised based on your needs
- fast and easy to install
- being transparent
- Extensive use of command line to do things which is very useful to being comfortable with while coding/programming

How to install Linux ?

To install Linux, you will first need to choose a Linux distribution, the most common Linux distribution is Ubuntu. To Install Ubuntu, please follow the steps below :

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

- Download an Ubuntu Image
- Create a Bootable USB stick
- Boot from USB flash drive
- Choose your Installation Setup
- Choose Drive Management

8. Functional component are one of the most common components in React which are plain JavaScript functions.

Class components are simple classes which contains multiple functions that add functionality to the application.

Class component vs Functional components

- A functional accepts props as an argument and returns a React element while a class component requires you to extend from React component and create a render function (render is a method to show the output in the browser.)which returns a React element

- there is no render method used in functionnal component but it is mandatory to have a render method with class components

- React lifecycle methods can't be used wit functional component but can be used with class components
- Hooks is more difficult to use in class components than functional components
- Syntax is different : Functional component is written shorter and simpler and easier to develop, understand and test
- it is hard to reus stateful lofic between components in the class components
- Class component can be confusing and complex components are hard to understand in class component

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/

- functional component are more updated and are the main focus for React team, they even recommend that new apps should be built with functional components and hooks
- React team announced that React docs will be focusing on React using functional component and hooks which therefore functional components are more supported and more documented than class components

https://www.c-sharpcorner.com/article/what-and-why-reactjs/
https://blog.logrocket.com/why-you-should-render-react-on-the-server-side-a50507163b79/
https://skillcrush.com/blog/what-is-react-js/
https://www.codecademy.com/article/react-virtual-dom
https://www.w3schools.com/js/js_htmldom.asp#:~:text=The%20DOM%20defines%20a%20standard,and%20style%20of%20a%20document.%22
https://reactjs.org/docs/hooks-overview.html
https://www.opcito.com/blogs/lifecycle-methods-and-hooks-in-react#
https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/
https://www.educba.com/advantage-of-linux/?source=leftnav
https://ubuntu.com/tutorials/install-ubuntu-desktop#6-drive-management
https://blog.boardinfinity.com/working-with-redux-pros-and-cons/
https://medium.com/@chandinibaratam/why-and-when-to-use-redux-b57f7dae9269
https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/