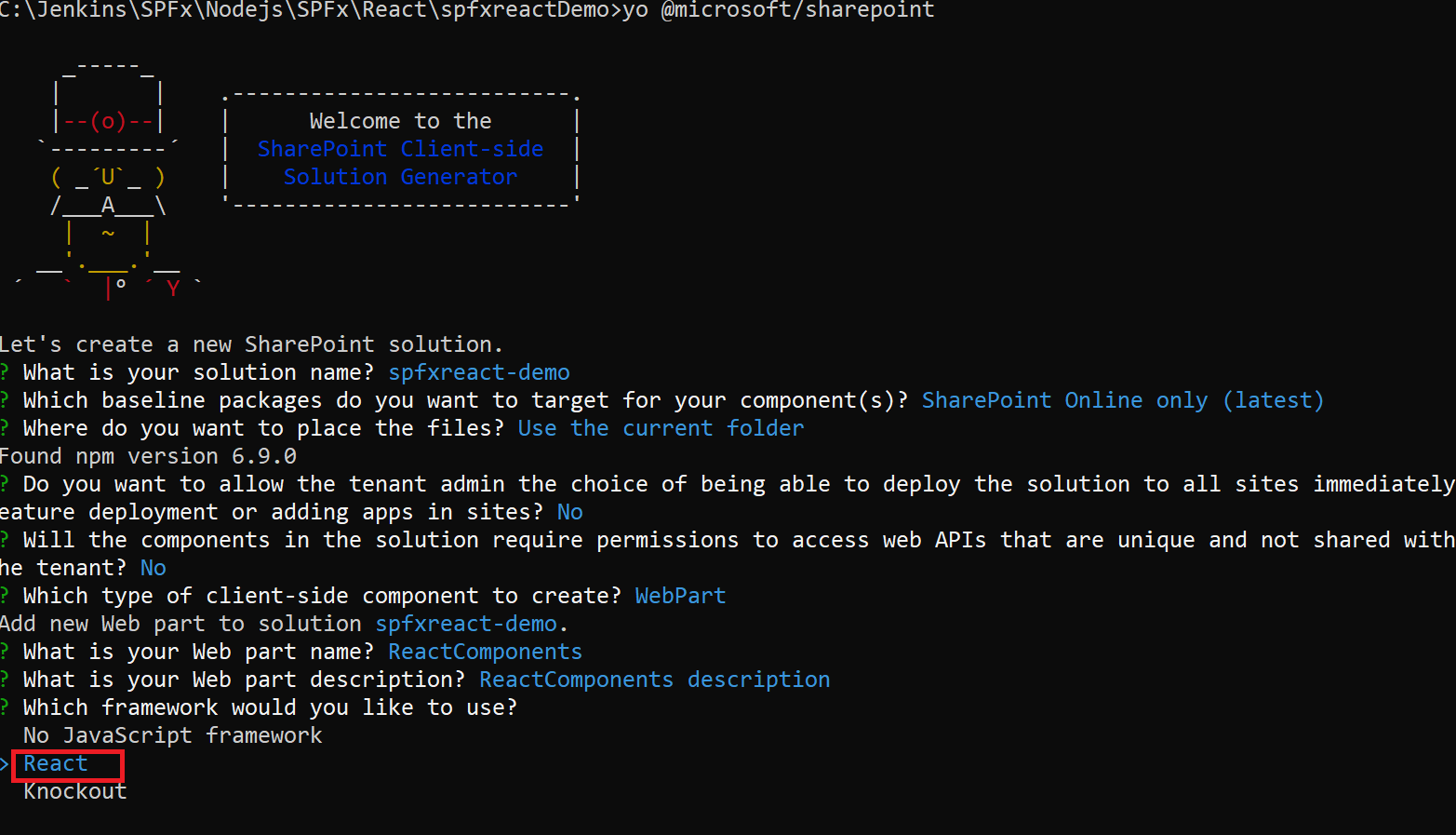
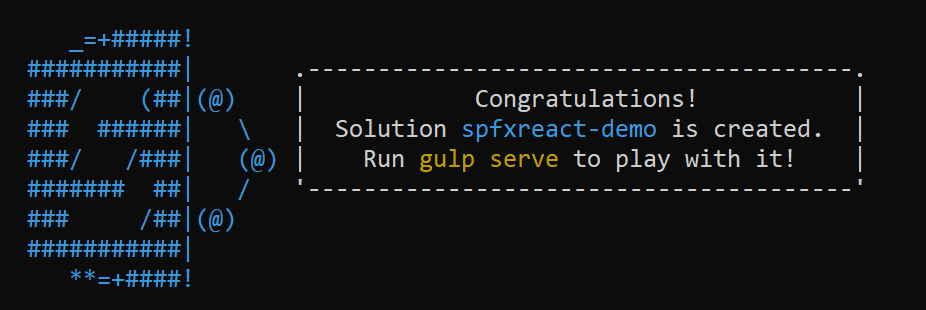
**Create SPFx webpart using React**

**Example 1 : Create a react webpart and add components**

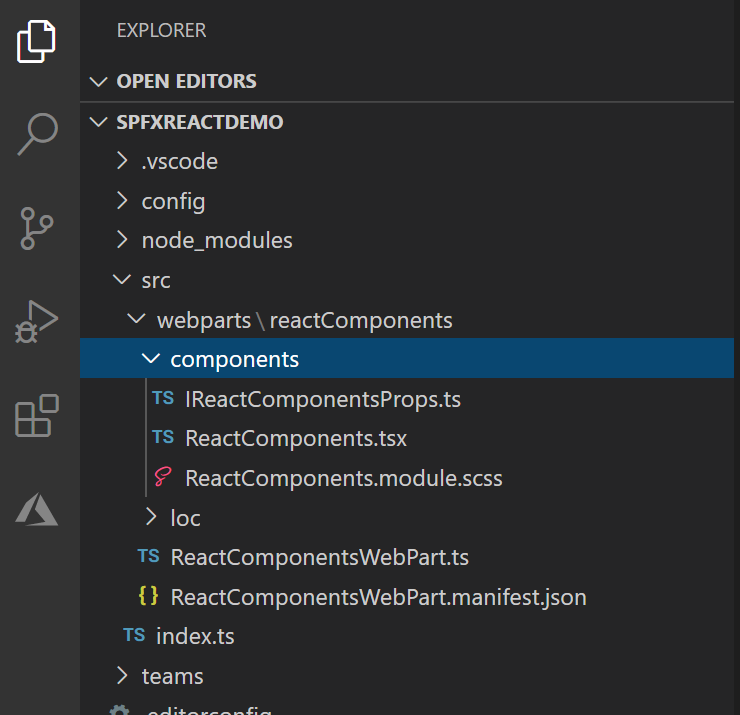
1. Create a folder “spfxreactDemo” under your favourite folder
2. Move to the folder “spfxreactDemo”
3. Create a webpart same as previous example, and select the framework as React



1. Enter to create a Webpart



1. Open the solution in Visual Studio Code
2. Type **Code .**



1. To know the starting point
2. Open “ReatComponentWebpart.ts” file
3. public render(): void {
4. const element: React.ReactElement<IReactComponentsProps> = React.createElement(
5. ReactComponents,
6. {
7. description: this.properties.description
8. }
9. );
10. ReactDom.render(element, this.domElement);
11. }

* In this example going to add some components
  + Welcome
  + MyCart
  + Products
  + Orders
* For this create Welcome.tsx file under component folder
  + Add below code

import \* as React from 'react';

export default class Welcome extends React.Component

{

  public render()

  {

    return <div>Welcome to JPOWER4 Shopping Cart! </div>;

  }

}

* Same as create
  + Products.tsx file

1. import \* as React from 'react';
2. export default class Products extends React.Component
3. {
4. public render()
5. {
6. return <div>Welcome to Products! </div>;
7. }
8. }
   * Orders.tsx file
9. import \* as React from 'react';
10. export default class Orders extends React.Component
11. {
12. public render()
13. {
14. return <div>Welcome to Orders! </div>;
15. }
16. }
    * MyCart.tsx file
17. import \* as React from 'react';
18. export default class MyCart extends React.Component
19. {
20. public render()
21. {
22. return <div>Welcome to MyCart! </div>;
23. }
24. }

* Import the all classes in ReactComponents.tsx file

1. import Welcome from './Welcome';
2. import Products from './Products';
3. import Orders from './Orders';
4. import MyCart from './MyCart';

* then remove the default welcome messages and add the component tags

export default class ReactComponents extends React.Component<IReactComponentsProps, {}> {

  public render(): React.ReactElement<IReactComponentsProps> {

    return (

      <div className={ styles.reactComponents }>

        <div className={ styles.container }>

          <div className={ styles.row }>

            <div className={ styles.column }>

              <Welcome/>

              <Products/>

              <Orders/>

              <MyCart/>

            </div>

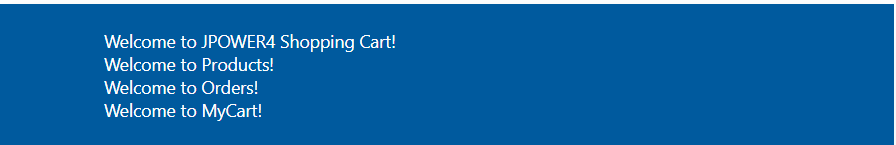
          </div>

        </div>

      </div>

    );

  }



**Example 2 : add components with props (dynamic data)**

Follow the same webpart example 1

1. We can bring dynamic parameters to load the component
2. Open welcome.tsx file
3. Add a new interface with name parameter

export interface IWelcomeProps{

  name:string;

}

1. Modify the class with IWelcomeProps parameter and also add return type in render method with the reactElement

export default class Welcome extends React.Component<IWelcomeProps,{}>

{

  public render():React.ReactElement<IWelcomeProps>  {

    return <div>Welcome to {this.props.name}</div>;

  }

}

1. Now you can able to call the Welcome component from any other components with parameter name.
2. Open Orders.tsx file
   1. Import welcome component

import Welcome from './Welcome';

* 1. Update the return div with custom parameter value

return <div><Welcome name="Orders dynamic data"/></div>;

and

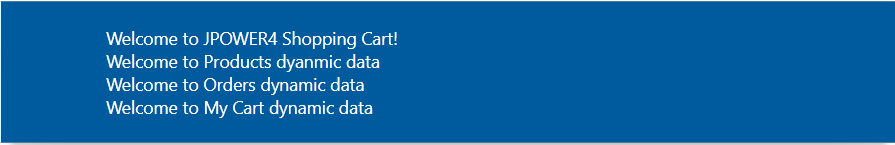
return <div><Welcome name="Products dyanmic data"/></div>;

1. Same as follow for Products.tsx, MyCart.tsx

return <div><Welcome name="My Cart dynamic data"/></div>;

1. Then update the <Welcome>/> tag in ReactComponent.tsx file

<Welcome name="JPOWER4 Shopping Cart!"/>



**Example 3 : add components with state (update specific data dynamically)**

1. Add State functionality
2. Open Products.tsx file and update the render method
   1. i.e add Electronics tag

public render()

  {

    return <div>

      <Welcome name="Products dyanmic data"/>

      <Electronics description="Product Description" />

    </div>;

  }

1. Create a new file Electronics.tsx
2. Paste the below code

import \* as React from 'react';

import { render } from 'react-dom';

import {IReactComponentsProps} from './IReactComponentsProps';

export interface IItemsProps{

  title:string;

  year:string;

  price:number;

}

export default class Electronics extends React.Component<IReactComponentsProps,IItemsProps, {}>

{

  constructor(props: IReactComponentsProps){

    super(props);

    let title = "Laptop";

    let year = "2020";

    let price = 700;

    this.state = { title, year, price };

  }

  private applydiscount():void

  {

    this.setState({

      price:250

    });

  }

  public render():React.ReactElement<IItemsProps>

  {

    const element = (

    <React.Fragment>

      <div>

      <ul>

            <li>

              <div>

                  <b>Book Name : {this.state.title}</b>

                  <p>Year : {this.state.year}</p>

                  <p>Price : {this.state.price}</p>

              </div>

            </li>

          </ul>

          <button onClick={() =>this.applydiscount()}>Apply discount</button>

      </div>

    </React.Fragment>

    );

    return element;

  }

}

Then you can able to set the new value dynamically.