

React: Components and Props

Phase 2 // Week 1, Day 1

Today's Objectives

Today, we'll answer the following questions:

1. What is **React**?
2. What are **components**, **JSX**, and **props**?
3. What do we (not) like about React?

What is React?

- **React** is a JavaScript library for building websites and apps.
- Its manifesto, if it had one, might go something like this:
 1. Every application can be reduced to a combination of one or more **components**.
 2. **Components** should be reusable.
 3. **Components** should only re-render when they need to.

How do we write a component?

- **Components** are just like functions: we can call them anywhere, pass them inputs, define things inside of them, and return things from them.

```
function Dish({ name }) {  
  return (  
    <h1>{name}</h1>  
  )  
}
```

```
function Restaurant() {  
  const dishNames = ["pasta", "bread"];  
  return (<>  
    <Dish name="sushi" />  
    <Dish name={dishNames[0]} />  
  </>)  
}
```


How do we write a component?

- **Components** are just like functions: we can render them in other components, pass them props, define things inside of them, and return **JSX** from them.

```
function Dish({ name }) {  
  return (  
    <h1>{name}</h1>  
  )  
}
```

```
function Restaurant() {  
  const dishNames = ["pasta", "bread"];  
  return (<>  
    <Dish name="sushi" />  
    <Dish name={dishNames[0]} />  
  </>)  
}
```

How do we write a component?

- **Components** are just like functions: we can render them in other components, pass them props, define things inside of them, and return **JSX** from them.

```
function ChildComponentName ({ propNameA, propNameB }) {  
  /* ..things.. */  
  return (<htmlTag>{propNameA} | {propNameB}</htmlTag>)  
}
```

```
function ParentComponentName ({ propNameC, propNameD, ... }) {  
  /* ..things.. */  
  return (<ChildComponentName propNameA={propValueA} />)  
}
```


Let's try it



How do we feel about React?

Thanks!