

IApi

August 17, 2026

IApi

Kind: Interface**Source:** Frontend/src/app/services/core/base-api-service/i-api.interface.ts (line 4)**Part of:** Frontend

`IApi<T>` defines the shared contract for API service classes that manage resources of type `T`. It exposes resource metadata through `keyName` and `resourceName`, plus observable methods for loading, saving, and deleting data.

Properties

Property	Type
<code>keyName</code>	<code>string</code>
<code>resourceName</code>	<code>string</code>

Diagram

```
graph LR
  IApi["IApi<T>"]
  IApi --> KeyName["keyName: string"]
  IApi --> ResourceName["resourceName: string"]
  IApi --> Get["get(): Observable<T[]>"]
  IApi --> GetById["getById(): Observable<T>"]
  IApi --> Save["save(): Observable<T>"]
  IApi --> Delete["delete(): Observable<{}>"]
```

Usage

```
import { IApi } from './i-api.interface';

type User = {
```

```

    id: string;
    name: string;
  };

  declare const usersApi: IApi<User>;

  console.log(usersApi.resourceName);
  console.log(usersApi.keyName);

  usersApi.get().subscribe((users) => {
    console.log(users);
  });

  usersApi.getById().subscribe((user) => {
    console.log(user);
  });

  usersApi.save().subscribe((savedUser) => {
    console.log(savedUser);
  });

  usersApi.delete().subscribe(() => {
    console.log('User deleted');
  });

```

AI Coding Instructions

- Implement every `IApi<T>` method with the declared observable return type.
- Keep `keyName` aligned with the identifier field used by the resource model.
- Keep `resourceName` aligned with the API route or resource path used by the service.
- Do not add method parameters to implementations unless the interface contract is updated at the same time.
- Subscribe to returned observables in consuming code, or compose them with RxJS operators.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (1)

- `BaseApiService` — `Frontend/src/app/services/core/base-api-service/base-api.service.ts :1`

