# HTTP interceptors in Angular

Interceptors are a way to do some work for every single HTTP request or response.

Here are a few examples of common use cases for interceptors:

Add a token or some custom HTTP header for all outgoing HTTP requests

Catch HTTP responses to do some custom formatting (i.e. convert CSV to JSON) before handing the data over to your service/component

Log all HTTP activity in the console

Handle Errors globally

Let’s see how to create a basic interceptor that will intercept all HTTP requests.

First, create a service that implements **HttpInterceptor**:

**import { HttpInterceptor} from '@angular/common/http';  
import { Injectable } from '@angular/core';@Injectable()  
export class TokenInterceptorService implements HttpInterceptor { intercept(req: HttpRequest<any>, next: HttpHandler): Observable<HttpEvent<any>> {  
 // All HTTP requests are going to go through this method  
 }  
}**

In order for that interceptor to be called, it has to be added to the list of all **HTTP\_INTERCEPTORS**, which can be done that way in **app.module.ts**:

@NgModule({  
 ...  
 providers: [ {provide: **HTTP\_INTERCEPTORS**, useClass: **TokenInterceptorService**, multi: true}]})  
export class AppModule { }

The above syntax is very important to understand. You could have several different interceptors, which is the reason why we provide our interceptor service with the option **multi: true**.

**Error Handling HttpInterceptor**

To handle errors properly, HttpInterceptor is the best way, and It intercepts Http request.

HttpInterceptor service offers a better way to handle errors in an Angular app systematically.

import {

  HttpEvent,

  HttpHandler,

  HttpRequest,

  HttpErrorResponse,

  HttpInterceptor

} from '@angular/common/http';

import { Observable, throwError } from 'rxjs';

import { catchError } from 'rxjs/operators';

export class ErrorIntercept implements HttpInterceptor {

  intercept(req: HttpRequest<any>, next: HttpHandler): Observable<HttpEvent<any>> {

    return next.handle(req).pipe(

      catchError(error=>{

        console.log(error);

        return throwError(error);

      })

      )

  }

}