Connecting with firebase :

Sending the post request :

Create a service with anme server.service.ts

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

import {Http} from '@angular/http';

@Injectable()

export class ServerService

{

constructor(private http : Http){ }

addRecord(data : any){

console.log(data)

return this.http.post('https://ecommerse-cf4aa.firebaseio.com/data.json',data);

}

}

**Create the form :**

<form [formGroup]="signUp">

<p>

First name : <input formControlName="first\_name" name="first\_name">

</p>

<p>

Last name : <input formControlName="last\_name" name="last\_name">

</p>

<p>

<button (click)="addRecord()">Add Record</button>

</p>

</form>

**Component file :**

import { Component, OnInit } from '@angular/core';

import {FormGroup,FormControl} from '@angular/forms';

import {ServerService} from './serve.service';

@Component({

selector: 'app-httpmodule',

templateUrl: './httpmodule.component.html',

styleUrls: ['./httpmodule.component.css'],

providers : [ServerService]

})

export class HttpmoduleComponent implements OnInit {

signUp : FormGroup;

constructor(private srve : ServerService) { }

ngOnInit() {

this.signUp = new FormGroup({

first\_name : new FormControl(null),

last\_name : new FormControl(null)

});

}

addRecord(){

console.log(this.signUp)

this.srve.addRecord({first\_name : this.signUp.value.first\_name,last\_name : this.signUp.value.last\_name}).subscribe(

(response)=>console.log(response),

(error)=>console.log(error)

);

}

}

**Adjusting request header :**

import {Headers} from ‘@angular/http’

const headers = new Headers({‘Content-Type’ : ‘application/json’});

return this.http.post('https://ecommerse-cf4aa.firebaseio.com/data.json',data,{headers : headers});

**Sending get request**

getdata(){

return this.http.get(‘'https://ecommerse-cf4aa.firebaseio.com/data.json’);

}

Inside the component we have :

getRecord(){

this.srve.getRecord().subscribe(

(data : Response)=>{

const res = data.json()

console.log(res);

},

(error)=>console.log(error)

); }

Acting with map method :

map() operator simply : It simply take the old observable and wrap the data into new observable.

Inside the service :

getData(){

return this.http.get(‘…’).map(

(response : Response)=>{

const data = response.json();

return data;

}

);}

Now, inside our component ,

this.srve.getRecord().subscribe(

(users : any[])=> console.log(users),

(error)=>console.log(error)

);

**Render data to screen :**