

IApi

August 18, 2026

IApi

Kind: Interface

Source: Pams/Web/Pams.Web/Client/app/services/core/base-api-service/i-api.interface.ts
(line 3)

Part of: Pams

`IApi<T>` defines the contract for API services that manage a resource collection and individual resource records. It exposes resource metadata through `keyName` and `resourceName`, plus observable methods for reading, saving, and deleting data.

Properties

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

Diagram

```
graph LR
    Client --> IApi
    IApi --> keyName
    IApi --> resourceName
    IApi --> get["get(): Observable<T[]>"]
    IApi --> getById["getById(): Observable<T>"]
    IApi --> save["save(): Observable<T>"]
    IApi --> delete["delete(): Observable<{}>"]
```

Usage

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

```

interface Project {
  id: string;
  name: string;
}

function loadProjects(api: IApi<Project>): void {
  api.get().subscribe((projects) => {
    console.log(projects);
  });
}

function saveProject(api: IApi<Project>): void {
  api.save().subscribe((project) => {
    console.log(project);
  });
}

```

AI Coding Instructions

- Implement `IApi<T>` with the resource type used by the API service.
- Keep `keyName` aligned with the identifier field returned by the backend.
- Keep `resourceName` aligned with the API resource route or resource identifier used by the service.
- Return `Observable` values from all interface methods rather than subscribing inside the API service.
- Handle subscriptions, errors, and cleanup in the calling component, facade, or state layer.

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` — `Pams/Web/Pams.Web/Client/app/services/core/base-api-service/base-api.service.ts :1`