

# Day - 9 TASK React, Transpiler, FileStructure 04.12.2025

## 1) What is React?

React is a **JavaScript UI library** used for building fast and interactive web interfaces using reusable components.

### ◆ Key Points

- **Component-based** ⇒ UI divided into reusable blocks
- **Unidirectional Data Flow** ⇒ data flows parent → child
- **Virtual DOM** ⇒ efficient rendering
- **Reconciliation** ⇒ only modified parts update
- **SPA (Single Page Application)** ⇒ no full-page reload

### ◆ Why React is Fast?

1. React does **not update entire DOM**
2. It updates **only changed elements**
3. Uses **Virtual DOM + Diff Algorithm**

### ◆ Why React is Popular?

- Very high **performance**
- **Code reusability**
- Strong **developer community**
- Big ecosystem (**Hooks, Router, Redux** etc.)

---

## What is Diff Algorithm?

The **Diff Algorithm** is a technique used in React to:

- Compare the **old Virtual DOM** with the **new Virtual DOM**
- Detect the **minimum amount of changes**
- Update only the **changed UI elements**, not the whole DOM

### ◆ Why Diff Algorithm is Needed?

Updating the **Real DOM** directly is slow.  
Updating the **Virtual DOM** is fast.

So React:

1. Creates a **new Virtual DOM**
2. **Compares** it with the previous Virtual DOM
3. Updates only **modified nodes** in Real DOM

---

## 2) What is Reconciliation?

**Reconciliation** is the process React uses to update the UI when the data changes.

### ◆ Steps of Reconciliation:

1. React generates the **Virtual DOM**
2. When state changes → **New Virtual DOM**
3. React **compares old vs new Virtual DOM**
4. Only the **difference** is applied to the Real DOM

### ◆ Reconciliation Uses:

- **Diff Algorithm**
- **Fiber Architecture**

### ◆ Why Reconciliation is Needed?

Updating Real DOM directly is expensive (slow).

Updating Virtual DOM is cheap (fast).

So React:

- **First updates Virtual DOM**
  - Then applies **only required changes** to Real DOM.
- 

## 3) What is Transpiler?

A **Transpiler** converts code from **one high-level language syntax** to **another readable syntax**.

### ◆ In React, Transpilers convert:

- **JSX → JavaScript**
- **ES6 → ES5**
- **TypeScript → JavaScript**

Transpiling is needed because the **browser cannot understand JSX or TypeScript directly**.

---

## 4) What is Compiler?

A **Compiler** converts **human-readable code** into **machine code (binary)** which the CPU can execute directly.

### ◆ Compiler Output:

- Produces **binary or assembly**
- Used by languages like **C, C++, Java, Rust**

### ◆ Note:

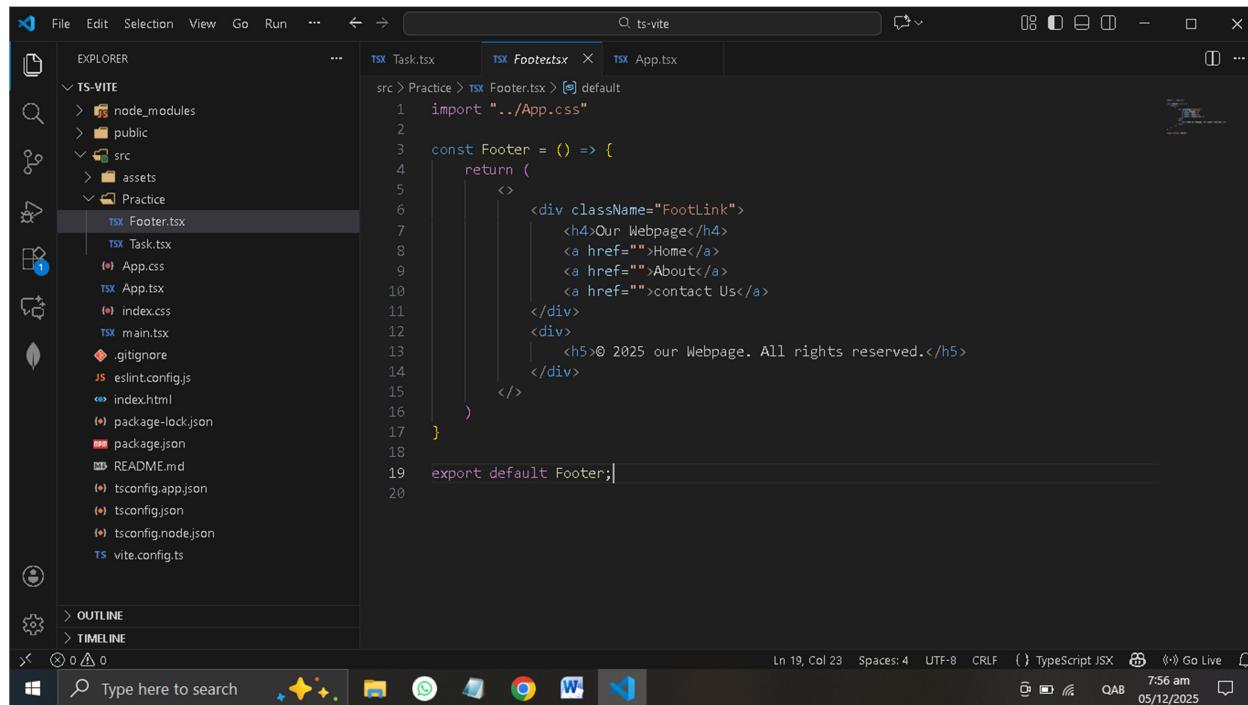
React **does not compile into machine code**.  
It uses **transpilers** (Babel, TypeScript) instead of compilers.

---

## 5) React Components & main.jsx (Deep Explanation)

### ◆ Component Rules:

1. Component name must start with a **Capital letter**
2. Must **return JSX**
3. Must be **reusable and pure**

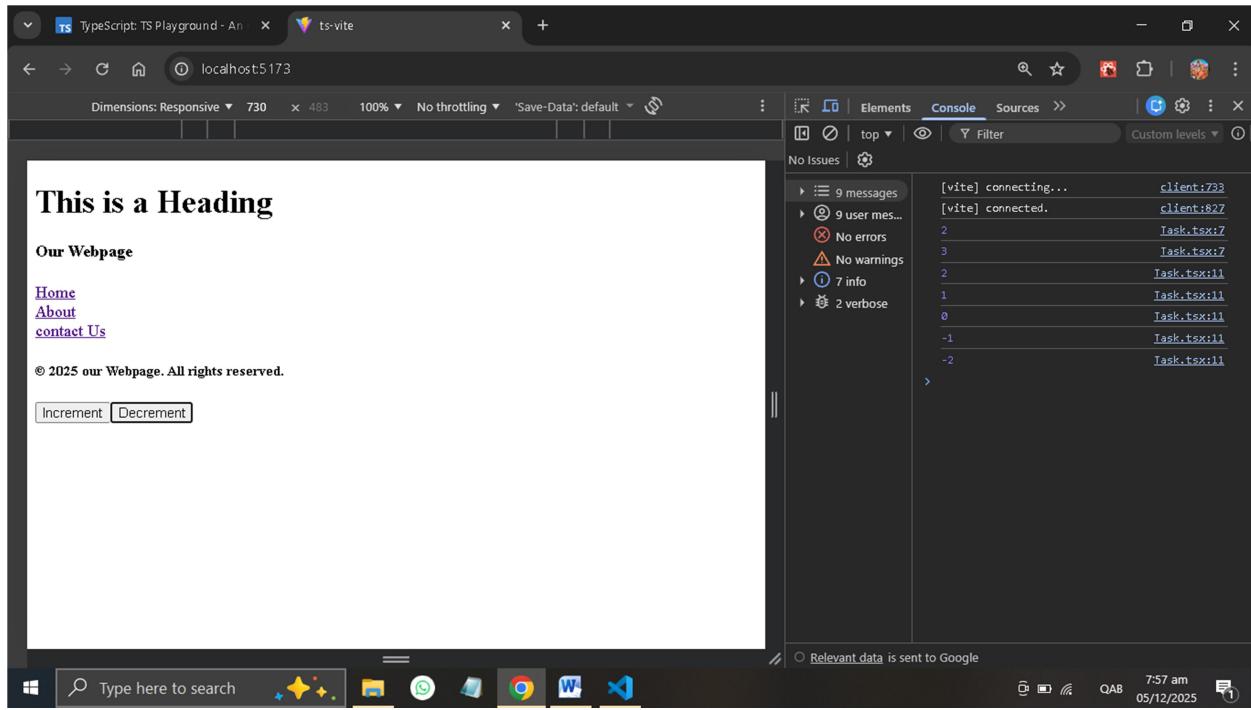


```

File Edit Selection View Go Run ... ⏪ ⏩ ts-vite
EXPLORER TSX Task.tsx TSX Footer.tsx TSX App.tsx
src > Practice > TSX Footer.tsx > [ ] default
1 import "../App.css"
2
3 const Footer = () => {
4   return (
5     <>
6       <div className="FootLink">
7         <h4>Our Webpage</h4>
8         <a href="">Home</a>
9         <a href="">About</a>
10        <a href="">Contact Us</a>
11      </div>
12      <div>
13        <h5>© 2025 our Webpage. All rights reserved.</h5>
14      </div>
15    </>
16  )
17 }
18
19 export default Footer;
20

```

The screenshot shows the VS Code interface with the Footer.tsx file open in the center editor tab. The left sidebar displays the project structure in the Explorer panel, showing files like Task.tsx, App.css, and App.tsx. The bottom status bar indicates the code is at line 19, column 23, in TypeScript JSX mode, with the date and time as 05/12/2025 at 7:56 am.



## This is a Heading

### Our Webpage

[Home](#)  
[About](#)  
[contact Us](#)

© 2025 our Webpage. All rights reserved.



Type here to search



QAB 7:57 am

05/12/2025