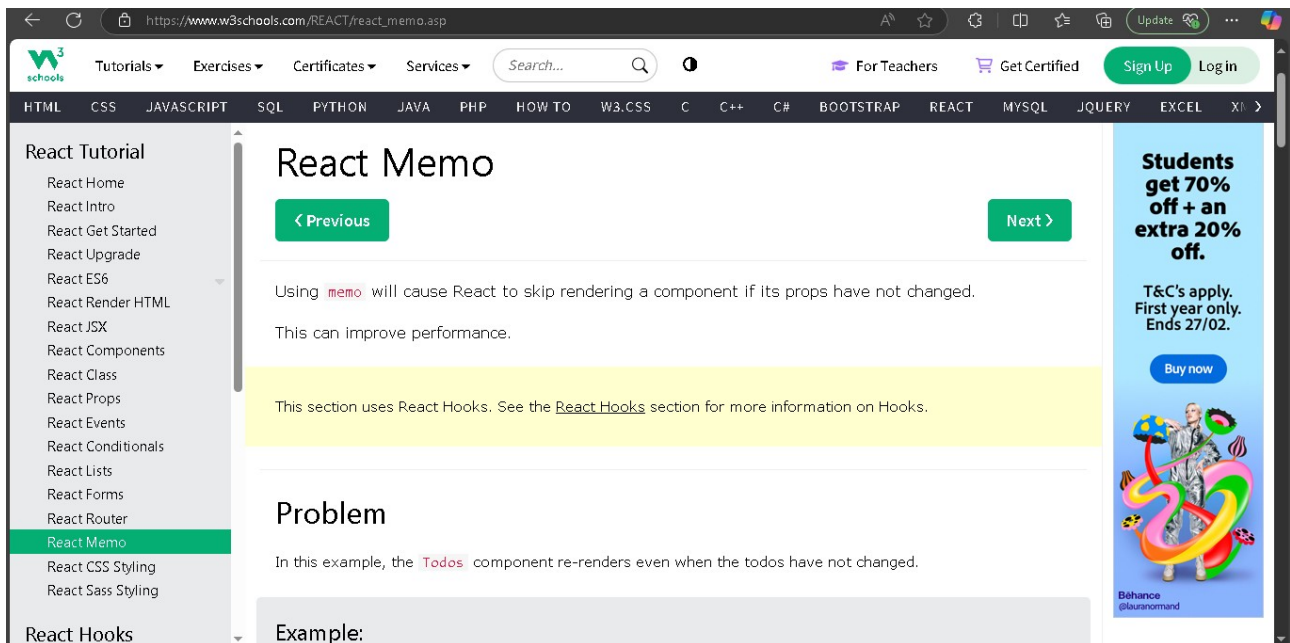


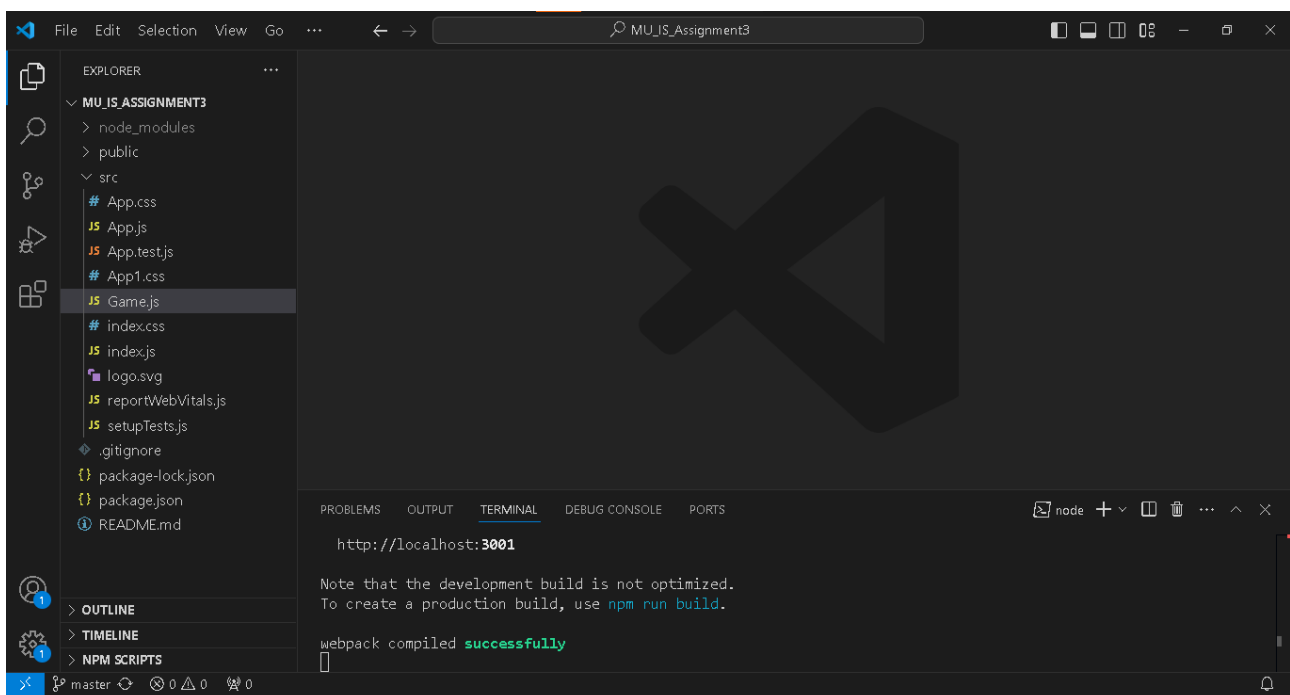
Gursimran Singh Basra  
23251097

## 1. Self study



The screenshot shows the 'React Memo' tutorial page on w3schools.com. The page title is 'React Memo'. It features a sidebar on the left with a list of topics, including 'React Memo' which is currently selected. The main content area explains that using `memo` will cause React to skip rendering a component if its props have not changed, which can improve performance. A yellow highlighted section states: 'This section uses React Hooks. See the [React Hooks](#) section for more information on Hooks.' Below this, a 'Problem' section describes an example where the `Todos` component re-renders even when the todos have not changed. An 'Example:' section is partially visible at the bottom. On the right side, there is a promotional banner for a 70% discount on a course, with a 'Buy now' button and a small image of a person.

## 2. Playing a bit with React JS



The screenshot shows a Visual Studio Code editor window with a project named 'MU\_JS\_Assignment3'. The Explorer sidebar on the left shows the file structure, including 'node\_modules', 'public', and 'src' folders. The 'src' folder contains files like 'App.css', 'App.js', 'App.test.js', 'App1.css', 'Game.js', 'index.css', 'index.js', 'logo.svg', 'reportWebVitals.js', 'setupTests.js', '.gitignore', 'package-lock.json', 'package.json', and 'README.md'. The main editor area displays a large, faint 'X' watermark. At the bottom, the Terminal panel shows the output of a command, displaying 'http://localhost:3001' and a message: 'Note that the development build is not optimized. To create a production build, use npm run build.' Below this, it says 'webpack compiled successfully'.

←↻🔒https://react.dev/learn

🔍 Search

LearnReferenceCommunityBlog

⚙️🔍🔄

GET STARTED

Quick Start

Tutorial: Tic-Tac-Toe

Thinking in React

Installation

Setup

LEARN REACT

Describing the UI

Adding Interactivity

Managing State

Escape Hatches

Is this page useful?👍👎

```
export default function MyApp() {
  return (
    <div>
      <h1>Welcome to my app</h1>
      <MyButton />
    </div>
  );
}
```

Notice that `<MyButton />` starts with a capital letter. That's how you know it's a React component. React component names must always start with a capital letter, while HTML tags must be lowercase.

Have a look at the result:

App.js

DownloadResetFork

```
1 function MyButton() {
2   return (
3     <button>
4       I'm a button
5     </button>
6   );
}
```

Welcome to my app

I'm a button

←↻🔒https://react.dev/learn/tutorial-tic-tac-toe

🔍 Search

LearnReferenceCommunityBlog

⚙️🔍🔄

GET STARTED

Quick Start

Tutorial: Tic-Tac-Toe

Thinking in React

Installation

Setup

LEARN REACT

Describing the UI

Adding Interactivity

Managing State

Escape Hatches

Is this page useful?👍👎

LEARN REACT > QUICK START >

# Tutorial: Tic-Tac-Toe

You will build a small tic-tac-toe game during this tutorial. This tutorial does not assume any existing React knowledge. The techniques you'll learn in the tutorial are fundamental to building any React app, and fully understanding it will give you a deep understanding of React.

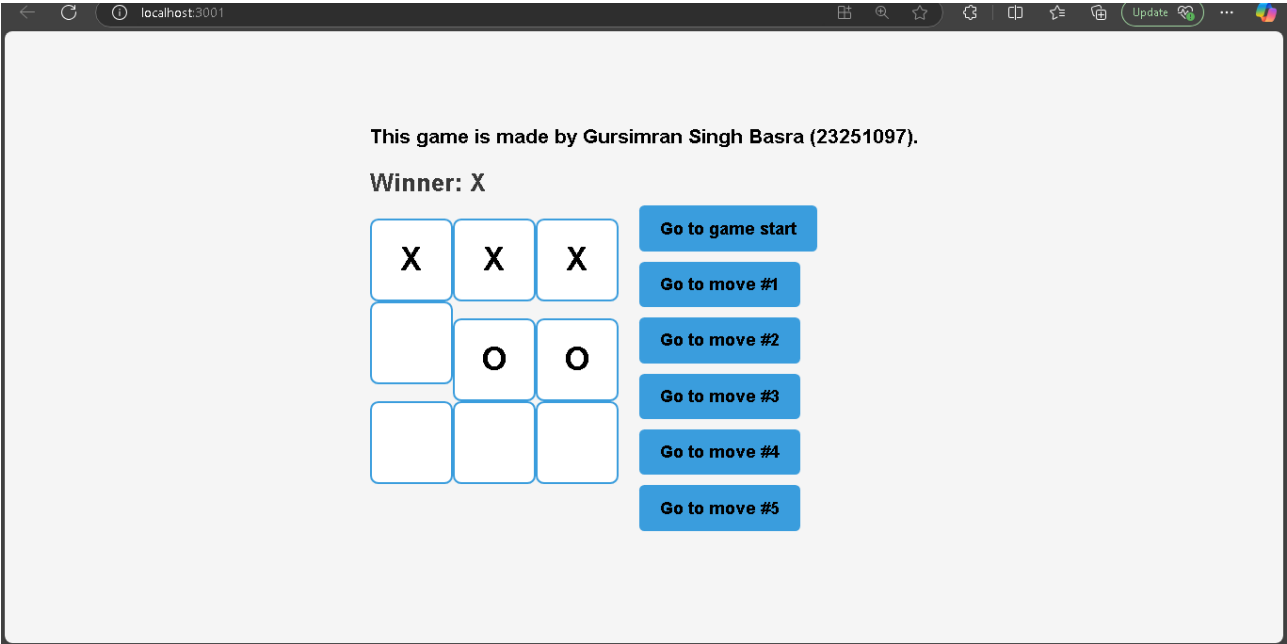
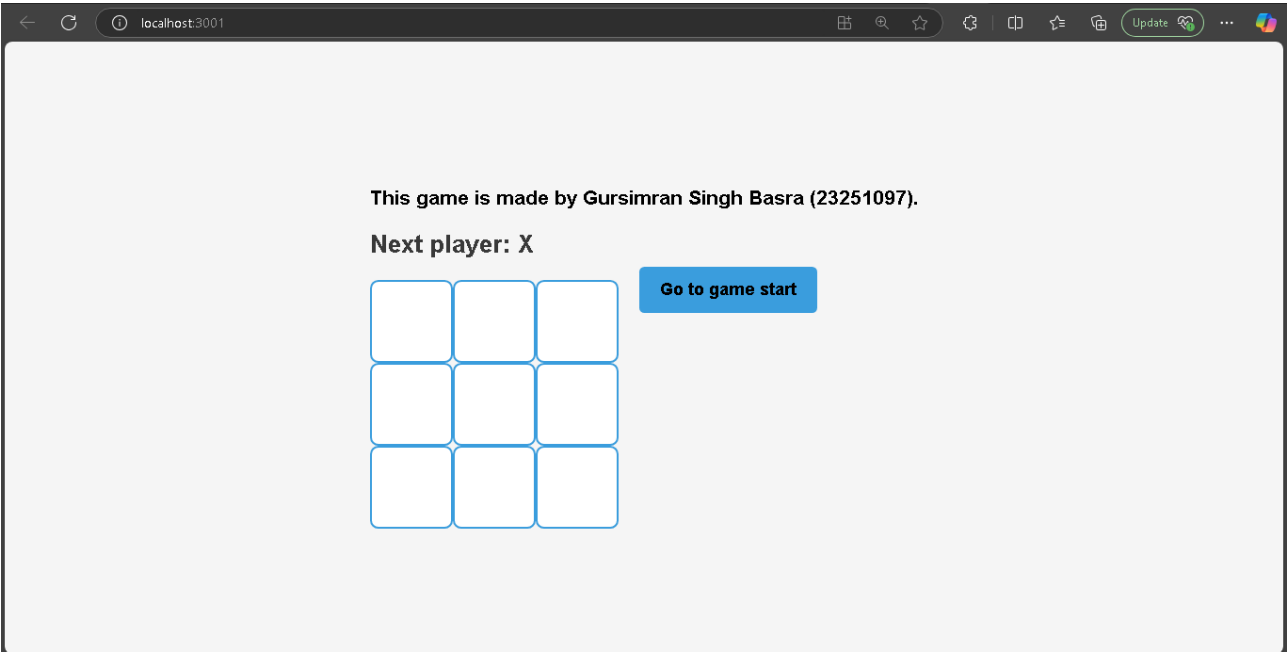
Note

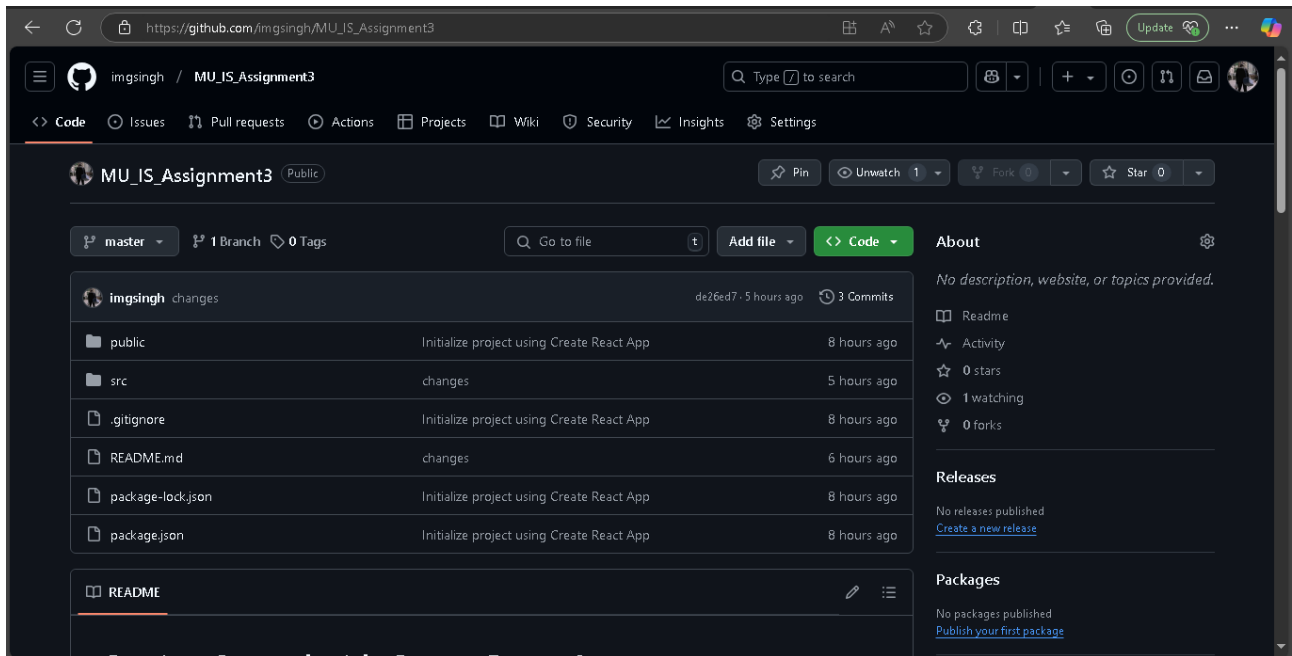
This tutorial is designed for people who prefer to **learn by doing** and want to quickly try making something tangible. If you prefer learning each concept step by step, start with [Describing the UI](#).

The tutorial is divided into several sections:

- [Setup for the tutorial](#) will give you a **starting point** to follow the tutorial.
- [Overview](#) will teach you **the fundamentals** of React: components, props, and state.

3. React JS Tutorial ( Tic-Tac-Toe)





Link to Github: [imgsingh/MU\\_IS\\_Assignment3](https://github.com/imgsingh/MU_IS_Assignment3)