

Working with Backend APIs

You inject `HttpClient` service into service/components.
(from `@angular/common/http`)

You also need it to providers in `main.ts` (for standalone components)

```
bootstrapApplication (AppComponent &  
  providers: [ provide HttpClient() ]
```

```
  ?).catch (err) => console.log (err));
```

↓
this fn. execution, ~~get~~ sets-up.

`HttpClient` provider.

```
this.httpClient.get ('https://localhost:3000/places')  
  .subscribe();
```

→ observables produced by `HttpClient` is that they'll only emit value typically by default (the default can be changed, with additional config. in the method `get()`)

We could configure `get()`, by adding `{observe: 'response'}`

after URL in `get()`, (`get('http://... places', {observe: 'response'})`)

↓
this is `HttpResponse` object containing body, headers, status-code.

observer : 'events'



will trigger callback fn. for next multiple times, getting triggered on different events in entire request-response lifecycle.

Request sent → event emitted, callback triggered.

Response received → " " " "

So, if the use-case involves doing something with request-response lifecycle, set observer : 'events'

optimistic updating

Update UI before communicating to backend.

Interceptors

special fn. executed when a request is about to be sent / response arrived.