Namaste React

Homework Day – 4

Q1. Is JSX mandatory for react?

Ans – JSX is not a requirement for using React. Using React without JSX is especially convenient when you don't want to set up compilation in your build environment.

Q2. Is ES6 mandatory?

Ans – It is not mandatory but is highly recommendable. The latest projects created on React rely a lot on ES6.

Q3. What is <React.Fragment> </React.Fragment> and <> </>?

Ans – React Fragment is a feature in React that allows you to return multiple elements from a React component by allowing you to group a list of children without adding extra nodes to the DOM.

Q4. What is Virtual Dom?

Ans - The virtual DOM (VDOM) is a programming concept where an ideal, or "virtual", representation of a UI is kept in memory and synced with the "real" DOM.

Q5. What is Reconciliation in React?

Ans - The mechanism to diff one tree with another to determine which parts need to be changed and then update the original DOM with it is called Reconciliation

Q6. What is React Fiber?

Ans - React Fiber is an ongoing reimplementation of React's core algorithm. The goal of React Fiber is to increase its suitability for areas like animation, layout, and gestures. Its headline feature is incremental rendering: the ability to split rendering work into chunks and spread it out over multiple frames. Other key features include the ability to pause, abort, or reuse work as new updates come in; the ability to assign priority to different types of updates; and new concurrency primitives.

Q7. Why and when do we need keys in React?

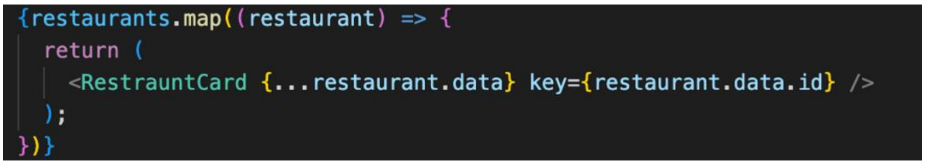
Ans - Keys help React identify which items have changed, are added, or are removed. Keys should be given to the elements inside the array to give the elements a stable identity. The best way to pick a key is to use a string that uniquely identifies a list item among its siblings. When you don't have stable IDs for rendered items, you may use the item index as a key as a last resort.

Q8. What are props in React? Ways too?

Ans –

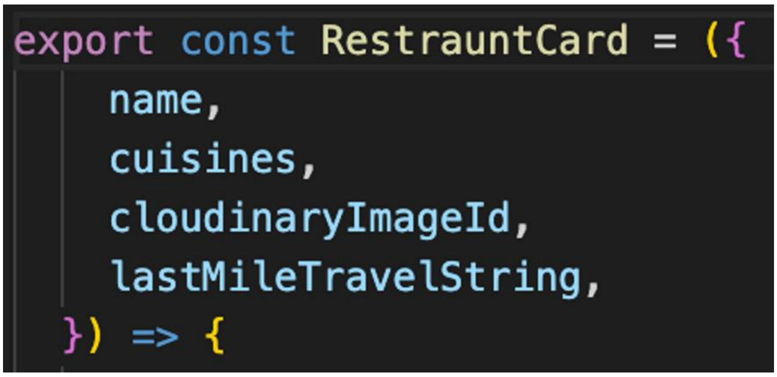
React Props are like function arguments in JavaScript *and* attributes in HTML. Props are also how you pass data from one component to another, as parameters.

Way 1: - pass the whole array in the props: -



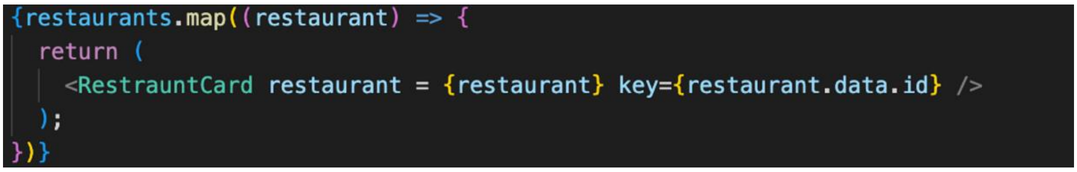
```
{restaurants.map((restaurant) => {
  return (
    <RestrauntCard {...restaurant.data} key={restaurant.data.id} />
  );
})}
```

and destructor like this: -



```
export const RestrauntCard = ({
    name,
    cuisines,
    cloudinaryImageId,
    lastMileTravelString,
  }) => {
```

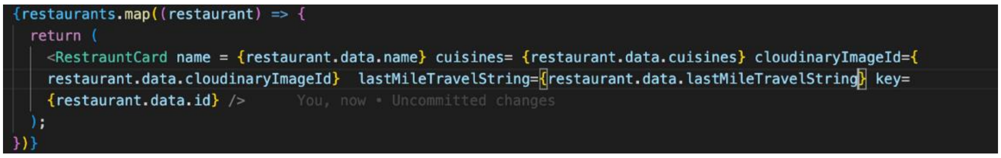
Way 2: - pass as a prop



```
{restaurants.map((restaurant) => {
  return (
    <RestrauntCard restaurant = {restaurant} key={restaurant.data.id} />
  );
})}
```

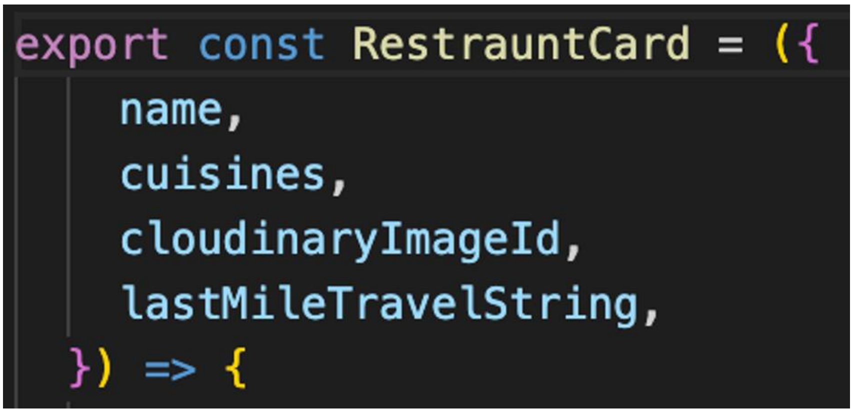
and use it like {props.restaurant.data.name}

Way3: - send each prop in a component



```
{restaurants.map((restaurant) => {
  return (
    <RestrauntCard name = {restaurant.data.name} cuisines= {restaurant.data.cuisines} cloudinaryImageId={
    restaurant.data.cloudinaryImageId}  lastMileTravelString={restaurant.data.lastMileTravelString} key=
    {restaurant.data.id} />      You, now • Uncommitted changes
  );
})}
```

and use it like this: -



```
export const RestrauntCard = ({
    name,
    cuisines,
    cloudinaryImageId,
    lastMileTravelString,
  }) => {
```

Q9. What is a Config Driven UI?

Ans - Config Driven UI is nothing but a website that is handled by the backend. What I mean is it is a configurable website where backend configuration handles what needs to be displayed on the front end and what does not according to region, city, state, or country. It helps to keep the same design but different content accordingly.
