# React Basics

## As a beginner in React, it’s important to cover the fundamental concepts and get hands-on experience. Here’s a structured 5-hour study table to help you get started with the basics and build a strong foundation in React.

## ### Study Table: 5 Hours

## #### Hour 1: Introduction to React

## 1. \*\*React Overview (15 mins)\*\*

## - What is React?

## - Benefits of using React

## - Setting up the development environment

## 2. \*\*Creating a React App (15 mins)\*\*

## - Using Create React App to set up a new project

## - Project structure and files

## 3. \*\*JSX Syntax (30 mins)\*\*

## - What is JSX?

## - Embedding expressions in JSX

## - Differences between JSX and HTML

## \*\*Resources:\*\*

## - React Official Documentation: [React Getting Started](https://reactjs.org/docs/getting-started.html)

## - Create React App: [Create React App](https://reactjs.org/docs/create-a-new-react-app.html#create-react-app)

## #### Hour 2: Components and Props

## 1. \*\*Understanding Components (30 mins)\*\*

## - Functional components

## - Class components

## - Component hierarchy

## 2. \*\*Props (30 mins)\*\*

## - Passing props to components

## - Using props within components

## - Default props and prop types

## \*\*Resources:\*\*

## - React Official Documentation: [Introducing JSX](https://reactjs.org/docs/introducing-jsx.html)

## - React Official Documentation: [Components and Props](https://reactjs.org/docs/components-and-props.html)

## #### Hour 3: State and Lifecycle

## 1. \*\*State in Functional Components (30 mins)\*\*

## - useState hook

## - Updating state

## - Managing multiple state variables

## 2. \*\*Lifecycle Methods in Class Components (30 mins)\*\*

## - Constructor

## - componentDidMount

## - componentDidUpdate

## - componentWillUnmount

## \*\*Resources:\*\*

## - React Official Documentation: [State and Lifecycle](https://reactjs.org/docs/state-and-lifecycle.html)

## - React Official Documentation: [Using the State Hook](https://reactjs.org/docs/hooks-state.html)

## #### Hour 4: Event Handling and Conditional Rendering

## 1. \*\*Handling Events (30 mins)\*\*

## - Adding event handlers

## - Handling form events

## - Passing arguments to event handlers

## 2. \*\*Conditional Rendering (30 mins)\*\*

## - if-else statements

## - Ternary operators

## - Logical && operator

## \*\*Resources:\*\*

## - React Official Documentation: [Handling Events](https://reactjs.org/docs/handling-events.html)

## - React Official Documentation: [Conditional Rendering](https://reactjs.org/docs/conditional-rendering.html)

## #### Hour 5: Lists and Keys, Forms

## 1. \*\*Rendering Lists (30 mins)\*\*

## - Using map() to render lists

## - Understanding keys in React

## - Best practices for using keys

## 2. \*\*Controlled Components and Forms (30 mins)\*\*

## - Handling form inputs

## - Controlled vs uncontrolled components

## - Managing form state

## \*\*Resources:\*\*

## - React Official Documentation: [Lists and Keys](https://reactjs.org/docs/lists-and-keys.html)

## - React Official Documentation: [Forms](https://reactjs.org/docs/forms.html)

## ### Practice Project

## As a final step, spend some time building a simple React application that incorporates the concepts you’ve learned. Here’s a suggestion for a small project:

## \*\*To-Do List App\*\*

## - Create a To-Do List application where users can add, remove, and mark tasks as completed.

## - Use functional components and hooks (useState) to manage the state.

## - Implement event handling for form submissions and list item interactions.

## - Practice conditional rendering to display different UI based on the state of tasks.

## \*\*Additional Resources:\*\*

## - React Official Documentation: [Main Concepts](https://reactjs.org/docs/hello-world.html)

## - FreeCodeCamp: [React Basics](https://www.freecodecamp.org/learn/front-end-libraries/react/)

## This study table provides a comprehensive introduction to React, covering the essential concepts and providing practical exercises to reinforce your learning. By the end of these 5 hours, you should have a solid foundation to start working on more complex React projects

## SHORTCUTS:

Command+shift+r 🡪 to search the snippet

GO to RAFCE for the standard Create a react functional component to Module ES6

React Hooks

Code: import React from "react";

import { useState } from "react";

const Content = () => {

const [name,setName]= useState("Grow");

function handleNameChange() {

const names=["Earn","Learn","Understand","Grow"];

const int = Math.floor(Math.random()\*4);

setName ( names[int])

}

/\*When you use handleNameChange() in the onClick attribute, it calls the function immediately during the render, causing state to update and triggering a re-render. This results in an infinite loop.

• By passing the function reference handleNameChange without parentheses, React will call this function only when the button is clicked.\*/

return (

<main>

<p>Let's {name}</p>

<button onClick={handleNameChange}>Challenge Accepted</button>

</main>

)

}

export default Content

`