**1.Differences between Observables and Promise.**

1. Declaration Methodology.
2. Observables are part of RXJS but Promise is a default part of JavaScript.
3. Observables can produce multiple result set but promise can produce only one set of record.
4. Observables are lazy loading with subscribe but promise is eager loading.
5. Observables can be cancelled with unsubscribe but promise can’t be cancelled.

**Declaration and calling a promise and an observer:**

  ngOnInit(){

    //Declaration of Promise

    const promiser = new Promise(resolve=>{

      setTimeout(()=>{resolve('promise working')},1000)

    })

    //calling promise

    promiser.then((result)=>{console.log(result)})

    //Declaration of Observables - need RXJS

    const observer = new Observable(sub =>{

      setTimeout(()=>{sub.next('observer working')},1000)

    })

    //calling observer

    observer.subscribe((result)=>{console.log(result)})

  }

**Multiple output:**

  ngOnInit(){

    //Declaration of Promise

    const promiser = new Promise(resolve=>{

      setTimeout(()=>{resolve('promise working')},1000)

      setTimeout(()=>{resolve('promise working')},1000)

    })

    //calling promise

    promiser.then((result)=>{console.log(result)})

    //Declaration of Observables - need RXJS

    const observer = new Observable(sub =>{

      setTimeout(()=>{sub.next('observer working')},1000)

      setTimeout(()=>{sub.next('observer working')},1000)

      setTimeout(()=>{sub.next('observer working')},1000)

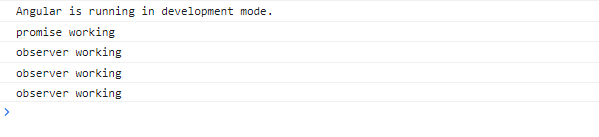
    })

    //calling observer

    observer.subscribe((result)=>{console.log(result)})

  }

Output:



**Promise are eager loading and observable are lazy loading. Even if you not use “then” still promise will load. Observable will not load until subscribed.**

ngOnInit(){

    //Declaration of Promise

    const promiser = new Promise(resolve=>{

      console.log('Promise...')

      setTimeout(()=>{resolve('promise working')},1000)

    })

    //calling promise

   // promiser.then((result)=>{console.log(result)})

    //Declaration of Observables - need RXJS

    const observer = new Observable(sub =>{

      console.log('Observable...')

      setTimeout(()=>{sub.next('observer working')},1000)

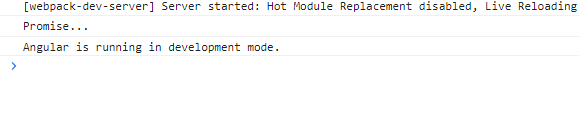
    })

    //calling observer

   // observer.subscribe((result)=>{console.log(result)})

  }

Output:



**Explain the process of adding bootstrap in your project?**

There are 2 ways to add bootstrap in project: 1. CDN 2.NPM Install

**CDN:**

To add bootstrap, copy the 4 files URL bootstrap.min.css, bootstrap.min.js, jquery.slim.min.js, & popper.min.js

Add bootstrap.min.css inside index.html 🡪 Head section.

Add bootstrap.min.js, jquery.slim.min.js, popper.min.js inside body but before closing tag. 🡪 Body

**NPM Install:**

npm I jquery , npm I popper.js , npm I bootstrap

Setup 🡪 Angular-cli.json

Styles :[“../node\_modules/…….bootstrap.min.css”]

Script: [“../node\_modules/…..jquery.min.js”,

“../node\_modules/…..popper.min.js”,

“../node\_modules/…..bootstrap.min.js”

]