TRAINOSYS

REACTJS & TYPESCRIPT

Day 2

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

Course Overview

- Introduction to ReactJS and TypeScript (Day 1)
- React Component Development (Day 2)
- React Routing and Data Fetching (Day 3)
- State Management with Redux and TypeScript (Day 4)
- Advanced Topics and Project Development (Day 5)

Program

Day 01

- Introduction to ReactJS and it's benefits
- Setting up the development environment with NodeJS and npm
- Creating a new React project with Typescript using Create React App
- Understanding the basics of TypeScript: types, interfaces, and modules
- Building a simple React component with TypeScript

Day 02

- Recap of React component structure and lifecycle methods
- Creating functional components with TypeScript
- Working with props and prop types in TypeScript
- State management in React components using hooks with TypeScript
- Building a simple React component with TypeScript
- Handling events and form inputs in TypeScript-based components

TRAINOSYS

REACTJS + TYPESCRIPT

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

RECAP

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

React Code Example

```
import * as React from "react";

class HelloWorld extends React.Component {
  render() {
    return (
      <div>
        <h1>Hello Delegates</h1>
        <p>And good morning!</p>
      </div>
    );
  }
}
```

imports the React Library
* (imports all)

New Class Component called "Hello World"
that extends React.Component

responsible for returning JSX
JSX - JavaScript XML

JSX - defining our HTML

TRAINOSYS

Let’s create our first project with ReacJS + Typescript

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

Starter Kit using Vite

```
npm create vite@latest
npm create vite@4.1.0
```

Generate a project template for ReactJS & TypeScript and many more

STEP 1 →

```
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs> npm create vite@latest
? Project name: » vite-project
```

STEP 2 →

```
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs> npm create vite@latest
√ Project name: ... my-app
? Select a framework: » - Use arrow-keys. Return to submit.
    Vanilla
    Vue
>   React
    Preact
    Lit
    Svelte
    Solid
    Qwik
    Others
```

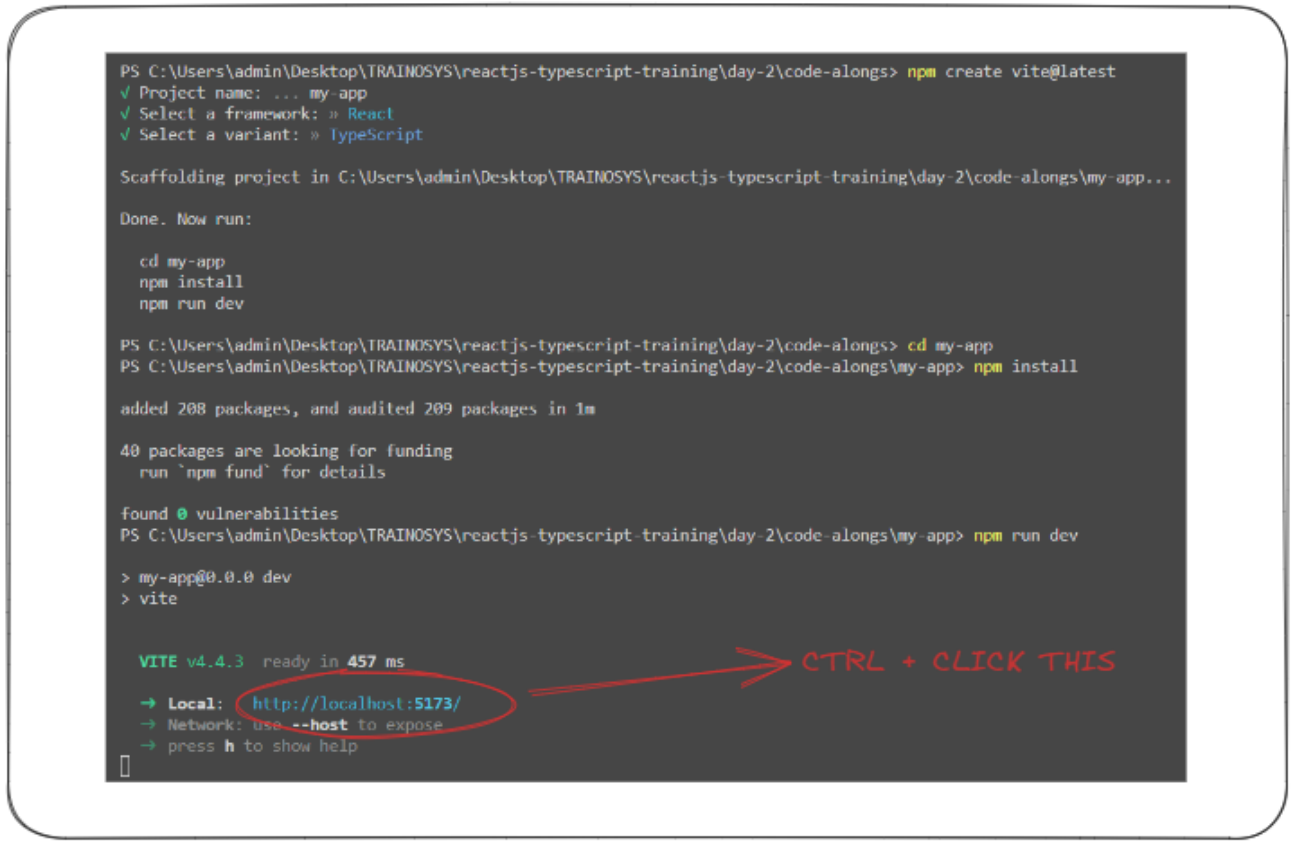
STEP 3 →

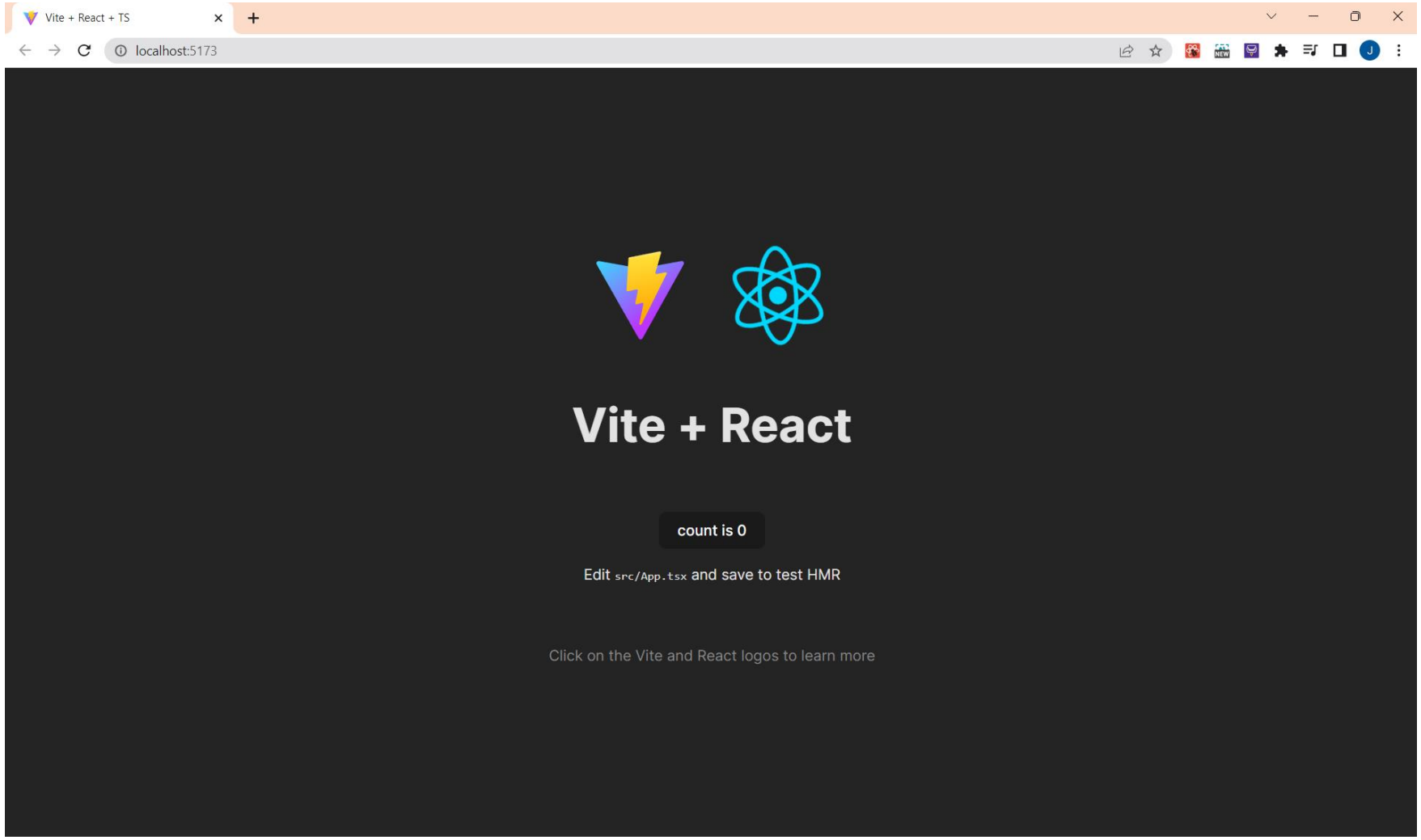
```
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs> npm create vite@latest
√ Project name: ... my-app
√ Select a framework: » React
? Select a variant: » - Use arrow-keys. Return to submit.
>   TypeScript
    TypeScript + SWC
    JavaScript
    JavaScript + SWC
```

STEP 4 →

```
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs> npm create vite@latest
√ Project name: ... my-app
√ Select a framework: » React
√ Select a variant: » TypeScript

Scaffolding project in C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs\my-app...

Done. Now run:

  cd my-app
  npm install
  npm run dev

PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs>
```

STEP 5 →


PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs> npm create vite@latest
√ Project name: ... my-app
√ Select a framework: » React
√ Select a variant: » TypeScript
Scaffolding project in C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs\my-app...
Done. Now run:
cd my-app
npm install
npm run dev
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs> cd my-app
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs\my-app> npm install
added 208 packages, and audited 209 packages in 1m
40 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\admin\Desktop\TRAINOSYS\reactjs-typescript-training\day-2\code-alongs\my-app> npm run dev
> my-app@0.0.0 dev
> vite
VITE v4.4.3 ready in 457 ms
➜ Local: http://localhost:5173/
➜ Network: use --host to expose
➜ press h to show help
CTRL + CLICK THIS



CRA

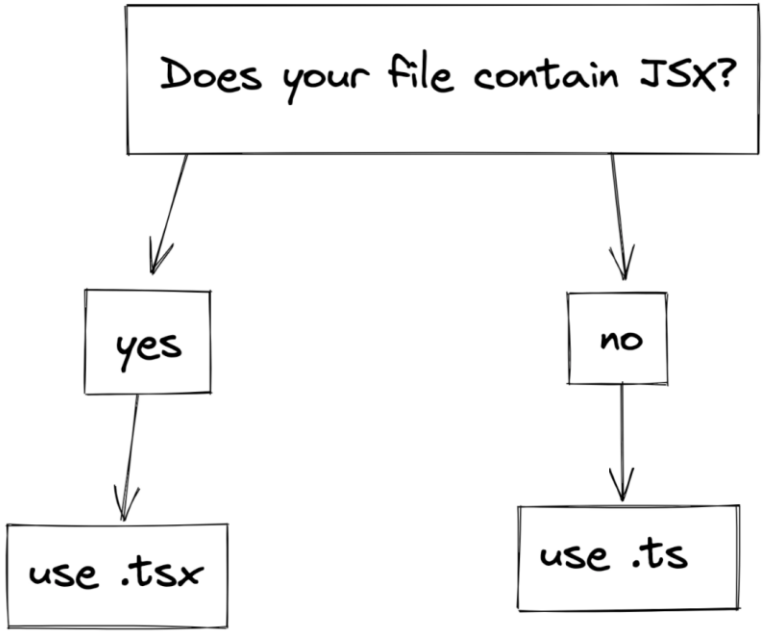
- create-react-app
- Great for beginners
- Easy to use and has some great features
- Uses Webpack under the hood which makes quite slow
- Webpack is responsible for bundling and optimizing your project's code and assets

VITE

- Modern JS Build Tool
- Focuses on developer experience by leveraging ES browser module support
- In other words: VITE IS FASTER AND MODERN

.tsx and .ts files

.tsx vs .ts



TRAINOSYS

CODE ALONG

HYBRID
TECHNOLOGY
COURSES

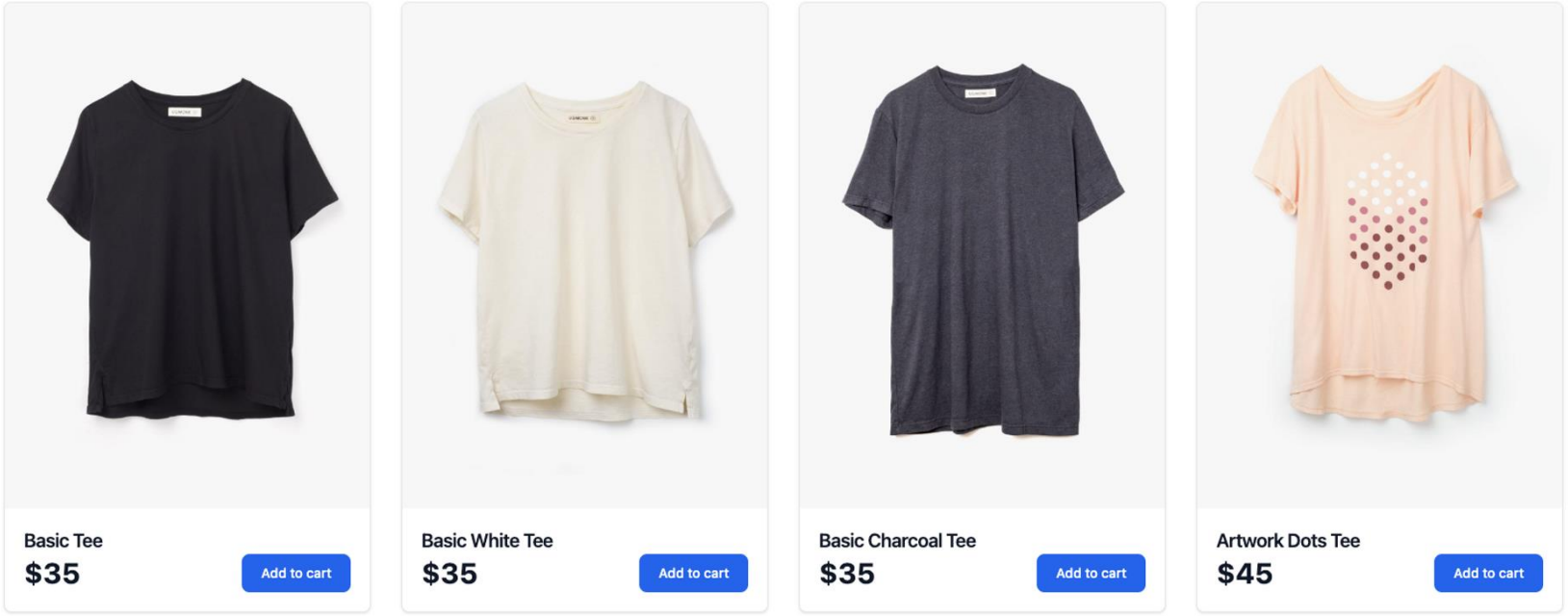
TRAINOSYS

Activity 1

- ❑ npm create vite@latest / npm create vite@4.1.0
- ❑ Project name: rts-d2-act-one

- ❑ Create a reusable <Card /> component with TypeScript
- ❑ It takes props for imageUrl, title, price, buttonText

Activity 1 – expected output



TRAINOSYS

REACT DATA

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

React Data

- ✓ Props
- ✓ State
- ❑ Events
- ❑ Two-way binding
- ❑ Conditionals

Events

- A react component can specify event handlers
- Just like DOM, use onClick for buttons, onChange for inputs, etc

Events – Data Binding

TRAINOSYS

App.tsx

```tsx
const [firstName, setFirstName] = useState<string>("");
const [age, setAge] = useState<number>(0);

const handleFirstNameChanged = (e: ChangeEvent<HTMLInputElement>): void => {
   setFirstName(e.target.value);
}

return (
    <input
        type="text"
        value={firstName}
        onChange={(e) => handleFirstNameChanged(e)}
    />
)
```

CodeImage

TRAINOSYS

REACT JSX – CONDITIONALS

- React won't render false, null or undefined values in curlies
- You can't use if but you can use conditionals and ternary

TRAINOSYS

ACTIVITY – Kahoot!

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

ACTIVITY 2

HYBRID
TECHNOLOGY
COURSES

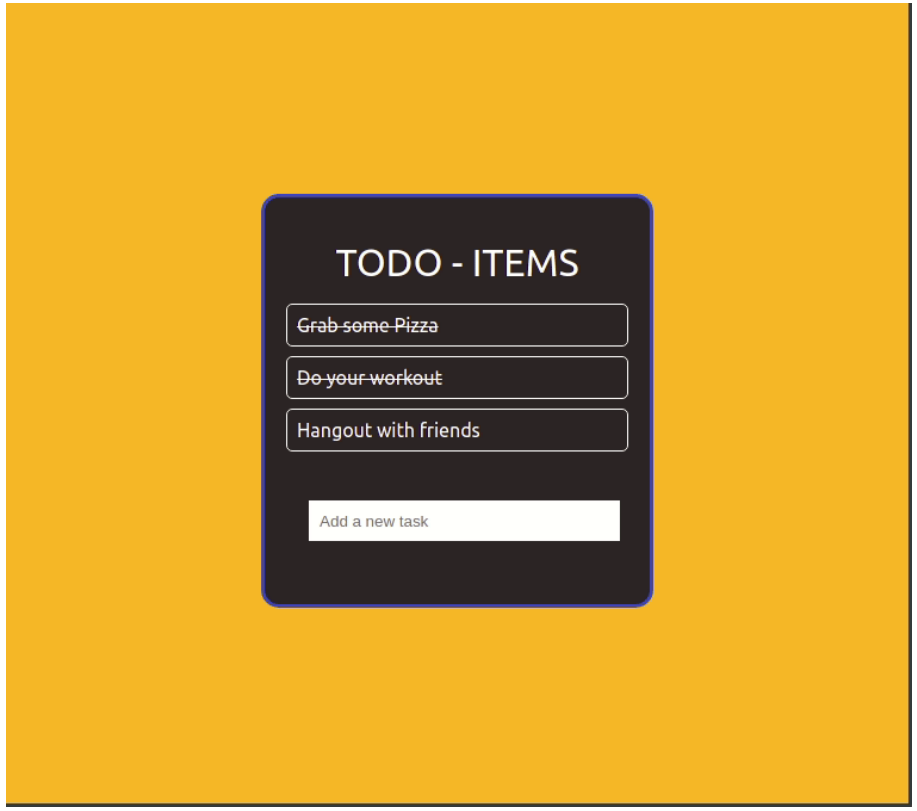
TRAINOSYS

Activity 2

- ❑ npm create vite@latest / npm create vite@4.1.0
- ❑ Project name: rts-d2-act-two

- ❑ Create an input form that accepts a string when an "Add task" button is clicked
- ❑ Create a <Task /> component that accepts task's name as props and display the initial Tasks array
- ❑ Conditionally render the component if the Tasks array is empty - "No Tasks for today <insert a beautiful quote here>"

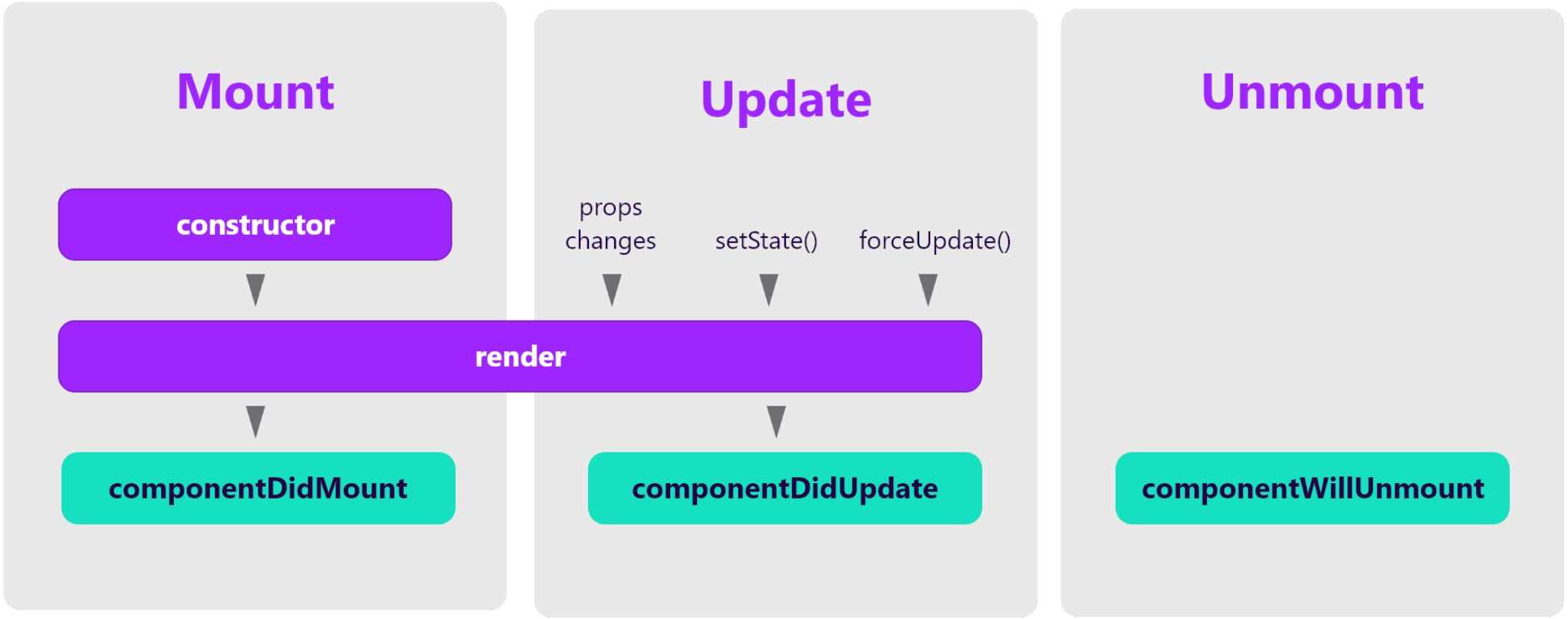
Activity 2 – example expected output



TRAINOSYS

LIFECYCLE METHODS

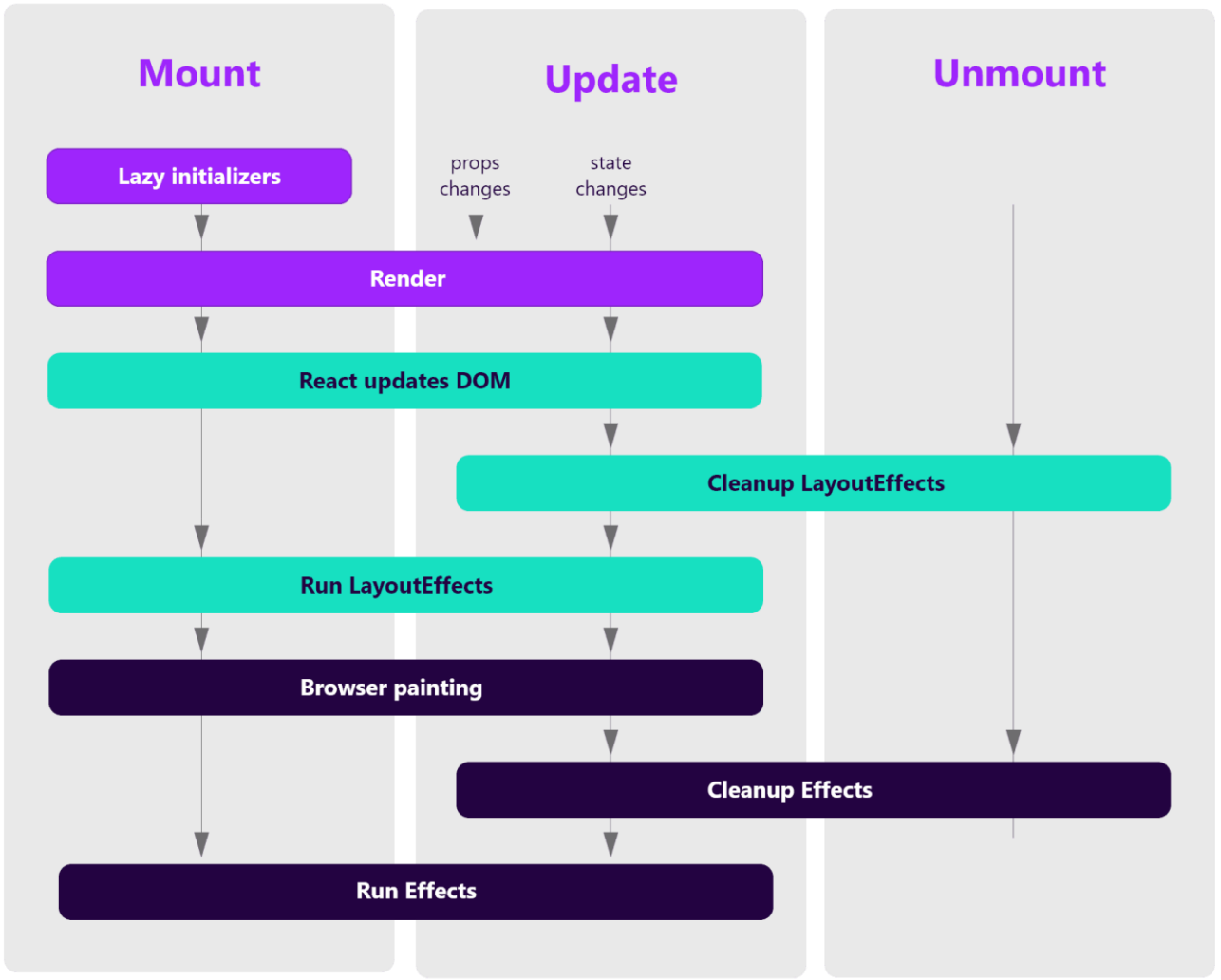
HYBRID
TECHNOLOGY
COURSES


Mount
constructor
render
componentDidMount
Update
props changes
setState()
forceUpdate()
render
componentDidUpdate
Unmount
componentWillUnmount

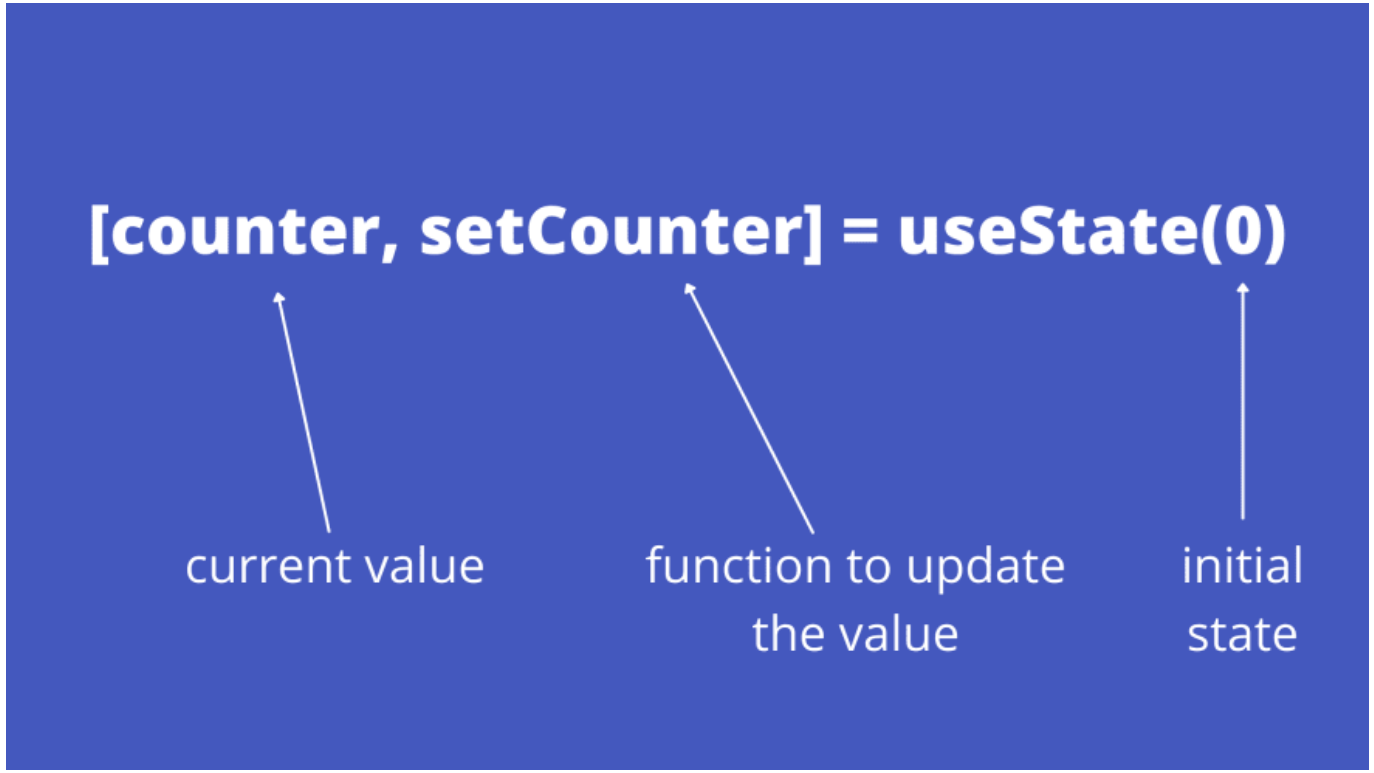
TRAINOSYS

REACT HOOKS

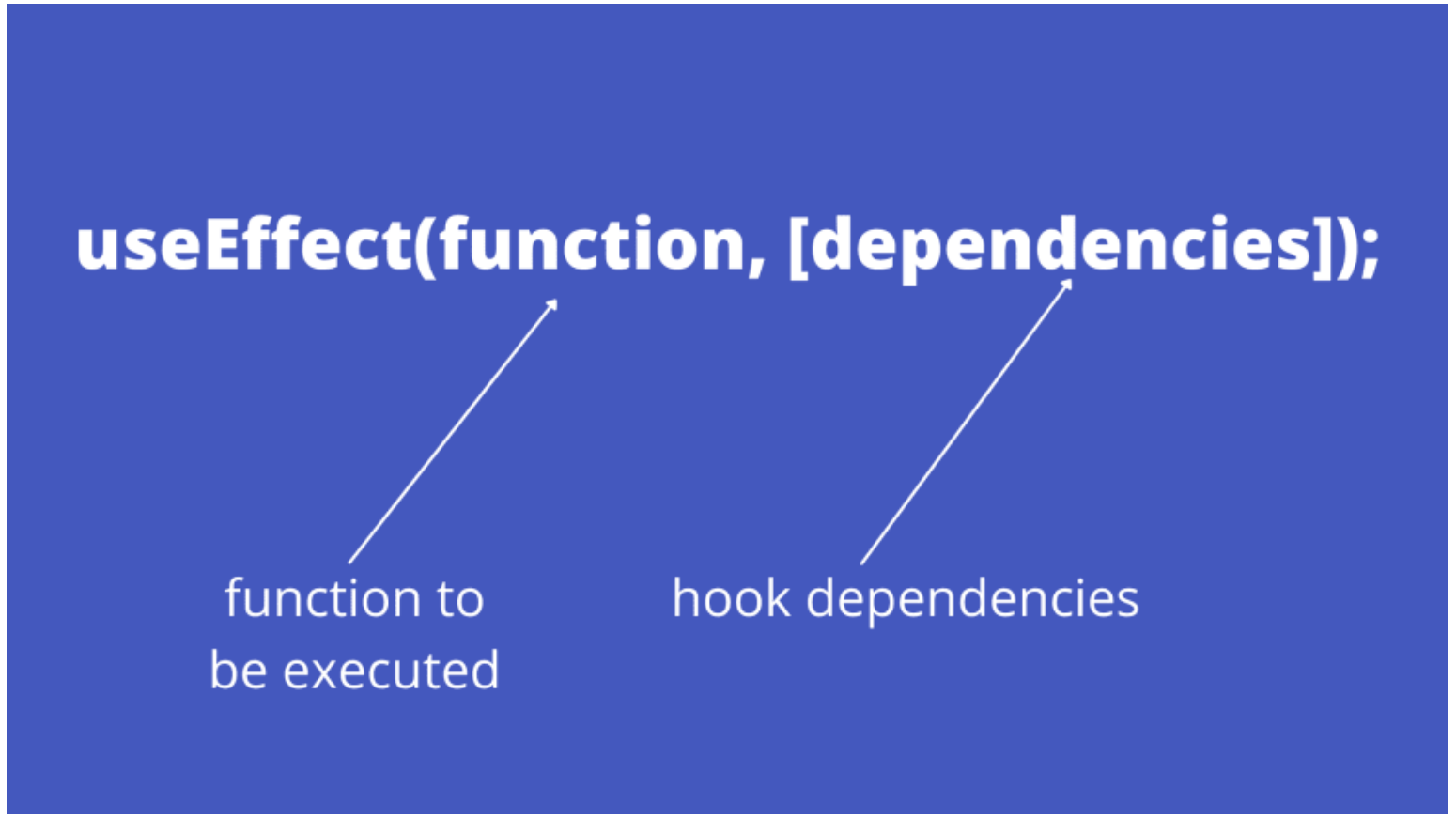
HYBRID
TECHNOLOGY
COURSES



useState() hook



useEffect() hook



Mount

```
useEffect(() => {

}, [])
```

Set "[]" empty dependency array to run an effect only on "mount"

Update

```
useEffect(() => {

})
```

Do not pass a dependency array at all to run an effect on each component update

```
useEffect(() => {

}, [dependencies])
```

Set dependencies array to run an effect only if any dependency change

Unmount

```
useEffect(() => {
  return () => {

  }
}, [])
```

Set return function to run on component unmount lifecycle.

TRAINOSYS

TRAINOSYS

CODE ALONG

HYBRID
TECHNOLOGY
COURSES

TRAINOSYS

ACTIVITY

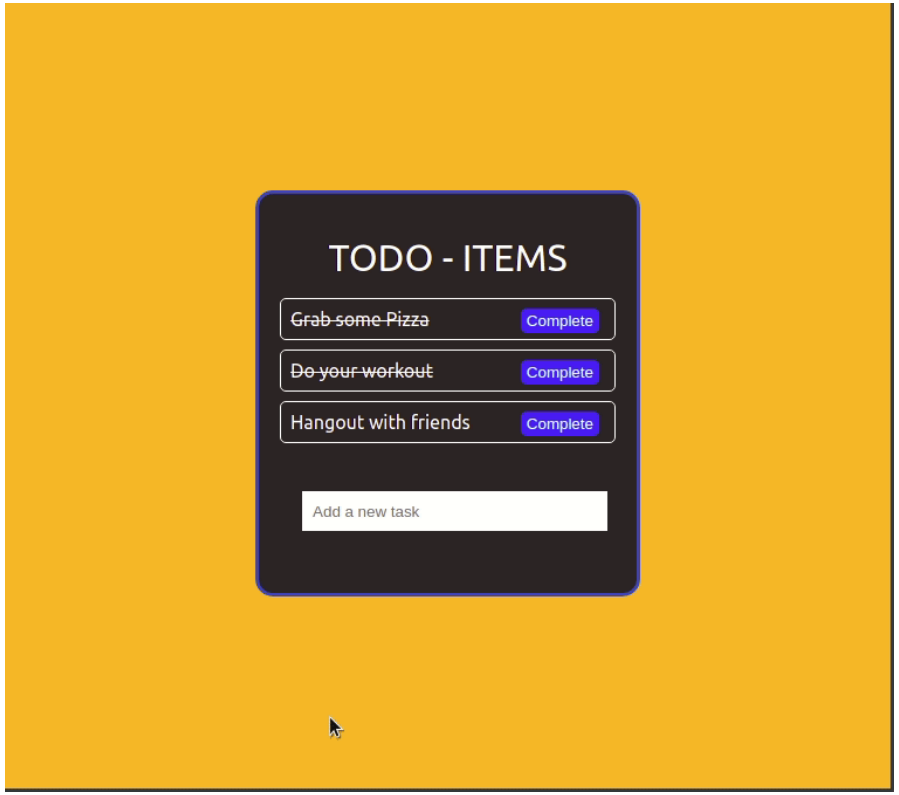
HYBRID
TECHNOLOGY
COURSES

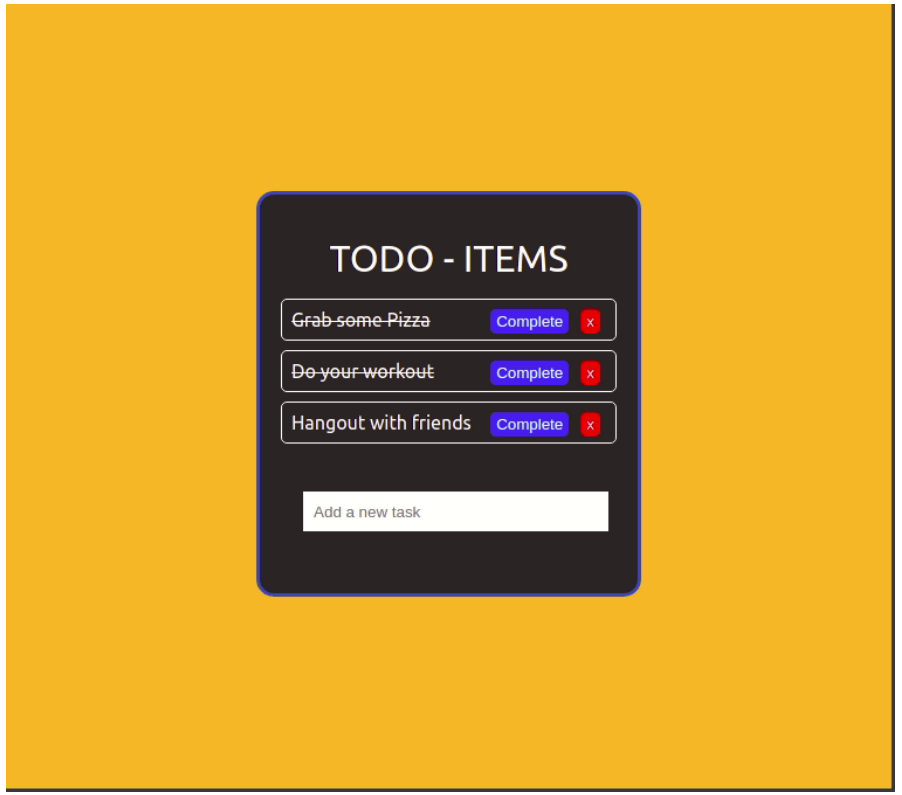
TRAINOSYS

Activity 2 continuation

- ❑ Continue the Activity 2 and add a done, edit and delete button / icon in the <Task /> component
- ❑ Create another component <AddTask />
- ❑ Migrate the input form in the <AddTask /> component and make a function adding a task in the Tasks array
- ❑ Make the done button / icon work by creating a *doneTask* function
- ❑ Make the edit button / icon work by creating an *editTask* function (you can use a modal for the edit form)
- ❑ Make the delete button / icon work by creating a *deleteTask* function

Activity 2 cont – example expected output





TRAINOSYS
Training the Future Today

Reach Us!

Visit Us

12th/F The Trade & Financial Tower Unit 1206
32nd Street & 7th Avenue Bonifacio Global City,
Taguig 1634 Philippines

Email Us

inquiry@trainosys.com

Browse Our Website

www.trainosys.com

TRAINOSYS

Training the Future Today