Marvellous Infosystems: Angular Assignment 9

Name: Shrirang Jagdish Nikam

Enrollment No: 396AM_Shrirang

1.Create the application which contains one service named as Arithmetic.
Arithmetic service contains two methods named as Add and Sub.
Both of this Methods accepts two integers and perform addition and subtraction respectively.
We have to create one child component named as Demo under app component which uses Arithmetic service using Dependency Injection.
That methods from the service add and sub should be called from Demo component by passing some hardcoded values. And display the result of addition and subtraction inside Demo component.

Answer:

arithmetic.service.ts:

```
import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root'
})
export class ArithmeticService {

  constructor() { }

  add(num1: number, num2: number): number {
    return num1 + num2;
  }

  sub(num1: number, num2: number): number {
    return num1 - num2;

  }
}
```

app.component.html:

```
<div style="text-align:center">
  <h1>
    Welcome to {{ title }}
  </h1>
  <app-demo></app-demo>
</div>
```

demo.component.html:

```
<h1>The Arithmetic Output is</h1>
```

demo.component.ts:

```
import { Component, OnInit } from '@angular/core';
import { ArithmeticService } from '../arithmetic.service';

@Component({
  selector: 'app-demo',
  templateUrl: './demo.component.html',
  styleUrls: ['./demo.component.css']
})
export class DemoComponent implements OnInit {
  constructor(private arithmeticService: ArithmeticService) { }

    ngOnInit(): void {
    const num1 = 10;
    const num2 = 5;

    const sum = this.arithmeticService.add(num1, num2);
    const difference = this.arithmeticService.sub(num1, num2);

    console.log(`Sum: ${sum}`);
    console.log(`Difference: ${difference}`);
  }
}
```

2.Create the application which contains two services named as Number and String. Number service contains one method—named as check prime () which accepts number and check whether that number is prime or not.
String service contains one method named as Count Capital () which counts number of capital characters and return its count.
We have to create two child Component named as Child1 & Child2 under app component Child1 uses Number service & Child2 uses String service using Dependency Injection. Call both the methods from the respective components by passing some hardcoded values and display the result.

Answer:

number.service.ts:

```
import { Injectable } from '@angular/core';

@Injectable({
```

```
  providedIn: 'root'
})
export class NumberService {

  ChkPrime(num: number): boolean {
    if (num <= 1) {
      return false;
    }
    for (let i = 2; i <= Math.sqrt(num); i++) {
      if (num % i === 0) {
        return false;
      }
    }
    return true;
  }

}
```

string.service.ts:

```
import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root'
})
export class StringService {

  CountCapital(str: string): number {
    let count = 0;
    for (let i = 0; i < str.length; i++) {
      if (str[i] >= 'A' && str[i] <= 'Z') {
        count++;
      }
    }
    return count;
  }

}
```

child1.component.ts:

```
import { Component } from '@angular/core';
import { NumberService } from '../number.service';

@Component({
  selector: 'app-child1',
  template: '<p>12 Is a prime number? {{ isPrime }}</p>',
```

```
})
export class Child1Component {

  isPrime: boolean;

  constructor(private numberService: NumberService) {
    this.isPrime = this.numberService.ChkPrime(7);
  }

}
```

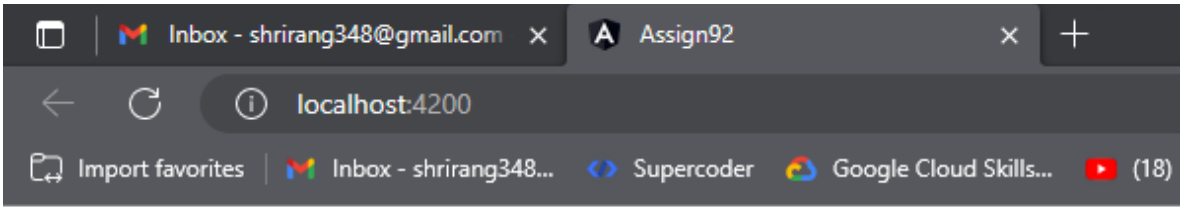
child2.component.ts:

```
import { Component } from '@angular/core';
import { StringService } from '../string.service';

@Component({
  selector: 'app-child2',
  template: '<p>Number of capital letters in "Shrirang Nikam" : {{ count }}</p>',
})
export class Child2Component {

  count: number;

  constructor(private stringService: StringService) {
    this.count = this.stringService.CountCapital('Hello World');
  }

}
```

app.component.ts:

```
import { Component } from '@angular/core';
import { Child1Component } from './child1/child1.component';
import { Child2Component } from './child2/child2.component';

@Component({
  selector: 'app-root',
  template: '<app-child1></app-child1><app-child2></app-child2>',
  providers: [Child1Component, Child2Component]
})
export class AppComponent {
}
```

Output:



12 Is a prime number? true

Number of capital letters in "Shrirang Nikam": 2

3.Create the application which contains two Services named as Number and String. Number service contains one method named Check Prime () ich accepts number and check whether that number is prime or not.
String service contains one method name count Capital () which counts number of capital characters and return its count.
We have to create one child component named as Child under app component uses Number service String service using Dependency Injection.
Call both the methods from the components by passing some hardcoded values and display the result.

Answer:

number.service.ts:

```
import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root'
})
export class NumberService {

  CheckPrime(num: number): boolean {
    if (num <= 1) {
      return false;
    }
    for (let i = 2; i <= Math.sqrt(num); i++) {
      if (num % i === 0) {
        return false;
      }
    }
    return true;
  }
```

```
}
```

string.service.ts:

```
import { Injectable } from '@angular/core';

@Injectable({
  providedIn: 'root'
})
export class StringService {

  countCapital(str: string): number {
    let count = 0;
    for (let i = 0; i < str.length; i++) {
      if (str[i] >= 'A' && str[i] <= 'Z') {
        count++;
      }
    }
    return count;
  }

}
```

child.component.ts:

```
import { Component, OnInit } from '@angular/core';
import { NumberService } from '../number.service';
import { StringService } from '../string.service';

@Component({
  selector: 'app-child',
  template: `
    <h2>Child Component</h2>
    <p>3 is a prime number? {{ isPrime }}</p>
    <p>The number of capital letters in 'Nikam Shrirang' is {{ capitalCount }}</p>
  `,
})
export class ChildComponent implements OnInit {

  isPrime: boolean | undefined;
  capitalCount: number | undefined;

  constructor(
    private numberService: NumberService,
    private stringService: StringService
  ) { }
```

```
  ngOnInit(): void {
    this.isPrime = this.numberService.CheckPrime(7);
    this.capitalCount = this.stringService.countCapital('Hello World');
  }

}
```

app.module.ts:

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { AppComponent } from './app.component';
import { ChildComponent } from './child/child.component';
import { NumberService } from './number.service';
import { StringService } from './string.service';

@NgModule({
  declarations: [
    AppComponent,
    ChildComponent
  ],
  imports: [
    BrowserModule
  ],
  providers: [
    NumberService,
    StringService
  ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

app.component.html:

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  template: `
    <h1>App Component</h1>
    <app-child></app-child>
  `,
})
```

```
export class AppComponent {
 }
```

Output: