

1/125

1. Get Request:

Used to fetch data from the server

Syntax: `$http.get(Curl, config).then(successCallback, errorCallback);`

Ex:

```
$http.get('https://api.example.com/data')
  .then(function(response) {
    console.log('Data fetched successfully:',
      response.data);
  })
  .catch(function(error) {
    console.error('Error fetching data:',
      error);
  });
```

2. POST request:

Used to send data to the server to create a new resource.

Syntax:

`$http.post(Curl, data, config).then(successCallback, errorCallback);`

Ex: `$http.post('https://api.example.com/data', {
 name: 'John Doe',
 age: 30
})`

```
  .then(function(response) {
    console.log('Data posted successfully',
      response.data);
  })
```

```
  .catch(function(error) {
    console.error('Error posting data:',
      error);
  });
```

3. PUT Request:

used to update a resource on the server.

Syntax:

```
$http.put(Curl, data, config).then(successCallback,  
errorCallback);
```

Example:

```
$http.put('https://api.example.com/data/123',  
  {  
    name: 'John Doe',  
    age: 31  
  })  
  .then(function(response) {  
    console.log('Data updated successfully',  
      response.data);  
  })  
  .catch(function(error) {  
    console.error('Error updating data:', error);  
  });
```

4. Delete Request:

used to delete a resource from the server.

Syntax:

```
$http.delete(Curl, config).then(successCallback,  
errorCallback);
```

Example:

```
$http.delete('https://api.example.com/data/123')  
  .then(function(response) {  
    console.log('Data deleted successfully',  
      response.data);  
  })  
  .catch(function(error) {  
    console.error('Error deleting data:', error);  
  });
```