

Service.ts

- With the help of service we can transfer the data between two components easily.
- And connect UI with backend database. Performs actions like post data from forms, delete data from tables, edit data from tables, get data from db .

```
import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';
import { environment } from 'src/environments/environment';
import { Observable } from 'rxjs';

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

  baseUrl=environment.dataUrl;
  empdata:object[]=[];
  accdata:object[]=[];
  tradata:object[]=[];
  transdata:object[]=[];
  constructor(private http:HttpClient) { }

  readAdmin():Observable<any>{
    let url=`${this.baseUrl}/api/admin/account`;
    return this.http.get(url);
  }
  readOne():Observable<any>{
    let url=`${this.baseUrl}/api/admin/flightone`;
    return this.http.get(url);
  }
  sendfrom(data):Observable<any>{
    let url=`${this.baseUrl}/api/admin/allflights`;
    return this.http.get(url,data);
  }
  newacc(data):Observable<any>{
```

```

        let url=`${this.baseUrl}/api/admin/registration`;
        return this.http.post(url,data);
    }
    resetacc(data):Observable<any>{
        let url=`${this.baseUrl}/api/admin/loopies`;
        return this.http.post(url,data);
    }
    acc(accd):Observable<any>{
        let url=`${this.baseUrl}/api/user/accept/${accd.firstName}`;
        return this.http.post(url,accd);
    }
    readcustomer():Observable<any>{
        let url=`${this.baseUrl}/api/loopies`;
        return this.http.get(url);
    }

    deleteAdmin(deldata):Observable<any>{
        console.log(deldata);
        let url=`${this.baseUrl}/api/loopies/${deldata.id}`;
        return this.http.delete(url);
    }
    updateacc(updata):Observable<any>{
        let url=`${this.baseUrl}/api/loopies/${updata.id}`;
        return this.http.put(url,updata);
    }
}

bookflights(obj):Observable<any>{
    console.log(obj)
    let url=`${this.baseUrl}/api/admin/flightsall`;
    return this.http.get(url);
}

setData(profile:object[]){
    this.empdata.push(profile)
}
getData(){
    return this.empdata;
}

setDATA(traData:object[]){
    this.tradata.push(traData);
}

```

```
}  
getDATA(){  
    return this.tradata;  
}  
SETdata(transData:object[]){  
    this.transdata.push(transData);  
}  
GETdata(){  
    return this.transdata;  
}  
}
```