INNOVIZE ESKILLBRIDGE EDUTECH GLOBAL LLP

What is React?

React is a **JavaScript library** used for building user interfaces, particularly **web applications**. It allows developers to create reusable **components** that form the different parts of a web page. Instead of creating an entire webpage from scratch every time, React enables developers to create individual components (like buttons, forms, etc.) and then combine them to build dynamic web pages.

Let's compare this to an **employee system**. Imagine a website where employees can view their profiles, update details, and see tasks. In React, each part of the webpage—like the profile card, task list, or even the button to update details—can be a **component**. You don't need to recreate the page each time something changes. React updates only the part of the page that has been modified (like when an employee's profile is updated), making it **fast** and **efficient**.

Example: Employee Profile in React

Let's say you want to display an employee's profile with details like name, position, and department. In React, you'd create a **component** like this:

javascript

Copy code

```
function EmployeeProfile(props) {
  return (
    <div>
      <h2>{props.name}</h2>
      <p>Position: {props.position}</p>
      <p>Department: {props.department}</p>
    </div>
  );
}
```

In this example:

- EmployeeProfile is a **component** that takes in **props** (properties like name, position, and department) and returns a piece of the webpage displaying that information.
- When the data changes (e.g., the employee's position is updated), React will **automatically update** only that specific part of the page, not the entire webpage.

Why React?

React is popular because it's **fast, flexible**, and makes coding more **manageable**. Let's go back to the employee system example:

1. **Reusability**: With React, you can reuse components across different pages. For instance, if you have an employee profile card, you can use the same card on the homepage, dashboard, and profile page. This reduces the need to write the same code multiple times.

2. **Efficiency**: React uses a concept called the **Virtual DOM**. This means that instead of refreshing the entire webpage when something changes, React only updates the specific part that needs to be changed. Imagine having hundreds of employees listed on a page. If one employee updates their profile, React won't reload all the profiles—it will just update the changed one.
3. **Component-based**: Everything in React is broken down into components. Think of components like building blocks. For example, in the employee system, you may have a **Profile** component, a **Task List** component, and an **Update Button** component. Each of these can be managed individually, making the code easier to write and maintain.
4. **Strong Community Support**: Since React is widely used by large companies like Facebook (who created React), there is a large community of developers constantly improving it. This means if students face any challenges, they can easily find solutions and resources.

Example: Dynamic Employee Task List

If we want to show a list of tasks for an employee, we can use React to display tasks dynamically:

javascript

Copy code

```
function TaskList(props) {
  return (
    <ul>
      {props.tasks.map((task, index) => (
        <li key={index}>{task}</li>
      ))}
    </ul>
  );
}
```

In this code:

- The TaskList component takes in an array of tasks (props) and displays each one in a list.
- If new tasks are assigned to the employee, React will automatically update the list without reloading the whole page.

Conclusion

React is a powerful tool because it **simplifies** how we build web applications by focusing on **components** that can be reused and updated efficiently. For a 10th-grade student learning React, it's like building a web page with **Lego blocks**—you create one block (component) and reuse it wherever needed.

This combination of **speed**, **reusability**, and **efficiency** is why React is a preferred choice for developers and companies around the world. It makes building dynamic websites easier and more organized, especially when handling systems like an employee management system.