

# AngularJS Ajax

Topics : [AngularJS](#)

Written on [January 09, 2024](#)

In AngularJS, you can perform AJAX (Asynchronous JavaScript and XML) requests to retrieve data from a server and update your application dynamically. AngularJS provides the `$http` service to make HTTP requests and interact with web servers. Here's a guide on how to use AJAX with AngularJS:

## Basic AJAX Request:

```
<!DOCTYPE html>
<html lang="en" ng-app="myApp">
<head>
<meta charset="UTF-8">
<title>AngularJS AJAX</title>
<script src="https://ajax.googleapis.com/ajax/libs/angularjs/1.8.2/angular.min.js"></script>
</head>
<body ng-controller="AjaxController">
<button ng-click="getData()">Get Data</button>

<div ng-if="responseData">
<h3>Data received:</h3>
<pre>{{ responseData | json }}</pre>
</div>

<script>
angular.module('myApp', []).controller('AjaxController', function($scope, $http) {
$scope.getData = function() {
$http.get('https://jsonplaceholder.typicode.com/posts/1')
.then(function(response) {
$scope.responseData = response.data;
})
.catch(function(error) {
console.error('Error fetching data:', error);
});
});
});
</script>
</body>
</html>
```

In this example:

- The `$http.get` method is used to make a GET request to a sample JSONPlaceholder endpoint.
- The `then` method is used to handle the successful response.
- The `catch` method is used to handle errors.

## **Sending Data with POST:**

You can use the `$http.post` method to send data to the server using a POST request:

```
$scope.sendData = function() {  
  var dataToSend = { key: 'value' };  
  
  $http.post('https://jsonplaceholder.typicode.com/posts', dataToSend)  
    .then(function(response) {  
      $scope.responseData = response.data;  
    })  
    .catch(function(error) {  
      console.error('Error sending data:', error);  
    });  
};
```

## **Handling Headers:**

You can include custom headers in your request:

```
$scope.getDataWithHeaders = function() {  
  var config = {  
    headers: {  
      'Authorization': 'Bearer yourAccessToken',  
      'Custom-Header': 'CustomValue'  
    }  
  };  
  
  $http.get('https://jsonplaceholder.typicode.com/posts/1', config)  
    .then(function(response) {  
      $scope.responseData = response.data;  
    })  
    .catch(function(error) {  
      console.error('Error fetching data:', error);  
    });  
};
```

## **Canceling Requests:**

AngularJS allows you to cancel HTTP requests using the `$http` service and a cancellation token. This can be useful in scenarios where you want to cancel an ongoing request when a user navigates away from a page or takes some other action.

Here's a simplified example using the `timeout` property:

```
$scope.getDataWithCancellation = function() {
```

```
var canceler = $q.defer();

$http.get('https://jsonplaceholder.typicode.com/posts/1', { timeout: canceler.promise })
  .then(function(response) {
    $scope.responseData = response.data;
  })
  .catch(function(error) {
    if (error && error.status === -1) {
      console.log('Request canceled');
    } else {
      console.error('Error fetching data:', error);
    }
  });

// Simulate canceling after 2 seconds
$timeout(function() {
  canceler.resolve();
}, 2000);
};
```

In this example, the request will be canceled after 2 seconds.

© Copyright **Aryatechno**. All Rights Reserved. Written tutorials and materials by [Aryatechno](#)