

# Namespace TeamDynamix.Api

## Classes

### [BaseRequestBuilder](#)

Abstract base class for building and sending HTTP requests to the TeamDynamix API.

Provides helper methods to create requests with optional JSON bodies and send them using a [TdxBaseClient](#) instance.

### [TdxBaseClient](#)

Base abstract client for interacting with the TeamDynamix API. Provides core HTTP communication features such as:

- Configurable [HttpClient](#) with base address setup.
- Retry policy handling for HTTP 429 (Too Many Requests) throttling responses.
- Token-based authorization header management.
- Generic request sending methods with built-in retry logic and deserialization.

### [TdxClient](#)

Client for interacting with the TeamDynamix API, extending [TdxBaseClient](#). Provides strongly typed request builders for People, Tickets, and Accounts endpoints, and handles authentication using provided credentials.

### [TdxClientFactory](#)

Factory for creating [HttpClient](#) instances with advanced control over the message handler pipeline.

# Class BaseRequestBuilder

Namespace: [TeamDynamix.Api](#)

Assembly: TeamDynamix.Api.dll

Abstract base class for building and sending HTTP requests to the TeamDynamix API. Provides helper methods to create requests with optional JSON bodies and send them using a [TdxBaseClient](#) instance.

```
public abstract class BaseRequestBuilder
```

## Inheritance

[object](#) ← BaseRequestBuilder

## Derived

[AccountItemRequestBuilder](#), [AccountsRequestBuilder](#), [AssetItemRequestBuilder](#),  
[AssetSearchBuilder](#), [AssetsRequestBuilder](#), [AuthRequestBuilder](#),  
[LocationItemRequestBuilder](#), [LocationsRequestBuilder](#), [PeopleRequestBuilder](#),  
[PersonByIdRequestBuilder](#), [PersonLookupRequestBuilder](#), [PersonSearchRequestBuilder](#),  
[TicketCreateObjectRequestBuilder](#), [TicketCreateRequestBuilder](#), [TicketItemRequestBuilder](#),  
[TicketsRequestBuilder](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Constructors

### BaseRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [BaseRequestBuilder](#) class with the specified base path and client.

```
protected BaseRequestBuilder(string path, TdxBaseClient client)
```

## Parameters

`path` [string](#)

The base API path for the resource (relative URI).

`client` [TdxBaseClient](#)

The [TdxBaseClient](#) to use for sending requests.

## Fields

### Client

The [TdxBaseClient](#) instance used to send HTTP requests.

`protected readonly TdxBaseClient Client`

### Field Value

[TdxBaseClient](#)

### Path

The base path (relative URI) for the API endpoint this request builder targets.

`protected readonly string Path`

### Field Value

[string](#)

## Methods

### CreateRequest(HttpMethod, string, object?)

Creates an [HttpRequestMessage](#) with the specified HTTP method, optional relative path, and optional request body serialized as JSON.

```
protected HttpRequestMessage CreateRequest(HttpMethod method, string relativePath = "", object? body = null)
```

## Parameters

method [HttpRequestMessage](#)

The HTTP method to use (GET, POST, etc.).

relativePath [string](#)

An optional relative path to append to the base [Path](#). If null or empty, the base [Path](#) is used as-is.

body [object](#)

An optional object to serialize as JSON in the request body. If null, no body content is added.

## Returns

[HttpRequestMessage](#)

A configured [HttpRequestMessage](#) instance.

## SendAsync(HttpRequestMessage)

Sends an HTTP request asynchronously without expecting a response body.

```
protected Task SendAsync(HttpRequestMessage request)
```

## Parameters

request [HttpRequestMessage](#)

The [HttpRequestMessage](#) to send.

## Returns

[Task](#)

A task representing the asynchronous send operation.

## SendAsync<T>(HttpRequestMessage)

Sends an HTTP request asynchronously and deserializes the JSON response content into an instance of type `T`.

```
protected Task<T> SendAsync<T>(HttpRequestMessage request)
```

### Parameters

`request` [HttpRequestMessage](#)

The [HttpRequestMessage](#) to send.

### Returns

[Task](#)<T>

A task representing the asynchronous operation, with the deserialized response object as result.

### Type Parameters

`T`

The expected type of the deserialized response.

# Class TdxBaseClient

Namespace: [TeamDynamix.Api](#)

Assembly: TeamDynamix.Api.dll

Base abstract client for interacting with the TeamDynamix API. Provides core HTTP communication features such as:

- Configurable [HttpClient](#) with base address setup.
- Retry policy handling for HTTP 429 (Too Many Requests) throttling responses.
- Token-based authorization header management.
- Generic request sending methods with built-in retry logic and deserialization.

```
public abstract class TdxBaseClient
```

## Inheritance

[object](#) ← TdxBaseClient

## Derived

[TdxClient](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Remarks

This class serves as the foundation for more specific TeamDynamix API clients. It handles common concerns like retrying throttled requests, managing authentication tokens, and deserializing API responses into strongly-typed objects.

## Constructors

### TdxBaseClient(HttpClient, string, TdxClientOptions?)

Initializes a new instance of [TdxBaseClient](#), setting the base URI if not already set on the provided [HttpClient](#).

```
protected TdxBaseClient(HttpClient httpClient, string tenant,  
TdxClientOptions? options)
```

## Parameters

`httpClient` [HttpClient](#)

The HTTP client to use for requests.

`tenant` [string](#)

The tenant identifier or base URL part used to build the API base URI if necessary.

`options` [TdxClientOptions](#)

Optional client configuration options.

## TdxBaseClient(HttpClient, TdxClientOptions?)

Initializes a new instance of [TdxBaseClient](#) with the given [HttpClient](#) and client options, creating the retry policy accordingly.

```
protected TdxBaseClient(HttpClient httpClient, TdxClientOptions? options)
```

## Parameters

`httpClient` [HttpClient](#)

The HTTP client to use for requests.

`options` [TdxClientOptions](#)

Optional client configuration options.

## Fields

### HttpClient

The underlying [HttpClient](#) used to send HTTP requests.

```
protected readonly HttpClient HttpClient
```

## Field Value

[HttpClient](#)

## RetryPolicy

The retry policy used to handle throttling responses (HTTP 429) with backoff.

```
protected readonly AsyncRetryPolicy<HttpResponseMessage> RetryPolicy
```

## Field Value

[AsyncRetryPolicy](#)<[HttpResponseMessage](#)>

## TdxClientOptions

The configuration options for this client instance.

```
protected readonly TdxClientOptions TdxClientOptions
```

## Field Value

[TdxClientOptions](#)

## Token

The bearer token used for API authorization. Null if not authenticated.

```
protected string? Token
```

## Field Value

[string](#)

# Methods

## AddAuthorization(HttpRequestMessage)

Adds the bearer token authorization header to the specified HTTP request if the token is set.

```
protected void AddAuthorization(HttpRequestMessage request)
```

### Parameters

**request** [HttpRequestMessage](#)

The HTTP request message to add the authorization header to.

## BuildBaseUri(string, TdxClientOptions)

Builds the base URI for API calls based on the tenant or explicit BaseApiUrl.

```
protected static Uri BuildBaseUri(string tenant, TdxClientOptions options)
```

### Parameters

**tenant** [string](#)

The tenant name or base URL fragment.

**options** [TdxClientOptions](#)

Client options that may contain a full BaseApiUrl override.

### Returns

[Uri](#)

A [Uri](#) representing the API base address.

### Exceptions

[ArgumentException](#)

Thrown if both BaseApiUrl and tenant are null or whitespace.

## CreateRetryPolicy(TdxClientOptions)

Creates a retry policy that handles HTTP 429 responses (rate limiting) with exponential backoff and optional throttle wait/continue callbacks.

```
protected static AsyncRetryPolicy<HttpResponseMessage>
CreateRetryPolicy(TdxClientOptions clientOptions)
```

### Parameters

`clientOptions` [TdxClientOptions](#)

The client options controlling retry behavior.

### Returns

[AsyncRetryPolicy](#)<[HttpResponseMessage](#)>

An [AsyncRetryPolicy<TResult>](#) configured to handle throttling.

## SendRequestAsync(HttpRequestMessage)

Sends an HTTP request and returns the raw [HttpResponseMessage](#). The request will be retried according to the retry policy if throttled.

```
public Task<HttpResponseMessage> SendRequestAsync(HttpRequestMessage request)
```

### Parameters

`request` [HttpRequestMessage](#)

The HTTP request message to send.

### Returns

[Task](#)<[HttpResponseMessage](#)>

The HTTP response message.

## SendRequestAsync<T>(HttpRequestMessage)

Sends an HTTP request, ensures success, and deserializes the JSON response content to the specified type **T**.

```
public Task<T> SendRequestAsync<T>(HttpRequestMessage request)
```

### Parameters

**request** [HttpRequestMessage](#)

The HTTP request message to send.

### Returns

[Task](#)<T>

The deserialized response object of type **T**.

### Type Parameters

**T**

The type to deserialize the response content to.

### Exceptions

[InvalidOperationException](#)

Thrown if the response is unsuccessful, content is empty, or deserialization fails.

## SetToken(string)

Sets the bearer token to use for authorization on subsequent requests.

```
public void SetToken(string token)
```

## Parameters

**token** [string](#) ↗

The bearer token string.

# Class TdxClient

Namespace: [TeamDynamix.Api](#)

Assembly: TeamDynamix.Api.dll

Client for interacting with the TeamDynamix API, extending [TdxBaseClient](#). Provides strongly typed request builders for People, Tickets, and Accounts endpoints, and handles authentication using provided credentials.

```
public class TdxClient : TdxBaseClient
```

## Inheritance

[object](#) ← [TdxBaseClient](#) ← TdxClient

## Inherited Members

[TdxBaseClient.HttpClient](#) , [TdxBaseClient.RetryPolicy](#) , [TdxBaseClient.TdxClientOptions](#) ,  
[TdxBaseClient.Token](#) , [TdxBaseClient.BuildBaseUri\(string, TdxClientOptions\)](#) ,  
[TdxBaseClient.CreateRetryPolicy\(TdxClientOptions\)](#) , [TdxBaseClient.SetToken\(string\)](#) ,  
[TdxBaseClient.AddAuthorization\(HttpRequestMessage\)](#) ,  
[TdxBaseClient.SendRequestAsync\(HttpRequestMessage\)](#) ,  
[TdxBaseClient.SendRequestAsync<T>\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Remarks

Currently only supports admin login.

This client abstracts API interaction details, managing authentication tokens, HTTP client configuration, and exposes high-level endpoint request builders. It supports multiple constructors for flexible configuration, including default options, fully custom options, or supplying a preconfigured [HttpClient](#) instance.

## Constructors

### TdxClient(HttpClient, string, TdxClientOptions?)

Initializes a new instance of [TdxClient](#) using a supplied [HttpClient](#).

```
public TdxClient(HttpClient httpClient, string tenant, TdxClientOptions? options = null)
```

## Parameters

**httpClient** [HttpClient](#)

A preconfigured [HttpClient](#) instance to use for requests.

**tenant** [string](#)

The tenant name or base URL for API access.

**options** [TdxClientOptions](#)

Optional configuration settings for the client.

## Remarks

Use this constructor to supply your own [HttpClient](#) instance, for example, one configured with custom message handlers, logging, or retry policies. If [BaseAddress](#) is not set, it will be inferred from the tenant and options.

## TdxClient(string)

Initializes a new instance of [TdxClient](#) with default retry policy and no throttling callbacks.

```
public TdxClient(string tenant)
```

## Parameters

**tenant** [string](#)

The tenant name or base URL for API access.

## TdxClient(string, TdxClientOptions)

Initializes a new instance of [TdxClient](#) with full configuration options.

```
public TdxClient(string tenant, TdxClientOptions options)
```

## Parameters

**tenant** [string](#)

The tenant name or base URL for API access.

**options** [TdxClientOptions](#)

Optional configuration settings for the client, such as retry policies.

## Properties

### Accounts

Provides access to Accounts-related API requests.

```
public AccountsRequestBuilder Accounts { get; }
```

### Property Value

[AccountsRequestBuilder](#)

### Auth

Provides authentication-related actions such as admin login, SSO login, and current user retrieval for the TeamDynamix API.

```
public AuthRequestBuilder Auth { get; }
```

### Property Value

[AuthRequestBuilder](#)

## Locations

Contains methods for working with locations.

```
public LocationsRequestBuilder Locations { get; }
```

Property Value

[LocationsRequestBuilder](#)

## People

Provides access to People-related API requests.

```
public PeopleRequestBuilder People { get; }
```

Property Value

[PeopleRequestBuilder](#)

## Methods

### Assets(string)

Provides methods for working with assets. To the specified App

```
public AssetsRequestBuilder Assets(string appId)
```

Parameters

appId [string](#)

The application ID used to scope asset requests.

Returns

[AssetsRequestBuilder](#)

A [AssetsRequestBuilder](#) instance for the app

## Tickets(string)

Provides access to Tickets-related API requests for a specified application ID.

```
public TicketsRequestBuilder Tickets(string appId)
```

### Parameters

appId [string](#)

The application ID used to scope ticket requests.

### Returns

[TicketsRequestBuilder](#)

A [TicketsRequestBuilder](#) instance for the specified appId.

# Class TdxClientFactory

Namespace: [TeamDynamix.Api](#)

Assembly: TeamDynamix.Api.dll

Factory for creating [HttpClient](#) instances with advanced control over the message handler pipeline.

```
public static class TdxClientFactory
```

## Inheritance

[object](#) ← TdxClientFactory

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Remarks

This factory allows consumers to compose a customized [HttpClient](#) by chaining any number of [DelegatingHandler](#) instances such as logging, authentication, retry policies, or telemetry.

By providing a collection of handlers, the factory builds a pipeline that executes handlers in the specified order, enabling flexible, reusable, and testable HTTP communication tailored to specific needs.

Use this factory when the default [HttpClient](#) behavior is insufficient or when custom cross-cutting concerns need to be injected into the HTTP request/response pipeline.

## Methods

### CreateHttpClient(IEnumerable<DelegatingHandler>, string, HttpMessageHandler?)

Creates a new [HttpClient](#) instance configured with a pipeline of [DelegatingHandler](#) instances.

```
public static HttpClient CreateHttpClient(IEnumerable<DelegatingHandler> handlers,
string tenantOrUrl, HttpMessageHandler? rootHandler = null)
```

## Parameters

handlers [IEnumerable](#)<[DelegatingHandler](#)>

A collection of [DelegatingHandler](#) instances to be chained together in the order provided. These handlers can implement cross-cutting concerns like logging, retry, or authentication.

tenantOrUrl [string](#)

The base URI for the [HttpClient](#). This can be a full URI string, or a tenant identifier which will be used to construct a standard TeamDynamix API URL.

rootHandler [HttpMessageHandler](#)

An optional root [HttpMessageHandler](#) to terminate the pipeline. If null, the default [HttpClientHandler](#) is used.

## Returns

[HttpClient](#)

A new [HttpClient](#) instance with the specified handler pipeline and base address.

# Namespace TeamDynamix.Api.Accounts

## Classes

### [AccountItemRequestBuilder](#)

Provides operations to retrieve information about a specific account/department in the TeamDynamix system.

### [AccountsRequestBuilder](#)

Contains methods for working with accounts/departments in the TeamDynamix API.

# Class AccountItemRequestBuilder

Namespace: [TeamDynamix.Api.Accounts](#)

Assembly: TeamDynamix.Api.dll

Provides operations to retrieve information about a specific account/department in the TeamDynamix system.

```
public class AccountItemRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← AccountItemRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### AccountItemRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [AccountItemRequestBuilder](#) class.

```
public AccountItemRequestBuilder(string path, TdxBaseClient client)
```

#### Parameters

**path** [string](#)

The path segment for the specific account resource.

**client** [TdxBaseClient](#)

The TdxBaseClient used to send HTTP requests.

# Methods

## GetAsync()

Retrieves detailed information for a specific [Account](#).

```
public Task<Account> GetAsync()
```

Returns

[Task](#)<[Account](#)>

An [Account](#) object with full details.

## UpdateAsync(Account)

Updates the specified account with the provided field values.

Important: Any properties not set in the `account` object will be treated as `null` by the TeamDynamix API and will overwrite existing data. This is due to how the TDX API handles this update operation.

Corresponds to the endpoint: `PUT`

`https://{{tenant}}.teamdynamix.com/TDWebApi/api/accounts/{id}`.

```
public Task<Account> UpdateAsync(Account account)
```

Parameters

`account` [Account](#)

The account object containing the updated values. Fields you want to retain should be mocked from the original object.

Returns

[Task](#)<[Account](#)>

A task representing the asynchronous operation, containing the updated account.

# Class AccountsRequestBuilder

Namespace: [TeamDynamix.Api.Accounts](#)

Assembly: TeamDynamix.Api.dll

Contains methods for working with accounts/departments in the TeamDynamix API.

```
public class AccountsRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← AccountsRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### AccountsRequestBuilder(TdxBaseClient)

Initializes a new instance of the [AccountsRequestBuilder](#) class with the specified client. The base path is set to "accounts".

```
public AccountsRequestBuilder(TdxBaseClient client)
```

## Parameters

**client** [TdxBaseClient](#)

The [TdxBaseClient](#) used to send requests.

## Properties

## this[string]

Gets a request builder for a specific account by its identifier. This allows further operations scoped to a single account. Corresponds to the endpoint: **GET** <https://{{tenant}}.teamdynamix.com/TDWebApi/api/accounts/{id}>.

```
public AccountItemRequestBuilder this[string] accountId { get; }
```

## Parameters

accountId [string](#)

The ID of the account.

## Property Value

[AccountItemRequestBuilder](#)

## Methods

### GetAsync()

Retrieves a list of all active accounts/departments. This method calls the endpoint: **GET** <https://{{tenant}}.teamdynamix.com/TDWebApi/api/accounts>. Note that this does not return full account or department details.

```
public Task<List<Account>> GetAsync()
```

## Returns

[Task](#)<[List](#)<[Account](#)>>

A task that represents the asynchronous operation. The task result contains a list of [Account](#) objects.

# Namespace TeamDynamix.Api.Accounts.Models

## Classes

### [Account](#)

Represents a physical or virtual location in the TeamDynamix system.

# Class Account

Namespace: [TeamDynamix.Api.Accounts.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a physical or virtual location in the TeamDynamix system.

```
public class Account
```

## Inheritance

[object](#) ← Account

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### Address1

The first address line of the account. Editable via the API.

```
[JsonProperty("Address1")]
public string? Address1 { get; set; }
```

### Property Value

[string](#)

### Address2

The second address line of the account. Editable via the API.

```
[JsonProperty("Address2")]
public string? Address2 { get; set; }
```

Property Value

[string](#)

## Address3

The third address line of the account. Editable via the API.

```
[JsonProperty("Address3")]
public string? Address3 { get; set; }
```

Property Value

[string](#)

## Address4

The fourth address line of the account. Editable via the API.

```
[JsonProperty("Address4")]
public string? Address4 { get; set; }
```

Property Value

[string](#)

## Attributes

The custom attributes of the account. Editable via the API.

```
[JsonProperty("Attributes")]
public List<CustomAttribute>? Attributes { get; set; }
```

Property Value

[List](#)<[CustomAttribute](#)>

## City

The city of the account. Editable via the API.

```
[JsonProperty("City")]
public string? City { get; set; }
```

### Property Value

[string](#) ↗

## Code

The code for the account. Editable via the API.

```
[JsonProperty("Code")]
public string? Code { get; set; }
```

### Property Value

[string](#) ↗

## Country

The country of the account. Editable via the API.

```
[JsonProperty("Country")]
public string? Country { get; set; }
```

### Property Value

[string](#) ↗

## CreatedDate

The date the account was created.

```
[JsonProperty("CreatedDate")]
public DateTime CreatedDate { get; set; }
```

## Property Value

[DateTime](#)

## Fax

The fax number of the account. Editable via the API.

```
[JsonProperty("Fax")]
public string? Fax { get; set; }
```

## Property Value

[string](#)

## Id

The ID of the account.

```
[JsonProperty("ID")]
public required int Id { get; set; }
```

## Property Value

[int](#)

## IndustryId

The ID of the industry associated with the account. Editable via the API.

```
[JsonProperty("IndustryID")]
public int IndustryId { get; set; }
```

Property Value

[int](#)

## IndustryName

The name of the industry associated with the account.

```
[JsonProperty("IndustryName")]
public string? IndustryName { get; set; }
```

Property Value

[string](#)

## IsActive

The active status of the account. Editable via the API.

```
[JsonProperty("IsActive")]
public bool IsActive { get; set; }
```

Property Value

[bool](#)

## ManagerFullName

The full name of the manager for the account.

```
[JsonProperty("ManagerFullName")]
public string? ManagerFullName { get; set; }
```

Property Value

[string](#)

## ManagerUid

The UID of the manager for the account. Editable via the API.

```
[JsonProperty("ManagerUID")]
public string? ManagerUid { get; set; }
```

### Property Value

[string](#)

## ModifiedDate

The last modified date of the account.

```
[JsonProperty("ModifiedDate")]
public DateTime ModifiedDate { get; set; }
```

### Property Value

[DateTime](#)

## Name

The name of the account. This field is required and editable via the API.

```
[JsonProperty("Name")]
public string? Name { get; set; }
```

### Property Value

[string](#)

## Notes

Notes for the account. Editable via the API.

```
[JsonProperty("Notes")]
public string? Notes { get; set; }
```

## PropertyValue

[string](#) ↗

## ParentId

The ID of the parent associated with the account, or null if the account has no parent.  
Editable via the API.

```
[JsonProperty("ParentID")]
public int? ParentId { get; set; }
```

## PropertyValue

[int](#) ↗?

## ParentName

The name of the parent associated with the account, or null if the account has no parent.

```
[JsonProperty("ParentName")]
public string? ParentName { get; set; }
```

## PropertyValue

[string](#) ↗

## Phone

The phone number of the account. Editable via the API.

```
[JsonProperty("Phone")]
public string? Phone { get; set; }
```

Property Value

[string](#)

## PostalCode

The postal code of the account. Editable via the API.

```
[JsonProperty("PostalCode")]
public string? PostalCode { get; set; }
```

Property Value

[string](#)

## StateAbbr

The abbreviation of the state/province associated with the account. Editable via the API.

```
[JsonProperty("StateAbbr")]
public string? StateAbbr { get; set; }
```

Property Value

[string](#)

## StateName

The state/province of the account.

```
[JsonProperty("StateName")]
public string? StateName { get; set; }
```

Property Value

[string](#)

## Url

The website URL of the account. Editable via the API.

```
[JsonProperty("Url")]
public string? Url { get; set; }
```

Property Value

[string](#) ↗

# Namespace TeamDynamix.Api.Apps. Models

## Classes

### [UserApplication](#)

An association that a user has with an application.

# Class UserApplication

Namespace: [TeamDynamix.Api.Apps.Models](#)

Assembly: TeamDynamix.Api.dll

An association that a user has with an application.

```
public class UserApplication
```

## Inheritance

[object](#) ← UserApplication

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### Description

The description of the application. Nullable.

```
[JsonProperty("Description")]
public string? Description { get; set; }
```

### Property Value

[string](#)

### ID

The ID of the application.

```
[JsonProperty("ID")]
public int ID { get; set; }
```

Property Value

[int](#)

## IsActive

The active status of the application.

```
[JsonProperty("IsActive")]
public bool IsActive { get; set; }
```

Property Value

[bool](#)

## IsAdministrator

The administrator status of the user for the application.

```
[JsonProperty("IsAdministrator")]
public bool IsAdministrator { get; set; }
```

Property Value

[bool](#)

## IsDefault

The default status of the application.

```
[JsonProperty("IsDefault")]
public bool IsDefault { get; set; }
```

Property Value

[bool](#)

## Name

The name of the application.

```
[JsonProperty("Name")]
public string? Name { get; set; }
```

## Property Value

[string](#)

## SecurityRoleId

The ID of the security role that the user has within the application. Nullable.

```
[JsonProperty("SecurityRoleId")]
public string? SecurityRoleId { get; set; }
```

## Property Value

[string](#)

## SecurityRoleName

The name of the security role that the user has within the application. Nullable.

```
[JsonProperty("SecurityRoleName")]
public string? SecurityRoleName { get; set; }
```

## Property Value

[string](#)

## SystemClass

The system-defined class of the application.

```
[JsonProperty("SystemClass")]
public string? SystemClass { get; set; }
```

Property Value

[string](#) ↗

# Namespace TeamDynamix.Api.Assets

## Classes

### [AssetItemRequestBuilder](#)

Provides operation for a single asset.

### [AssetSearchBuilder](#)

### [AssetsRequestBuilder](#)

# Class AssetItemRequestBuilder

Namespace: [TeamDynamix.Api.Assets](#)

Assembly: TeamDynamix.Api.dll

Provides operation for a single asset.

```
public class AssetItemRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← AssetItemRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### AssetItemRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [AssetItemRequestBuilder](#) class.

```
public AssetItemRequestBuilder(string path, TdxBaseClient client)
```

## Parameters

path [string](#)

Path to the specific asset.

client [TdxBaseClient](#)

The client used to send requests.

# Methods

## GetAsync()

```
public Task<Asset> GetAsync()
```

Returns

[Task](#) <[Asset](#)>

# Class AssetSearchBuilder

Namespace: [TeamDynamix.Api.Assets](#)

Assembly: TeamDynamix.Api.dll

```
public class AssetSearchBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← AssetSearchBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

AssetSearchBuilder(string,  
Action<AssetSearchOptions>, TdxBaseClient)

Searches assets. Will not return full asset information. The following properties will not be included in the results:

[Attachments](#)

[Attributes](#)

To retrieve such information, you must load an asset individually.

```
public AssetSearchBuilder(string path, Action<AssetSearchOptions> searchOptions,  
TdxBaseClient client)
```

## Parameters

path [string](#)

searchOptions [Action<AssetSearchOptions>](#)

client [TdxBaseClient](#)

## Methods

### GetAsync()

```
public Task<List<Asset>> GetAsync()
```

Returns

[Task](#)<[List](#)<[Asset](#)>>

# Class AssetsRequestBuilder

Namespace: [TeamDynamix.Api.Assets](#)

Assembly: TeamDynamix.Api.dll

```
public class AssetsRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← AssetsRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### AssetsRequestBuilder(TdxBaseClient, string)

Initializes a new instance of the [AssetsRequestBuilder](#) class.

```
public AssetsRequestBuilder(TdxBaseClient client, string appId)
```

#### Parameters

**client** [TdxBaseClient](#)

The TDX base client used to send HTTP requests.

**appId** [string](#)

The ID of the asset application.

## Exceptions

## [ArgumentException](#)

Thrown if `appId` is null or empty.

# Properties

## AppId

Gets the ID of the asset application associated with this request builder.

```
public string AppId { get; }
```

Property Value

[string](#)

## this[string]

Provides operations for a specific asset by its ID.

```
public AssetItemRequestBuilder this[string assetId] { get; }
```

Parameters

`assetId` [string](#)

The ID of the asset to manage.

Property Value

[AssetItemRequestBuilder](#)

An [AssetItemRequestBuilder](#) for the specified asset.

# Methods

## PostAsync(Asset)

```
public Task<Asset> PostAsync(Asset asset)
```

Parameters

asset [Asset](#)

Returns

[Task](#) <Asset>

## Search(Action<AssetSearchOptions>)

```
public AssetSearchBuilder Search(Action<AssetSearchOptions> searchOptions)
```

Parameters

searchOptions [Action](#) <AssetSearchOptions>

Returns

[AssetSearchBuilder](#)

# Namespace TeamDynamix.Api.Assets.Models

## Classes

### [Asset](#)

Represents an asset in the TeamDynamix system.

### [AssetSearchOptions](#)

A set of filtering options used for searching assets.

# Class Asset

Namespace: [TeamDynamix.Api.Assets.Models](#)

Assembly: TeamDynamix.Api.dll

Represents an asset in the TeamDynamix system.

```
public class Asset
```

## Inheritance

[object](#) ← Asset

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AcquisitionDate

The acquisition date of the asset.

```
[JsonProperty("AcquisitionDate")]
public DateTime AcquisitionDate { get; set; }
```

### Property Value

[DateTime](#)

### AppId

The ID of the asset/CI application containing the asset.

```
[JsonProperty("AppID")]
public int AppId { get; set; }
```

Property Value

[int](#)

## AppName

The name of the asset/CI application containing the asset.

```
[JsonProperty("AppName")]
public string? AppName { get; set; }
```

Property Value

[string](#)

## Attachments

The attachments associated with the asset.

```
[JsonProperty("Attachments")]
public List<Attachment>? Attachments { get; set; }
```

Property Value

[List](#)<[Attachment](#)>

## Attributes

The custom attributes associated with the asset.

```
[JsonProperty("Attributes")]
public List<CustomAttribute>? Attributes { get; set; }
```

Property Value

[List](#)<[CustomAttribute](#)>

## ConfigurationItemId

The ID of the configuration item record associated with the asset.

```
[JsonProperty("ConfigurationItemID")]
public int ConfigurationItemId { get; set; }
```

### Property Value

[int](#)

## CreatedDate

The created date of the asset.

```
[JsonProperty("CreatedDate")]
public DateTime CreatedDate { get; set; }
```

### Property Value

[DateTime](#)

## CreatedFullName

The full name of the user who created the asset.

```
[JsonProperty("CreatedFullName")]
public string? CreatedFullName { get; set; }
```

### Property Value

[string](#)

## CreatedUid

The UID of the user who created the asset.

```
[JsonProperty("CreatedUID")]
public Guid CreatedUid { get; set; }
```

Property Value

[Guid](#)

## ExpectedReplacementDate

The expected replacement date of the asset.

```
[JsonProperty("ExpectedReplacementDate")]
public DateTime ExpectedReplacementDate { get; set; }
```

Property Value

[DateTime](#)

## ExternalId

The external ID of the asset used for mapping to external sources.

```
[JsonProperty("ExternalID")]
public string? ExternalId { get; set; }
```

Property Value

[string](#)

## ExternalSourceId

The ID of the configuration item source associated with the asset.

```
[JsonProperty("ExternalSourceID")]
public int ExternalSourceId { get; set; }
```

## Property Value

[int ↗](#)

## ExternalSourceName

The name of the configuration item source associated with the asset.

```
[JsonProperty("ExternalSourceName")]
public string? ExternalSourceName { get; set; }
```

## Property Value

[string ↗](#)

## FormId

The ID of the form associated with the asset.

```
[JsonProperty("FormID")]
public int FormId { get; set; }
```

## Property Value

[int ↗](#)

## FormName

The name of the form associated with the asset.

```
[JsonProperty("FormName")]
public string? FormName { get; set; }
```

## Property Value

[string ↗](#)

## Id

The ID of the asset.

```
[JsonProperty("ID")]
public int Id { get; set; }
```

## Property Value

[int](#)

## LocationId

The ID of the location associated with the asset.

```
[JsonProperty("LocationID")]
public int LocationId { get; set; }
```

## Property Value

[int](#)

## LocationName

The name of the location associated with the asset.

```
[JsonProperty("LocationName")]
public string? LocationName { get; set; }
```

## Property Value

[string](#)

## LocationRoomId

The ID of the location room associated with the asset.

```
[JsonProperty("LocationRoomID")]
public int LocationRoomId { get; set; }
```

Property Value

[int](#)

## LocationRoomName

The name of the location room associated with the asset.

```
[JsonProperty("LocationRoomName")]
public string? LocationRoomName { get; set; }
```

Property Value

[string](#)

## MaintenanceScheduleId

The ID of the maintenance window associated with the asset.

```
[JsonProperty("MaintenanceScheduleID")]
public int MaintenanceScheduleId { get; set; }
```

Property Value

[int](#)

## MaintenanceScheduleName

The name of the maintenance window associated with the asset.

```
[JsonProperty("MaintenanceScheduleName")]
public string? MaintenanceScheduleName { get; set; }
```

Property Value

[string](#)

## ManufacturerId

The ID of the manufacturer associated with the asset.

```
[JsonProperty("ManufacturerID")]
public int ManufacturerId { get; set; }
```

Property Value

[int](#)

## ManufacturerName

The name of the manufacturer associated with the asset.

```
[JsonProperty("ManufacturerName")]
public string? ManufacturerName { get; set; }
```

Property Value

[string](#)

## ModifiedDate

The last modified date of the asset.

```
[JsonProperty("ModifiedDate")]
public DateTime ModifiedDate { get; set; }
```

Property Value

[DateTime](#)

## ModifiedFullName

The full name of the user who last modified the asset.

```
[JsonProperty("ModifiedFullName")]
public string? ModifiedFullName { get; set; }
```

### Property Value

[string](#)

## ModifiedUid

The UID of the user who last modified the asset.

```
[JsonProperty("ModifiedUID")]
public Guid ModifiedUid { get; set; }
```

### Property Value

[Guid](#)

## Name

The name of the asset.

```
[JsonProperty("Name")]
public string? Name { get; set; }
```

### Property Value

[string](#)

## OwningCustomerId

The UID of the owning user associated with the asset.

```
[JsonProperty("OwningCustomerID")]
public Guid OwningCustomerId { get; set; }
```

Property Value

[Guid](#)

## OwningCustomerName

The name of the owning user associated with the asset.

```
[JsonProperty("OwningCustomerName")]
public string? OwningCustomerName { get; set; }
```

Property Value

[string](#)

## OwningDepartmentId

The ID of the owning account/department associated with the asset.

```
[JsonProperty("OwningDepartmentID")]
public int OwningDepartmentId { get; set; }
```

Property Value

[int](#)

## OwningDepartmentName

The name of the owning account/department associated with the asset.

```
[JsonProperty("OwningDepartmentName")]
public string? OwningDepartmentName { get; set; }
```

## Property Value

[string](#) ↗

## ParentId

The ID of the parent associated with the asset.

```
[JsonProperty("ParentID")]
public int ParentId { get; set; }
```

## Property Value

[int](#) ↗

## ParentName

The name of the parent associated with the asset.

```
[JsonProperty("ParentName")]
public string? ParentName { get; set; }
```

## Property Value

[string](#) ↗

## ParentSerialNumber

The serial number of the parent associated with the asset.

```
[JsonProperty("ParentSerialNumber")]
public string? ParentSerialNumber { get; set; }
```

## Property Value

[string](#) ↗

## ParentTag

The service tag of the parent associated with the asset.

```
[JsonProperty("ParentTag")]
public string? ParentTag { get; set; }
```

### Property Value

[string](#)

## ProductModelId

The ID of the product model associated with the asset.

```
[JsonProperty("ProductModelID")]
public int ProductModelId { get; set; }
```

### Property Value

[int](#)

## ProductModelName

The name of the product model associated with the asset.

```
[JsonProperty("ProductModelName")]
public string? ProductModelName { get; set; }
```

### Property Value

[string](#)

## PurchaseCost

The purchase cost of the asset.

```
[JsonProperty("PurchaseCost")]
public double PurchaseCost { get; set; }
```

Property Value

[double](#)

## RequestingCustomerId

The UID of the requesting user associated with the asset.

```
[JsonProperty("RequestingCustomerID")]
public Guid RequestingCustomerId { get; set; }
```

Property Value

[Guid](#)

## RequestingCustomerName

The name of the requesting user associated with the asset.

```
[JsonProperty("RequestingCustomerName")]
public string? RequestingCustomerName { get; set; }
```

Property Value

[string](#)

## RequestingDepartmentId

The ID of the requesting account/department associated with the asset.

```
[JsonProperty("RequestingDepartmentID")]
public int RequestingDepartmentId { get; set; }
```

Property Value

[int ↗](#)

## RequestingDepartmentName

The name of the requesting account/department associated with the asset.

```
[JsonProperty("RequestingDepartmentName")]
public string? RequestingDepartmentName { get; set; }
```

Property Value

[string ↗](#)

## SerialNumber

The serial number of the asset.

```
[JsonProperty("SerialNumber")]
public string? SerialNumber { get; set; }
```

Property Value

[string ↗](#)

## StatusId

The ID of the status associated with the asset.

```
[JsonProperty("StatusID")]
public int StatusId { get; set; }
```

Property Value

[int ↗](#)

## StatusName

The name of the status associated with the asset.

```
[JsonProperty("StatusName")]
public string? StatusName { get; set; }
```

### Property Value

[string](#)

## SupplierId

The ID of the supplier associated with the asset.

```
[JsonProperty("SupplierID")]
public int SupplierId { get; set; }
```

### Property Value

[int](#)

## SupplierName

The name of the supplier associated with the asset.

```
[JsonProperty("SupplierName")]
public string? SupplierName { get; set; }
```

### Property Value

[string](#)

## Tag

The service tag of the asset.

```
[JsonProperty("Tag")]
public string? Tag { get; set; }
```

Property Value

[string](#)

## Uri

The URI to retrieve full details of the asset via the API.

```
[JsonProperty("Uri")]
public string? Uri { get; set; }
```

Property Value

[string](#)

# Class AssetSearchOptions

Namespace: [TeamDynamix.Api.Assets.Models](#)

Assembly: TeamDynamix.Api.dll

A set of filtering options used for searching assets.

```
public class AssetSearchOptions
```

## Inheritance

[object](#) ← AssetSearchOptions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AcquisitionDateFrom

The minimum acquisition date to filter on.

```
[JsonProperty("AcquisitionDateFrom")]
public DateTime AcquisitionDateFrom { get; set; }
```

Property Value

[DateTime](#)

### AcquisitionDateTo

The maximum acquisition date to filter on.

```
[JsonProperty("AcquisitionDateTo")]
public DateTime AcquisitionDateTo { get; set; }
```

Property Value

[DateTime](#)

## ContractEndDateFrom

The minimum contract end date to filter on.

```
[JsonProperty("ContractEndDateFrom")]
public DateTime ContractEndDateFrom { get; set; }
```

Property Value

[DateTime](#)

## ContractEndDateTo

The maximum contract end date to filter on.

```
[JsonProperty("ContractEndDateTo")]
public DateTime ContractEndDateTo { get; set; }
```

Property Value

[DateTime](#)

## ContractIds

The contract IDs to filter on.

```
[JsonProperty("ContractIDs")]
public int[]? ContractIds { get; set; }
```

Property Value

[int](#)[]

## ContractProviderId

The contract provider ID to filter on.

```
[JsonProperty("ContractProviderID")]
public int ContractProviderId { get; set; }
```

### Property Value

[int](#)

## CreatedDateFrom

The minimum created date to filter on.

```
[JsonProperty("CreatedDateFrom")]
public DateTime CreatedDateFrom { get; set; }
```

### Property Value

[DateTime](#)

## CreatedDateTo

The maximum created date to filter on.

```
[JsonProperty("CreatedDateTo")]
public DateTime CreatedDateTo { get; set; }
```

### Property Value

[DateTime](#)

## CustomAttributes

The custom attributes to filter on.

```
[JsonProperty("CustomAttributes")]
public List<CustomAttribute>? CustomAttributes { get; set; }
```

Property Value

[List](#)<[CustomAttribute](#)>

## ExcludeContractIds

The contract IDs to exclude on.

```
[JsonProperty("ExcludeContractIDs")]
public int[]? ExcludeContractIds { get; set; }
```

Property Value

[int](#)[]

## ExcludeTicketIds

The ticket IDs to exclude on.

```
[JsonProperty("ExcludeTicketIDs")]
public int[]? ExcludeTicketIds { get; set; }
```

Property Value

[int](#)[]

## ExpectedReplacementDateFrom

The minimum expected replacement date to filter on.

```
[JsonProperty("ExpectedReplacementDateFrom")]
public DateTime ExpectedReplacementDateFrom { get; set; }
```

Property Value

[DateTime](#)

## ExpectedReplacementDateTo

The maximum expected replacement date to filter on.

```
[JsonProperty("ExpectedReplacementDateTo")]
public DateTime ExpectedReplacementDateTo { get; set; }
```

Property Value

[DateTime](#)

## ExternalIDs

The external IDs to filter on.

```
[JsonProperty("ExternalIDs")]
public string[]? ExternalIDs { get; set; }
```

Property Value

[string](#)[]

## FormIds

The form IDs to filter on.

```
[JsonProperty("FormIDs")]
public int[]? FormIds { get; set; }
```

Property Value

[int](#)[]

## IsInService

The "in service" status to filter on.

```
[JsonProperty("IsInService")]
public bool? IsInService { get; set; }
```

Property Value

[bool](#)?

## LocationIds

The location IDs to filter on.

```
[JsonProperty("LocationIDs")]
public int[]? LocationIds { get; set; }
```

Property Value

[int](#)[]

## MaintenanceScheduleIds

The maintenance window IDs to filter on.

```
[JsonProperty("MaintenanceScheduleIDs")]
public int[]? MaintenanceScheduleIds { get; set; }
```

Property Value

[int](#)[]

## ManufacturerIds

The manufacturer IDs to filter on.

```
[JsonProperty("ManufacturerIDs")]
public int[]? ManufacturerIds { get; set; }
```

## Property Value

[int](#)[]

## MaxResults

The maximum number of records to return.

```
[JsonProperty("MaxResults")]
public int? MaxResults { get; set; }
```

## Property Value

[int](#)?

## Remarks

As a note when using this: The TDX API does not offer paging

## ModifiedDateFrom

The minimum modified date to filter on.

```
[JsonProperty("ModifiedDateFrom")]
public DateTime ModifiedDateFrom { get; set; }
```

## Property Value

[DateTime](#)

## ModifiedDateTo

The maximum modified date to filter on.

```
[JsonProperty("ModifiedDateTo")]
public DateTime ModifiedDateTo { get; set; }
```

Property Value

[DateTime](#)

## OnlyParentAssets

Whether only parent assets should be returned.

```
[JsonProperty("OnlyParentAssets")]
public bool OnlyParentAssets { get; set; }
```

Property Value

[bool](#)

## OwningCustomerIds

The owner UIDs to filter on.

```
[JsonProperty("OwningCustomerIDs")]
public Guid[]? OwningCustomerIds { get; set; }
```

Property Value

[Guid](#)[]

## OwningCustomerIdsPast

The past owner UIDs to filter on.

```
[JsonProperty("OwningCustomerIDsPast")]
public Guid[]? OwningCustomerIdsPast { get; set; }
```

Property Value

[Guid](#)[]

## OwningDepartmentIds

The owning account/department IDs to filter on.

```
[JsonProperty("OwningDepartmentIDs")]
public int[]? OwningDepartmentIds { get; set; }
```

Property Value

[int](#)[]

## OwningDepartmentIdsPast

The past owning account/department IDs to filter on.

```
[JsonProperty("OwningDepartmentIDsPast")]
public int[]? OwningDepartmentIdsPast { get; set; }
```

Property Value

[int](#)[]

## ParentIds

The parent asset IDs to filter on.

```
[JsonProperty("ParentIDs")]
public int[]? ParentIds { get; set; }
```

Property Value

[int](#)[]

## ProductModelIds

The product model IDs to filter on.

```
[JsonProperty("ProductModelIDs")]
public int[]? ProductModelIds { get; set; }
```

### Property Value

[int](#)[]

## PurchaseCostFrom

The minimum purchase cost to filter on.

```
[JsonProperty("PurchaseCostFrom")]
public double PurchaseCostFrom { get; set; }
```

### Property Value

[double](#)

## PurchaseCostTo

The maximum purchase cost to filter on.

```
[JsonProperty("PurchaseCostTo")]
public double PurchaseCostTo { get; set; }
```

### Property Value

[double](#)

## RequestingCustomerIds

The requestor UIDs to filter on.

```
[JsonProperty("RequestingCustomerIDs")]
public Guid[]? RequestingCustomerIds { get; set; }
```

Property Value

[Guid](#)[]

## RequestingDepartmentIds

The requesting account/department IDs to filter on.

```
[JsonProperty("RequestingDepartmentIDs")]
public int[]? RequestingDepartmentIds { get; set; }
```

Property Value

[int](#)[]

## RoomId

The location room ID to filter on.

```
[JsonProperty("RoomID")]
public int RoomId { get; set; }
```

Property Value

[int](#)

## SavedSearchId

The ID of the saved search associated with this search.

```
[JsonProperty("SavedSearchID")]
public int SavedSearchId { get; set; }
```

## Property Value

[int](#)

## SearchText

The search text to filter on. When specified, results will be sorted by their text relevancy.

```
[JsonProperty("SearchText")]
public string? SearchText { get; set; }
```

## Property Value

[string](#)

## Remarks

Pass an empty string to return everything

## SerialLike

The text to perform a LIKE search on the asset serial number and service tag.

```
[JsonProperty("SerialLike")]
public string? SerialLike { get; set; }
```

## Property Value

[string](#)

## StatusIds

The current status IDs to filter on.

```
[JsonProperty("StatusIDs")]
public int[]? StatusIds { get; set; }
```

Property Value

[int](#)[]

## StatusIdsPast

The past status IDs to filter on.

```
[JsonProperty("StatusIDsPast")]
public int[]? StatusIdsPast { get; set; }
```

Property Value

[int](#)[]

## SupplierIds

The supplier IDs to filter on.

```
[JsonProperty("SupplierIDs")]
public int[]? SupplierIds { get; set; }
```

Property Value

[int](#)[]

## TicketIds

The ticket IDs to filter on.

```
[JsonProperty("TicketIDs")]
public int[]? TicketIds { get; set; }
```

Property Value

[int](#)[]

## UsingCustomerIds

The using person UIDs to filter on.

```
[JsonProperty("UsingCustomerIDs")]
public Guid[]? UsingCustomerIds { get; set; }
```

Property Value

[Guid](#)[]

## UsingDepartmentIds

The using account/department IDs to filter on.

```
[JsonProperty("UsingDepartmentIDs")]
public int[]? UsingDepartmentIds { get; set; }
```

Property Value

[int](#)[]

# Namespace TeamDynamix.Api.Attachments.Models

## Classes

### [Attachment](#)

An attachment.

## Enums

### [AttachmentType](#)

Describes the different types of supported attachments.

# Class Attachment

Namespace: [TeamDynamix.Api.Attachments.Models](#)

Assembly: TeamDynamix.Api.dll

An attachment.

```
public class Attachment
```

## Inheritance

[object](#) ← Attachment

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AttachmentType

The type of the attachment.

```
[JsonProperty("AttachmentType")]
public AttachmentType AttachmentType { get; set; }
```

### Property Value

[AttachmentType](#)

### ContentUri

The URI to retrieve the contents of the attachment via the web API.

```
[JsonProperty("ContentUri")]
public string ContentUri { get; set; }
```

## Property Value

[string](#)

## CreatedDate

The upload date of the attachment.

```
[JsonProperty("CreatedDate")]
public DateTime CreatedDate { get; set; }
```

## Property Value

[DateTime](#)

## CreatedFullName

The full name of the user who uploaded the attachment.

```
[JsonProperty("CreatedFullName")]
public string CreatedFullName { get; set; }
```

## Property Value

[string](#)

## CreatedUid

The UID of the user who uploaded the attachment.

```
[JsonProperty("CreatedUID")]
public Guid CreatedUid { get; set; }
```

## Property Value

[Guid](#)

## Id

The ID of the attachment.

```
[JsonProperty("ID")]
public string? Id { get; set; }
```

## Property Value

[string](#)

## ItemId

The ID of the item associated with the attachment.

```
[JsonProperty("ItemID")]
public int ItemId { get; set; }
```

## Property Value

[int](#)

## Name

The file name of the attachment.

```
[JsonProperty("Name")]
public string Name { get; set; }
```

## Property Value

[string](#)

## Size

The size of the attachment, in bytes.

```
[JsonProperty("Size")]
public int Size { get; set; }
```

Property Value

[int](#)

## Uri

The URI to retrieve the full details of the attachment via the web API.

```
[JsonProperty("Uri")]
public string Uri { get; set; }
```

Property Value

[string](#)

# Enum AttachmentType

Namespace: [TeamDynamix.Api.Attachments.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different types of supported attachments.

```
public enum AttachmentType
```

## Fields

**Announcement** = 7

An announcement attachment.

**Asset** = 27

An asset attachment.

**CalendarEvent** = 57

A calendar event attachment.

**ConfigurationItem** = 63

A configuration item attachment.

**Contract** = 29

An asset contract attachment.

**Expense** = 62

An expense attachment.

**Forums** = 13

A forum post attachment.

**Issue** = 3

An issue attachment.

**KnowledgeBase** = 26

A Knowledge Base article attachment.

**Location** = 71

A location attachment.

**None** = 0

Indicates that the type of attachment is unknown.

**PortfolioIssue** = 83

A portfolio issue attachment.

**PortfolioRisk** = 84

A portfolio risk attachment.

**Project** = 1

A project attachment.

**Risk** = 72

A risk attachment.

**Service** = 47

A service attachment.

**Ticket** = 9

A ticket attachment.

# Namespace TeamDynamix.Api.Auth

## Classes

[AuthRequestBuilder](#)

# Class AuthRequestBuilder

Namespace: [TeamDynamix.Api.Auth](#)

Assembly: TeamDynamix.Api.dll

```
public class AuthRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← AuthRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### AuthRequestBuilder(TdxBaseClient)

Provides authentication-related actions such as admin login, SSO login, and current user retrieval for the TeamDynamix API.

```
public AuthRequestBuilder(TdxBaseClient client)
```

## Parameters

client [TdxBaseClient](#)

## Methods

### AuthenticateAdminAsync(string, string)

Authenticates the client by sending the BEID and WebServicesKey to the authentication endpoint and storing the returned token for subsequent API requests.

```
public Task AuthenticateAdminAsync(string beid, string webServicesKey)
```

## Parameters

beid [string](#)

webServicesKey [string](#)

## Returns

[Task](#)

A [Task](#) representing the asynchronous operation.

## Exceptions

[ArgumentException](#)

Thrown if the authentication token cannot be retrieved or is empty.

## AuthenticateUserAsync(string, string)

Logs in the current session with the specified parameters.

```
public Task AuthenticateUserAsync(string username, string password)
```

## Parameters

username [string](#)

password [string](#)

## Returns

[Task](#)

## Remarks

Expects that you are not using your SSO creds, this would have been set up a TDX admin

## GetCurrentUserAsync()

Gets the current user.

```
public Task<User> GetCurrentUserAsync()
```

Returns

[Task](#) <[User](#)>

The [user](#) instance of the current logged in user.

# Namespace TeamDynamix.Api.Custom Attributes.Models

## Classes

### [CustomAttribute](#)

Represents a custom attribute in the TeamDynamix system.

### [CustomAttributeChoice](#)

Represents a selectable choice for a custom attribute in the TeamDynamix system.

# Class CustomAttribute

Namespace: [TeamDynamix.ApiCustomAttributes.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a custom attribute in the TeamDynamix system.

```
public class CustomAttribute
```

## Inheritance

[object](#) ← CustomAttribute

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AssociatedItemIds

The item types (represented as IDs) associated with the attribute. Nullable.

```
[JsonProperty("AssociatedItemIDs")]
public List<int>? AssociatedItemIds { get; set; }
```

### Property Value

[List](#)<[int](#)>

### Choices

The choices associated with the attribute. Nullable.

```
[JsonProperty("Choices")]
public List<CustomAttributeChoice>? Choices { get; set; }
```

## Property Value

[List](#)<[CustomAttributeChoice](#)>

## ChoicesText

Not used. Nullable.

```
[JsonProperty("ChoicesText")]
public string? ChoicesText { get; set; }
```

## Property Value

[string](#)

## DataType

The data type of the attribute.

```
[JsonProperty("DataType")]
public string? DataType { get; set; }
```

## Property Value

[string](#)

## Description

The description of the attribute. Nullable.

```
[JsonProperty("Description")]
public string? Description { get; set; }
```

## Property Value

[string](#)

## FieldType

The field type of the attribute.

```
[JsonProperty("FieldType")]
public string? FieldType { get; set; }
```

## PropertyValue

[string](#)

## Id

The ID of the attribute.

```
[JsonProperty("ID")]
public int Id { get; set; }
```

## PropertyValue

[int](#)

## IsRequired

The required status of the attribute.

```
[JsonProperty("IsRequired")]
public bool IsRequired { get; set; }
```

## PropertyValue

[bool](#)

## IsUpdatable

The updatable status of the attribute.

```
[JsonProperty("IsUpdatable")]
public bool IsUpdatable { get; set; }
```

## Property Value

[bool](#)

## Name

The name of the attribute.

```
[JsonProperty("Name")]
public string? Name { get; set; }
```

## Property Value

[string](#)

## Order

The explicit sort order of the attribute.

```
[JsonProperty("Order")]
public int Order { get; set; }
```

## Property Value

[int](#)

## SectionId

The ID of the section associated with the attribute.

```
[JsonProperty("SectionID")]
public int SectionId { get; set; }
```

## PropertyValue

[int ↗](#)

## SectionName

The name of the section associated with the attribute. Nullable.

```
[JsonProperty("SectionName")]
public string? SectionName { get; set; }
```

## PropertyValue

[string ↗](#)

## Value

The current value of the attribute. Nullable.

```
[JsonProperty("Value")]
public string? Value { get; set; }
```

## PropertyValue

[string ↗](#)

## ValueText

The current value of the attribute in read-only text format. Nullable.

```
[JsonProperty("ValueText")]
public string? ValueText { get; set; }
```

## PropertyValue

[string ↗](#)

# Class CustomAttributeChoice

Namespace: [TeamDynamix.ApiCustomAttributes.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a selectable choice for a custom attribute in the TeamDynamix system.

```
public class CustomAttributeChoice
```

## Inheritance

[object](#) ← CustomAttributeChoice

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### DateCreated

The created date of the attribute choice.

```
[JsonProperty("DateCreated")]
public DateTime DateCreated { get; set; }
```

### Property Value

[DateTime](#)

### DateModified

The last modified date of the attribute choice.

```
[JsonProperty("DateModified")]
public DateTime DateModified { get; set; }
```

## Property Value

[DateTime](#)

## Id

The ID of the attribute choice.

```
[JsonProperty("ID")]
public int Id { get; set; }
```

## Property Value

[int](#)

## IsActive

The active status of the attribute choice. Editable via the API. Required.

```
[JsonProperty("IsActive")]
public bool IsActive { get; set; }
```

## Property Value

[bool](#)

## Name

The name of the attribute choice. This doubles as the display text. Editable via the API. Required.

```
[JsonProperty("Name")]
public string? Name { get; set; }
```

## Property Value

[string](#)

## Order

The sort order of the attribute choice in the list. Editable via the API. Required.

```
[JsonProperty("Order")]
public int Order { get; set; }
```

Property Value

[int ↗](#)

# Namespace TeamDynamix.Api.Exceptions

## Classes

[UserException](#)

# Class UserException

Namespace: [TeamDynamix.Api.Exceptions](#)

Assembly: TeamDynamix.Api.dll

```
public class UserException : Exception, ISerializable
```

## Inheritance

[object](#) ← [Exception](#) ← UserException

## Implements

[ISerializable](#)

## Inherited Members

[Exception.GetBaseException\(\)](#) , [Exception.GetType\(\)](#) , [Exception.ToString\(\)](#) ,  
[Exception.Data](#) , [Exception.HelpLink](#) , [Exception.HResult](#) , [Exception.InnerException](#) ,  
[Exception.Message](#) , [Exception.Source](#) , [Exception.StackTrace](#) , [Exception.TargetSite](#) ,  
[Exception.SerializeObjectState](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#)

## Constructors

### UserException()

```
public UserException()
```

### UserException(string?)

```
public UserException(string? message)
```

## Parameters

message [string](#)

## UserException(string?, Exception?)

```
public UserException(string? message, Exception? innerException)
```

### Parameters

message [string](#)

innerException [Exception](#)

# Namespace TeamDynamix.Api.Extensions

## Classes

### [TdxClientOptions](#)

Configuration options for the TeamDynamix API client.

This class provides settings that control the behavior of the client, including how it constructs the base API URL, and how it handles throttling responses from the API (HTTP 429 Too Many Requests).

Use these options to customize retry logic, logging, and notification callbacks related to API throttling to improve robustness and observability of API calls. Applies only if not overriding the default HttpClient.

# Class TdxClientOptions

Namespace: [TeamDynamix.Api.Extensions](#)

Assembly: TeamDynamix.Api.dll

Configuration options for the TeamDynamix API client.

This class provides settings that control the behavior of the client, including how it constructs the base API URL, and how it handles throttling responses from the API (HTTP 429 Too Many Requests).

Use these options to customize retry logic, logging, and notification callbacks related to API throttling to improve robustness and observability of API calls. Applies only if not overriding the default HttpClient.

```
public class TdxClientOptions
```

## Inheritance

[object](#) ↗ ← TdxClientOptions

## Inherited Members

[object.Equals\(object\)](#) ↗ , [object.Equals\(object, object\)](#) ↗ , [object.GetHashCode\(\)](#) ↗ ,  
[object.GetType\(\)](#) ↗ , [object.MemberwiseClone\(\)](#) ↗ , [object.ReferenceEquals\(object, object\)](#) ↗ ,  
[object.ToString\(\)](#) ↗

## Properties

### BaseUrl

Optional. The fully formed base API URL for the TeamDynamix API. If this is not set, the URL will be constructed automatically based on the Tenant identifier. Setting this explicitly can be useful for targeting different environments such as your demo environment or versions.

```
public string? BaseUrl { get; set; }
```

Property Value

[string](#) ↗

## EnableThrottleCountdownLogging

If true, enables logging of the countdown timer during throttle wait periods. Useful for debugging or monitoring retry behavior. Default is false.

```
public bool EnableThrottleCountdownLogging { get; set; }
```

Property Value

[bool](#)

## MaxRetries

The maximum number of retry attempts the client will make after receiving a throttling response (HTTP 429). Default value is 3, meaning three retries after the initial failure. Increasing this number allows more retries but may increase overall latency.

```
public int MaxRetries { get; set; }
```

Property Value

[int](#)

## OnThrottleContinue

An optional callback invoked immediately after the throttling wait has completed and before the client retries the request. This can be used to notify that the request will be retried.

```
public Action? OnThrottleContinue { get; set; }
```

Property Value

[Action](#)

## OnThrottleWait

An optional callback invoked before the client waits due to a throttling response (HTTP 429). The [TimeSpan](#) parameter indicates how long the client will wait before retrying.

```
public Action<TimeSpan>? OnThrottleWait { get; set; }
```

## Property Value

[Action](#)<[TimeSpan](#)>

## Remarks

This allows the caller to log or display information about the wait time.

# Namespace TeamDynamix.Api.KnowledgeBase.Models

## Enums

### [ArticleStatus](#)

Describes the different article statuses.

# Enum ArticleStatus

Namespace: [TeamDynamix.Api.KnowledgeBase.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different article statuses.

```
public enum ArticleStatus
```

## Fields

**Approved = 3**

Used for articles that have been approved.

**Archived = 5**

Used for articles that have been archived.

**None = 0**

A "none" status for filtering purposes. Should not be used in normal operations.

**NotSubmitted = 1**

Used for articles that have not been submitted.

**Rejected = 4**

Used for articles that have been rejected.

**Submitted = 2**

Used for articles that have been submitted.

# Namespace TeamDynamix.Api.Locations

## Classes

### [LocationItemRequestBuilder](#)

Provides operations to retrieve and update a specific location in the TeamDynamix API.

### [LocationsRequestBuilder](#)

Contains methods for working with locations.

# Class LocationItemRequestBuilder

Namespace: [TeamDynamix.Api.Locations](#)

Assembly: TeamDynamix.Api.dll

Provides operations to retrieve and update a specific location in the TeamDynamix API.

```
public class LocationItemRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← LocationItemRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### LocationItemRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [LocationItemRequestBuilder](#) class.

```
public LocationItemRequestBuilder(string path, TdxBaseClient client)
```

## Parameters

path [string](#)

The full API path to the location resource.

client [TdxBaseClient](#)

The [TdxBaseClient](#) instance used to send HTTP requests.

# Methods

## GetAsync()

Retrieves detailed information about a specific location. Corresponds to: **GET**

<https://{{tenant}}.teamdynamix.com/TDWebApi/api/locations/{{id}}>.

```
public Task<Location> GetAsync()
```

Returns

[Task](#)<Location>

A fully populated [Location](#) object.

## UpdateAsync(Location)

Updates an existing location with new values. Corresponds to: **PUT**

<https://{{tenant}}.teamdynamix.com/TDWebApi/api/locations/{{id}}>.

```
public Task<Location> UpdateAsync(Location location)
```

Parameters

**location** [Location](#)

The updated [Location](#) object to send.

Returns

[Task](#)<Location>

The updated [Location](#) returned by the API.

# Class LocationsRequestBuilder

Namespace: [TeamDynamix.Api.Locations](#)

Assembly: TeamDynamix.Api.dll

Contains methods for working with locations.

```
public class LocationsRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← LocationsRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### LocationsRequestBuilder(TdxBaseClient)

Initializes a new instance of the [LocationsRequestBuilder](#) class.

```
public LocationsRequestBuilder(TdxBaseClient client)
```

## Parameters

**client** [TdxBaseClient](#)

The [TdxBaseClient](#) instance used to send HTTP requests.

## Properties

## this[string]

Gets a request builder for operations on a specific location. Corresponds to endpoints like:

GET/PUT <https://{{tenant}}.teamdynamix.com/TDWebApi/api/locations/{{id}}>.

```
public LocationItemRequestBuilder this[string] locationId { get; }
```

## Parameters

**locationId** [string](#)

The ID of the location.

## Property Value

[LocationItemRequestBuilder](#)

## Methods

### GetAsync()

Retrieves a list of all active locations. This does not return full location details. Corresponds to: GET <https://{{tenant}}.teamdynamix.com/TDWebApi/api/locations>.

```
public Task<List<Location>> GetAsync()
```

## Returns

[Task](#)<[List](#)<[Location](#)>>

A list of [Location](#) objects with basic metadata.

# Namespace TeamDynamix.Api.Locations.Models

## Classes

### [Location](#)

Represents a physical or virtual location in the TeamDynamix system.

### [LocationRoom](#)

Represents a room at a specific location in the TeamDynamix system.

# Class Location

Namespace: [TeamDynamix.Api.Locations.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a physical or virtual location in the TeamDynamix system.

```
public class Location
```

## Inheritance

[object](#) ← Location

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### Address

The address of the location.

```
[JsonProperty("Address")]
public string? Address { get; set; }
```

### PropertyValue

[string](#)

### AssetsCount

The number of assets associated with the location.

```
[JsonProperty("AssetsCount")]
public int AssetsCount { get; set; }
```

Property Value

[int](#)

## Attributes

The custom attributes associated with the location.

```
[JsonProperty("Attributes")]
public CustomAttribute[]? Attributes { get; set; }
```

Property Value

[CustomAttribute\[\]](#)

## City

The city of the location.

```
[JsonProperty("City")]
public string? City { get; set; }
```

Property Value

[string](#)

## ConfigurationItemsCount

The number of standalone configuration items associated with the location.

```
[JsonProperty("ConfigurationItemCount")]
public int ConfigurationItemCount { get; set; }
```

Property Value

[int](#)

## Country

The country of the location.

```
[JsonProperty("Country")]
public string? Country { get; set; }
```

### Property Value

[string](#)

## CreatedDate

The created date of the location.

```
[JsonProperty("CreatedDate")]
public DateTime CreatedDate { get; set; }
```

### Property Value

[DateTime](#)

## CreatedFullName

The full name of the user who created the location.

```
[JsonProperty("CreatedFullName")]
public string CreatedFullName { get; set; }
```

### Property Value

[string](#)

## CreatedUid

The UID of the user who created the location.

```
[JsonProperty("CreatedUID")]
public Guid CreatedUid { get; set; }
```

## Property Value

[Guid](#)

## Description

The description of the location.

```
[JsonProperty("Description")]
public string? Description { get; set; }
```

## Property Value

[string](#)

## ExternalId

The external ID of the location.

```
[JsonProperty("ExternalID")]
public string? ExternalId { get; set; }
```

## Property Value

[string](#)

## Id

The ID of the location.

```
[JsonProperty("ID")]
public int Id { get; set; }
```

Property Value

[int](#)

## IsActive

The active status of the location.

```
[JsonProperty("IsActive")]
public bool IsActive { get; set; }
```

Property Value

[bool](#)

## IsRoomRequired

Whether the location requires a room when specified for an asset.

```
[JsonProperty("IsRoomRequired")]
public bool IsRoomRequired { get; set; }
```

Property Value

[bool](#)

## Latitude

The latitude of the location.

```
[JsonProperty("Latitude")]
public decimal? Latitude { get; set; }
```

Property Value

[decimal](#)

## Longitude

The longitude of the location.

```
[JsonProperty("Longitude")]
public decimal? Longitude { get; set; }
```

### Property Value

[decimal](#)?

## ModifiedDate

The last modified date of the location.

```
[JsonProperty("ModifiedDate")]
public DateTime ModifiedDate { get; set; }
```

### Property Value

[DateTime](#)

## ModifiedFullName

The full name of the user who last modified the location.

```
[JsonProperty("ModifiedFullName")]
public string ModifiedFullName { get; set; }
```

### Property Value

[string](#)

## ModifiedUid

The UID of the user who last modified the location.

```
[JsonProperty("ModifiedUID")]
public Guid ModifiedUid { get; set; }
```

## Property Value

[Guid](#)

## Name

The name of the location.

```
[JsonProperty("Name")]
public string Name { get; set; }
```

## Property Value

[string](#)

## PostalCode

The postal code of the location.

```
[JsonProperty("PostalCode")]
public string? PostalCode { get; set; }
```

## Property Value

[string](#)

## Rooms

The rooms associated with the location. The custom attributes for each room will not be retrieved.

```
[JsonProperty("Rooms")]
public LocationRoom[]? Rooms { get; set; }
```

## Property Value

[LocationRoom\[\]](#)

## RoomsCount

The number of rooms associated with the location.

```
[JsonProperty("RoomsCount")]
public int RoomsCount { get; set; }
```

## Property Value

[int](#)

## State

The state/province of the location.

```
[JsonProperty("State")]
public string? State { get; set; }
```

## Property Value

[string](#)

## TicketsCount

The number of tickets associated with the location.

```
[JsonProperty("TicketsCount")]
public int TicketsCount { get; set; }
```

## Property Value

[int](#)

## UsersCount

The number of users associated with the location.

```
[JsonProperty("UsersCount")]
public int UsersCount { get; set; }
```

Property Value

[int](#)

# Class LocationRoom

Namespace: [TeamDynamix.Api.Locations.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a room at a specific location in the TeamDynamix system.

```
public class LocationRoom
```

## Inheritance

[object](#) ← LocationRoom

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AssetsCount

Gets or sets the number of assets associated with the location room.

```
public int AssetsCount { get; set; }
```

Property Value

[int](#)

### Attributes

Gets or sets the custom attributes associated with the location room. This field is editable and nullable.

```
public List<CustomAttribute>? Attributes { get; set; }
```

## Property Value

[List](#)<[CustomAttribute](#)>

## Capacity

Gets or sets the capacity of the location room. This field is editable and nullable.

```
public int? Capacity { get; set; }
```

## Property Value

[int](#)?

## ConfigurationItemsCount

Gets or sets the number of configuration items associated with the location room.

```
public int ConfigurationItemCount { get; set; }
```

## Property Value

[int](#)

## CreatedDate

Gets or sets the date the location room was created.

```
public DateTime CreatedDate { get; set; }
```

## Property Value

[DateTime](#)

## CreatedFullName

Gets or sets the full name of the user who created the location room.

```
public string CreatedFullName { get; set; }
```

Property Value

[string](#)

## CreatedUID

Gets or sets the UID of the user who created the location room.

```
public Guid CreatedUID { get; set; }
```

Property Value

[Guid](#)

## Description

Gets or sets the description of the location room. This field is editable and nullable.

```
public string? Description { get; set; }
```

Property Value

[string](#)

## ExternalID

Gets or sets the external ID of the location room. This field is editable and nullable.

```
public string? ExternalID { get; set; }
```

Property Value

[string](#)

## Floor

Gets or sets the floor information associated with the location room. This field is editable and nullable.

```
public string? Floor { get; set; }
```

Property Value

[string](#)

## ID

Gets or sets the ID of the location room.

```
public int ID { get; set; }
```

Property Value

[int](#)

## Name

Gets or sets the name of the location room. This field is required and editable through the API.

```
public string Name { get; set; }
```

Property Value

[string](#)

## TicketsCount

Gets or sets the number of tickets associated with the location room.

```
public int TicketsCount { get; set; }
```

Property Value

[int](#)

## UsersCount

Gets or sets the number of users associated with the location room.

```
public int UsersCount { get; set; }
```

Property Value

[int](#)

# Namespace TeamDynamix.Api.Patching

## Classes

[PatchBuilder<T>](#)

[PatchOperation](#)

# Class PatchBuilder<T>

Namespace: [TeamDynamix.Api.Patching](#)

Assembly: TeamDynamix.Api.dll

```
public class PatchBuilder<T>
```

## Type Parameters

T

## Inheritance

[object](#) ← PatchBuilder<T>

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Methods

### Add(string, object)

Adds or replaces a value at the specified path.

```
public PatchBuilder<T> Add(string path, object value)
```

## Parameters

path [string](#)

value [object](#)

## Returns

[PatchBuilder](#)<T>

## Build()

Builds the patch document to send with a PATCH request.

```
public List<PatchOperation> Build()
```

Returns

[List](#)<[PatchOperation](#)>

## Remove(string)

Removes a value at the specified path.

```
public PatchBuilder<T> Remove(string path)
```

Parameters

path [string](#)

Returns

[PatchBuilder](#)<T>

# Class PatchOperation

Namespace: [TeamDynamix.Api.Patching](#)

Assembly: TeamDynamix.Api.dll

```
public class PatchOperation
```

## Inheritance

[object](#) ← PatchOperation

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### Op

The type of operation to perform (e.g., "add", "remove").

```
[JsonProperty("op")]
public string Op { get; set; }
```

### Property Value

[string](#)

### Path

The path to the property being modified (e.g., "/name" or "/attributes/1234").

```
[JsonProperty("path")]
public string Path { get; set; }
```

### Property Value

[string](#)

## Value

The value to apply. Not required for remove operations.

```
[JsonProperty("value", NullValueHandling = NullValueHandling.Ignore)]  
public object? Value { get; set; }
```

Property Value

[object](#)

# Namespace TeamDynamix.Api.People

## Classes

### [PeopleRequestBuilder](#)

Provides methods for interacting with people in the TeamDynamix system. This includes searching for people or retrieving them by UID.

### [PersonByIdRequestBuilder](#)

Provides functionality to retrieve a person by their unique identifier (UID) from the TeamDynamix people database.

### [PersonLookupRequestBuilder](#)

### [PersonSearchRequestBuilder](#)

Provides functionality to search for people in the TeamDynamix people database based on various search criteria.

# Class PeopleRequestBuilder

Namespace: [TeamDynamix.Api.People](#)

Assembly: TeamDynamix.Api.dll

Provides methods for interacting with people in the TeamDynamix system. This includes searching for people or retrieving them by UID.

```
public class PeopleRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← PeopleRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### PeopleRequestBuilder(TdxBaseClient)

Initializes a new instance of the [PeopleRequestBuilder](#) class.

```
public PeopleRequestBuilder(TdxBaseClient client)
```

## Parameters

### client [TdxBaseClient](#)

The base TDX API client used for sending requests.

## Properties

## this[string]

Gets a request builder to access a specific person by their UID.

```
public PersonByIdRequestBuilder this[string uid] { get; }
```

### Parameters

**uid** [string](#)

The UID of the person to retrieve.

### Property Value

[PersonByIdRequestBuilder](#)

A [PersonByIdRequestBuilder](#) for the specified person.

## Search

Gets a request builder used to search for people using various filters.

```
public PersonSearchRequestBuilder Search { get; }
```

### Property Value

[PersonSearchRequestBuilder](#)

# Class PersonByIdRequestBuilder

Namespace: [TeamDynamix.Api.People](#)

Assembly: TeamDynamix.Api.dll

Provides functionality to retrieve a person by their unique identifier (UID) from the TeamDynamix people database.

```
public class PersonByIdRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← PersonByIdRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### PersonByIdRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [PersonByIdRequestBuilder](#) class.

```
public PersonByIdRequestBuilder(string path, TdxBaseClient client)
```

## Parameters

**path** [string](#)

The relative API path to the specific person.

**client** [TdxBaseClient](#)

The TDX API client used to send the request.

# Methods

## GetAsync()

Sends a GET request to retrieve the person identified by the specified UID.

```
public Task<User> GetAsync()
```

Returns

[Task](#)<User>

A [User](#) object representing the person.

# Class PersonLookupRequestBuilder

Namespace: [TeamDynamix.Api.People](#)

Assembly: TeamDynamix.Api.dll

```
public class PersonLookupRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← PersonLookupRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### PersonLookupRequestBuilder(TdxBaseClient, string)

```
public PersonLookupRequestBuilder(TdxBaseClient client, string path)
```

## Parameters

client [TdxBaseClient](#)

path [string](#)

# Class PersonSearchRequestBuilder

Namespace: [TeamDynamix.Api.People](#)

Assembly: TeamDynamix.Api.dll

Provides functionality to search for people in the TeamDynamix people database based on various search criteria.

```
public class PersonSearchRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← PersonSearchRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### PersonSearchRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [PersonSearchRequestBuilder](#) class.

```
public PersonSearchRequestBuilder(string path, TdxBaseClient client)
```

## Parameters

### path [string](#)

The relative API path for the search endpoint.

### client [TdxBaseClient](#)

The TDX API client used to send the request.

# Methods

## GetAsync()

Executes the search with the provided criteria and returns a list of matching users.

```
public Task<List<User>> GetAsync()
```

Returns

[Task](#)<[List](#)<[User](#)>>

A list of [User](#) objects that match the search criteria.

Exceptions

[InvalidOperationException](#)

Thrown when no search criteria have been provided.

[UserException](#)

Thrown when no matching users are found.

## With(string, string)

Adds a custom search criterion to the search criteria.

```
public PersonSearchRequestBuilder With(string key, string value)
```

Parameters

**key** [string](#)

The key of the search criterion.

**value** [string](#)

The value of the search criterion.

Returns

## [PersonSearchRequestBuilder](#)

The current [PersonSearchRequestBuilder](#) instance.

## Exceptions

### [ArgumentException](#)

Thrown when `key` or `value` is null or empty.

## WithEmail(string)

Adds the email to the search criteria.

```
public PersonSearchRequestBuilder WithEmail(string email)
```

## Parameters

### `email` [string](#)

The email address of the person.

## Returns

## [PersonSearchRequestBuilder](#)

The current [PersonSearchRequestBuilder](#) instance.

## WithExternalId(string)

Adds the external ID to the search criteria.

```
public PersonSearchRequestBuilder WithExternalId(string externalId)
```

## Parameters

### `externalId` [string](#)

The external ID of the person.

Returns

## [PersonSearchRequestBuilder](#)

The current [PersonSearchRequestBuilder](#) instance.

Exceptions

### [ArgumentException](#)

Thrown when `externalId` is null or empty.

## WithFirstName(string)

Adds the first name to the search criteria.

```
public PersonSearchRequestBuilder WithFirstName(string firstName)
```

Parameters

### `firstName` [string](#)

The first name of the person.

Returns

## [PersonSearchRequestBuilder](#)

The current [PersonSearchRequestBuilder](#) instance.

## WithLastName(string)

Adds the last name to the search criteria.

```
public PersonSearchRequestBuilder WithLastName(string lastName)
```

Parameters

### `lastName` [string](#)

The last name of the person.

Returns

[PersonSearchRequestBuilder](#)

The current [PersonSearchRequestBuilder](#) instance.

## WithUsername(string)

Adds the username to the search criteria.

```
public PersonSearchRequestBuilder WithUsername(string username)
```

Parameters

username [string](#)

The username of the person.

Returns

[PersonSearchRequestBuilder](#)

The current [PersonSearchRequestBuilder](#) instance.

Exceptions

[ArgumentException](#)

Thrown when `username` is null or empty.

# Namespace TeamDynamix.Api.People.Models

## Classes

### [User](#)

Represents a user in the TeamDynamix system.

## Enums

### [UserType](#)

Describes the different types of users and customers tracked within TeamDynamix.

# Class User

Namespace: [TeamDynamix.Api.People.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a user in the TeamDynamix system.

```
public class User
```

## Inheritance

[object](#) ← User

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AboutMe

The "About Me" information associated with the user. Editable via API. Nullable.

```
[JsonProperty("AboutMe")]
public string? AboutMe { get; set; }
```

### PropertyValue

[string](#)

### AlertEmail

The alert email address of the user where system notifications are delivered. Editable via API. Nullable.

```
[JsonProperty("AlertEmail")]
```

```
public string? AlertEmail { get; set; }
```

Property Value

[string](#)

## AlternateEmail

The alternate email address of the user. Editable via API. Nullable.

```
[JsonProperty("AlternateEmail")]
public string? AlternateEmail { get; set; }
```

Property Value

[string](#)

## AlternateId

The alternate ID of the user. Editable via API. Nullable.

```
[JsonProperty("AlternateID")]
public string? AlternateId { get; set; }
```

Property Value

[string](#)

## Applications

The system-defined (non-platform) applications associated with the user. Editable via API. Nullable.

```
[JsonProperty("Applications")]
public List<string>? Applications { get; set; }
```

## Property Value

[List](#)<[string](#)>

## ApplyTechnicianSignatureToReplies

Determines whether the technician signature is applied to the replies for all tickets and ticket tasks. Editable via API.

```
[JsonProperty("ApplyTechnicianSignatureToReplies")]
public bool ApplyTechnicianSignatureToReplies { get; set; }
```

## Property Value

[bool](#)

## ApplyTechnicianSignatureToUpdatesAndComments

Determines whether the technician signature is applied to the updates and comments for all tickets and ticket tasks. Editable via API.

```
[JsonProperty("ApplyTechnicianSignatureToUpdatesAndComments")]
public bool ApplyTechnicianSignatureToUpdatesAndComments { get; set; }
```

## Property Value

[bool](#)

## Attributes

The custom attributes associated with the user. Editable via API. Nullable.

```
[JsonProperty("Attributes")]
public List<CustomAttribute>? Attributes { get; set; }
```

## Property Value

## AuthenticationProviderID

The ID of the authentication provider the new user will use for authentication. Editable via API. Nullable.

```
[JsonProperty("AuthenticationProviderID")]
public int? AuthenticationProviderID { get; set; }
```

Property Value

[int](#)?

## AuthenticationUserName

The authentication username of the user, used for authenticating with non-TeamDynamix authentication types. Editable via API. Nullable.

```
[JsonProperty("AuthenticationUserName")]
public string? AuthenticationUserName { get; set; }
```

Property Value

[string](#)

## Beld

The UID of the organization associated with the user.

```
[JsonProperty("BEID")]
public string? BeId { get; set; }
```

Property Value

[string](#)

## BeIdInt

The integer ID of the organization associated with the user.

```
[JsonProperty("BEIDInt")]
public int BeIdInt { get; set; }
```

## Property Value

[int](#)

## Company

The company of the user. Editable via API. Required.

```
[JsonProperty("Company")]
public string? Company { get; set; }
```

## Property Value

[string](#)

## CostRate

The cost rate of the user. Editable via API.

```
[JsonProperty("CostRate")]
public double CostRate { get; set; }
```

## Property Value

[double](#)

## DefaultAccountId

The ID of the default account/department associated with the user. Editable via API.

```
[JsonProperty("DefaultAccountId")]
public int DefaultAccountId { get; set; }
```

Property Value

[int](#)

## DefaultAccountName

The name of the default account/department associated with the user.

```
[JsonProperty("DefaultAccountName")]
public string? DefaultAccountName { get; set; }
```

Property Value

[string](#)

## DefaultPriorityID

The ID of the default priority associated with the user. Editable via API.

```
[JsonProperty("DefaultPriorityID")]
public int DefaultPriorityID { get; set; }
```

Property Value

[int](#)

## DefaultPriorityName

The name of the default priority associated with the user.

```
[JsonProperty("DefaultPriorityName")]
public string? DefaultPriorityName { get; set; }
```

## Property Value

[string](#) ↗

## DefaultRate

The default bill rate of the user. Editable via API.

```
[JsonProperty("DefaultRate")]
public double DefaultRate { get; set; }
```

## Property Value

[double](#) ↗

## EndDate

The date after which the user is no longer available for scheduling and no longer required to log time. Editable via API.

```
[JsonProperty("EndDate")]
public DateTime EndDate { get; set; }
```

## Property Value

[DateTime](#) ↗

## ExternalId

The organizational ID of the user. Editable via API. Nullable.

```
[JsonProperty("ExternalID")]
public string? ExternalId { get; set; }
```

## Property Value

[string](#) ↗

## Fax

The fax number of the user. Editable via API. Nullable.

```
[JsonProperty("Fax")]
public string? Fax { get; set; }
```

## PropertyValue

[string](#)

## FirstName

The first name of the user. Editable via API. Required.

```
[JsonProperty("FirstName")]
public string? FirstName { get; set; }
```

## PropertyValue

[string](#)

## FullName

The full name of the user.

```
[JsonProperty("FullName")]
public string? FullName { get; set; }
```

## PropertyValue

[string](#)

## GroupIds

The IDs of the groups associated with the user. Nullable.

```
[JsonProperty("GroupIDs")]
public List<int>? GroupIDs { get; set; }
```

Property Value

[List](#)<[int](#)>

## HomeAddress

The home address of the user. Editable via API. Nullable.

```
[JsonProperty("HomeAddress")]
public string? HomeAddress { get; set; }
```

Property Value

[string](#)

## HomeCity

The home city of the user. Editable via API. Nullable.

```
[JsonProperty("HomeCity")]
public string? HomeCity { get; set; }
```

Property Value

[string](#)

## HomeCountry

The home country of the user. Editable via API. Nullable.

```
[JsonProperty("HomeCountry")]
```

```
public string? HomeCountry { get; set; }
```

Property Value

[string](#)

## HomePhone

The home phone number of the user. Editable via API. Nullable.

```
[JsonProperty("HomePhone")]
public string? HomePhone { get; set; }
```

Property Value

[string](#)

## HomeState

The home state abbreviation of the user. Editable via API. Nullable.

```
[JsonProperty("HomeState")]
public string? HomeState { get; set; }
```

Property Value

[string](#)

## HomeZip

The home zip code of the user. Editable via API. Nullable.

```
[JsonProperty("HomeZip")]
public string? HomeZip { get; set; }
```

Property Value

[string](#)

## IMHandle

The Instant Messenger (IM) username/handle associated with the user. Editable via API. Nullable.

```
[JsonProperty("IMHandle")]
public string? IMHandle { get; set; }
```

Property Value

[string](#)

## IMProvider

The Instant Messenger (IM) provider associated with the user. Editable via API. Nullable.

```
[JsonProperty("IMProvider")]
public string? IMProvider { get; set; }
```

Property Value

[string](#)

## IsActive

The active status of the user. Editable via API.

```
[JsonProperty("IsActive")]
public bool IsActive { get; set; }
```

Property Value

[bool](#)

## IsCapacityManaged

Whether the user's capacity is managed, meaning they can have capacity and appear on capacity/availability reports. Editable via API.

```
[JsonProperty("IsCapacityManaged")]
public bool IsCapacityManaged { get; set; }
```

Property Value

[bool](#) ↗

## IsConfidential

The confidential status of the user. Editable via API.

```
[JsonProperty("IsConfidential")]
public bool IsConfidential { get; set; }
```

Property Value

[bool](#) ↗

## IsEmployee

The employee status of the user. Editable via API.

```
[JsonProperty("IsEmployee")]
public bool IsEmployee { get; set; }
```

Property Value

[bool](#) ↗

## LastName

The last name of the user. Editable via API. Required.

```
[JsonProperty("LastName")]
public string? LastName { get; set; }
```

## PropertyValue

[string](#)

## LocationId

The ID of the location associated with the user. Editable via API.

```
[JsonProperty("LocationID")]
public int LocationId { get; set; }
```

## PropertyValue

[int](#)

## LocationName

The name of the location associated with the user.

```
[JsonProperty("LocationName")]
public string? LocationName { get; set; }
```

## PropertyValue

[string](#)

## LocationRoomId

The ID of the location room associated with the user. Editable via API.

```
[JsonProperty("LocationRoomID")]
public int LocationRoomId { get; set; }
```

Property Value

[int](#)

## LocationRoomName

The name of the location room associated with the user.

```
[JsonProperty("LocationRoomName")]
public string? LocationRoomName { get; set; }
```

Property Value

[string](#)

## MiddleName

The middle name of the user. Editable via API. Nullable.

```
[JsonProperty("MiddleName")]
public string? MiddleName { get; set; }
```

Property Value

[string](#)

## MobilePhone

The mobile phone number of the user. Editable via API. Nullable.

```
[JsonProperty("MobilePhone")]
public string? MobilePhone { get; set; }
```

Property Value

[string](#)

## Nickname

The nickname of the user. Editable via API. Nullable.

```
[JsonProperty("Nickname")]
public string? Nickname { get; set; }
```

### Property Value

[string](#)

## OrgApplications

The organizationally-defined (platform) applications associated with the user. Editable via API. Nullable.

```
[JsonProperty("OrgApplications")]
public List<UserApplication>? OrgApplications { get; set; }
```

### Property Value

[List](#)<[UserApplication](#)>

## OtherPhone

The other phone number of the user. Editable via API. Nullable.

```
[JsonProperty("OtherPhone")]
public string? OtherPhone { get; set; }
```

### Property Value

[string](#)

## Pager

The pager number of the user. Editable via API. Nullable.

```
[JsonProperty("Pager")]
public string? Pager { get; set; }
```

## Property Value

[string](#)

## Permissions

The global security role permissions associated with the user. Nullable.

```
[JsonProperty("Permissions")]
public List<string>? Permissions { get; set; }
```

## Property Value

[List](#)<[string](#)>

## PrimaryClientPortalApplicationId

The ID of the primary client portal application associated with the user. Editable via API. Nullable.

```
[JsonProperty("PrimaryClientPortalApplicationID")]
public int? PrimaryClientPortalApplicationId { get; set; }
```

## Property Value

[int](#)?

## PrimaryEmail

The primary email address of the user. Editable via API. Required.

```
[JsonProperty("PrimaryEmail")]
public required string PrimaryEmail { get; set; }
```

Property Value

[string](#)

## PrimaryPhone

The primary phone number of the user. Editable via API. Nullable.

```
[JsonProperty("PrimaryPhone")]
public string? PrimaryPhone { get; set; }
```

Property Value

[string](#)

## ProfileImageFileName

The profile image file name of the user.

```
[JsonProperty("ProfileImageFileName")]
public string? ProfileImageFileName { get; set; }
```

Property Value

[string](#)

## Referenceld

The integer ID of the user.

```
[JsonProperty("ReferenceID")]
public int ReferenceId { get; set; }
```

Property Value

[int](#)

## ReportTimeAfterDate

The date after which the user should start reporting time, also governs capacity calculations. Editable via API.

```
[JsonProperty("ReportTimeAfterDate")]
public DateTime ReportTimeAfterDate { get; set; }
```

Property Value

[DateTime](#)

## ReportsToFullName

The full name of the person who the user reports to.

```
[JsonProperty("ReportsToFullName")]
public string? ReportsToFullName { get; set; }
```

Property Value

[string](#)

## ReportsToUid

The UID of the person who the user reports to. Editable via API. Nullable.

```
[JsonProperty("ReportsToUID")]
public string? ReportsToUid { get; set; }
```

Property Value

[string](#)

## ResourcePoolID

The ID of the resource pool associated with the user. Editable via API.

```
[JsonProperty("ResourcePoolID")]
public int ResourcePoolID { get; set; }
```

## Property Value

[int](#)

## ResourcePoolName

The name of the resource pool associated with the user.

```
[JsonProperty("ResourcePoolName")]
public string? ResourcePoolName { get; set; }
```

## Property Value

[string](#)

## Salutation

The salutation of the user. Editable via API. Nullable.

```
[JsonProperty("Salutation")]
public string? Salutation { get; set; }
```

## Property Value

[string](#)

## SecurityRoleId

The UID of the global security role associated with the user. Editable via API.

```
[JsonProperty("SecurityRoleID")]
public string? SecurityRoleId { get; set; }
```

Property Value

[string](#) ↗

## SecurityRoleName

The name of the global security role associated with the user.

```
[JsonProperty("SecurityRoleName")]
public string? SecurityRoleName { get; set; }
```

Property Value

[string](#) ↗

## ShouldReportTime

Whether the user should report time. Editable via API.

```
[JsonProperty("ShouldReportTime")]
public bool ShouldReportTime { get; set; }
```

Property Value

[bool](#) ↗

## TechnicianSignature

The technician signature associated with the user. Editable via API. Nullable.

```
[JsonProperty("TechnicianSignature")]
public string? TechnicianSignature { get; set; }
```

Property Value

[string](#) ↗

## Title

The title of the user. Editable via API. Nullable.

```
[JsonProperty("Title")]
public string? Title { get; set; }
```

### Property Value

[string](#)

## TypeId

The type of the user.

```
[JsonProperty("TypeID")]
public UserType TypeId { get; set; }
```

### Property Value

[UserType](#)

## TzName

The name of the time zone associated with the user.

```
[JsonProperty("TZName")]
public string? TzName { get; set; }
```

### Property Value

[string](#)

## Tzid

The ID of the time zone associated with the user. Editable via API.

```
[JsonProperty("TZID")]
public int Tzid { get; set; }
```

## Property Value

[int](#)

## Uid

The UID of the user.

```
[JsonProperty("UID")]
public required string Uid { get; set; }
```

## Property Value

[string](#)

## UserName

The username of the user.

```
[JsonProperty("UserName")]
public required string UserName { get; set; }
```

## Property Value

[string](#)

## WorkAddress

The work address of the user. Editable via API. Nullable.

```
[JsonProperty("WorkAddress")]
public string? WorkAddress { get; set; }
```

Property Value

[string](#)

## WorkCity

The work city of the user. Editable via API. Nullable.

```
[JsonProperty("WorkCity")]
public string? WorkCity { get; set; }
```

Property Value

[string](#)

## WorkCountry

The work country of the user. Editable via API. Nullable.

```
[JsonProperty("WorkCountry")]
public string? WorkCountry { get; set; }
```

Property Value

[string](#)

## WorkPhone

The work phone number of the user. Editable via API. Nullable.

```
[JsonProperty("WorkPhone")]
public string? WorkPhone { get; set; }
```

Property Value

[string](#)

## WorkState

The work state abbreviation of the user. Editable via API. Nullable.

```
[JsonProperty("WorkState")]
public string? WorkState { get; set; }
```

Property Value

[string](#) ↗

## WorkZip

The work zip code of the user. Editable via API. Nullable.

```
[JsonProperty("WorkZip")]
public string? WorkZip { get; set; }
```

Property Value

[string](#) ↗

## WorkableHours

The number of workable hours in a work day for the user. Editable via API.

```
[JsonProperty("WorkableHours")]
public double WorkableHours { get; set; }
```

Property Value

[double](#) ↗

# Enum UserType

Namespace: [TeamDynamix.Api.People.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different types of users and customers tracked within TeamDynamix.

```
public enum UserType
```

## Fields

**Customer = 2**

Indicates that the user is classified as a customer, which means that they cannot log in to TeamDynamix.

**None = 0**

Indicates that the type of the user could not be determined. Should not be used in normal operations.

**ResourcePlaceholder = 8**

Indicates that the user is classified as a resource placeholder. These users act as a placeholder for actual users when planning out projects without knowing exactly who will be acting as the resource in question.

**ServiceAccount = 9**

Indicates that the user is classified as a service account. These users will be able to authenticate and use the API, but they will not be able to log in to TDNext, TDMobile, TDAdmin, or TDClient.

**User = 1**

Indicates that the user is classified as a full TeamDynamix user.

# Namespace TeamDynamix.Api.Shared Models

## Classes

### [ResourceItem](#)

A single person, group, or account/department.

# Class ResourceItem

Namespace: [TeamDynamix.Api.SharedModels](#)

Assembly: TeamDynamix.Api.dll

A single person, group, or account/department.

```
public class ResourceItem
```

## Inheritance

[object](#) ← ResourceItem

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### GroupId

Extension for internal usage not an included property

```
public string? GroupId { get; set; }
```

### Property Value

[string](#)

### Initials

The initials to be displayed if no profile image is specified for the resource.

```
public string? Initials { get; set; }
```

### Property Value

[string](#)

## ItemRole

The role the resource has on the associated item.

```
public string? ItemRole { get; set; }
```

### Property Value

[string](#)

## Name

The name of the resource.

```
public required string Name { get; set; }
```

### Property Value

[string](#)

## ProfileImageFileName

The profile image file name of the resource.

```
public string? ProfileImageFileName { get; set; }
```

### Property Value

[string](#)

## RefValue

The integer ID of the resource.

```
public int RefValue { get; set; }
```

Property Value

[int](#)

## Value

The value of the resource.

```
public string? Value { get; set; }
```

Property Value

[string](#)

# Namespace TeamDynamix.Api.Tickets

## Classes

### [TicketCreateObjectRequestBuilder](#)

Provides a builder for creating tickets using a fully constructed [Ticket](#) object.

### [TicketCreateRequestBuilder](#)

Provides a fluent interface for constructing and submitting a new ticket using discrete fields.

### [TicketItemRequestBuilder](#)

Provides operations to retrieve a specific ticket by ID.

### [TicketsRequestBuilder](#)

Provides operations to manage and create tickets within a specific ticketing application in the TeamDynamix platform.

# Class TicketCreateObjectRequestBuilder

Namespace: [TeamDynamix.Api.Tickets](#)

Assembly: TeamDynamix.Api.dll

Provides a builder for creating tickets using a fully constructed [Ticket](#) object.

```
public class TicketCreateObjectRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← [TicketCreateObjectRequestBuilder](#)

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### TicketCreateObjectRequestBuilder(string, TdxBaseClient, Ticket)

Initializes a new instance of the [TicketCreateObjectRequestBuilder](#) class.

```
public TicketCreateObjectRequestBuilder(string path, TdxBaseClient client,  
Ticket ticket)
```

## Parameters

**path** [string](#)

The API endpoint path.

**client** [TdxBaseClient](#)

The client used to send the request.

## `ticket` [Ticket](#)

The ticket to be created.

# Methods

## `PostAsync()`

Submits the ticket creation request and returns the created [Ticket](#).

```
public Task<Ticket> PostAsync()
```

Returns

### [Task](#)<[Ticket](#)>

## `WithOptions(Action<TicketCreateOptions>)`

Configures additional query parameters for the creation request.

```
public TicketCreateObjectRequestBuilder WithOptions(Action<TicketCreateOptions> configure)
```

Parameters

### `configure` [Action](#)<[TicketCreateOptions](#)>

An action to configure [TicketCreateOptions](#).

Returns

### [TicketCreateObjectRequestBuilder](#)

# Class TicketCreateRequestBuilder

Namespace: [TeamDynamix.Api.Tickets](#)

Assembly: TeamDynamix.Api.dll

Provides a fluent interface for constructing and submitting a new ticket using discrete fields.

```
public class TicketCreateRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← [TicketCreateRequestBuilder](#)

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### TicketCreateRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [TicketCreateRequestBuilder](#) class.

```
public TicketCreateRequestBuilder(string path, TdxBaseClient client)
```

#### Parameters

**path** [string](#)

The base path for the ticket creation API endpoint.

**client** [TdxBaseClient](#)

The API client used to make requests.

# Methods

## PostAsync()

Submits the ticket creation request and returns the created [Ticket](#).

```
public Task<Ticket> PostAsync()
```

Returns

[Task](#)<[Ticket](#)>

## ValidateRequiredFields()

Validates that all required fields for ticket creation are present.

```
public void ValidateRequiredFields()
```

Exceptions

[InvalidOperationException](#)

Thrown when a required field is missing.

## WithAccountId(string)

Sets the account ID associated with the ticket.

```
public TicketCreateRequestBuilder WithAccountId(string accountId)
```

Parameters

accountId [string](#)

Returns

[TicketCreateRequestBuilder](#)

## WithDescription(string)

Sets the description for the ticket.

```
public TicketCreateRequestBuilder WithDescription(string description)
```

Parameters

description [string](#)

Returns

[TicketCreateRequestBuilder](#)

## WithOptions(Action<TicketCreateOptions>)

Applies additional options for the ticket creation request.

```
public TicketCreateRequestBuilder WithOptions(Action<TicketCreateOptions> configure)
```

Parameters

configure [Action](#)<[TicketCreateOptions](#)>

An action to configure the [TicketCreateOptions](#).

Returns

[TicketCreateRequestBuilder](#)

## WithRequestor(string)

Sets the UID of the requestor.

```
public TicketCreateRequestBuilder WithRequestor(string uid)
```

Parameters

`uid` [string](#)

Returns

[TicketCreateRequestBuilder](#)

## WithResponsible(string)

Sets the UID of the responsible person.

```
public TicketCreateRequestBuilder WithResponsible(string uid)
```

Parameters

`uid` [string](#)

Returns

[TicketCreateRequestBuilder](#)

## WithResponsibleGroup(string)

Sets Id of the responsible group associated with the ticket.

```
public TicketCreateRequestBuilder WithResponsibleGroup(string id)
```

Parameters

`id` [string](#)

Id Of the responsible group

Returns

[TicketCreateRequestBuilder](#)

## WithStatusId(string)

Sets the status ID of the ticket.

```
public TicketCreateRequestBuilder WithStatusId(string statusID)
```

Parameters

statusID [string](#)

Returns

[TicketCreateRequestBuilder](#)

## WithTitle(string)

Sets the title of the ticket.

```
public TicketCreateRequestBuilder WithTitle(string title)
```

Parameters

title [string](#)

Returns

[TicketCreateRequestBuilder](#)

## WithTypeId(string)

Sets the type ID of the ticket.

```
public TicketCreateRequestBuilder WithTypeId(string typeId)
```

Parameters

typeId [string](#)

Returns

## [TicketCreateRequestBuilder](#)

# Class TicketItemRequestBuilder

Namespace: [TeamDynamix.Api.Tickets](#)

Assembly: TeamDynamix.Api.dll

Provides operations to retrieve a specific ticket by ID.

```
public class TicketItemRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← TicketItemRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### TicketItemRequestBuilder(string, TdxBaseClient)

Initializes a new instance of the [TicketItemRequestBuilder](#) class.

```
public TicketItemRequestBuilder(string path, TdxBaseClient tdxBaseClient)
```

## Parameters

**path** [string](#)

The API path to the specific ticket.

**tdxBaseClient** [TdxBaseClient](#)

The client used to send requests.

# Methods

## GetAsync()

Retrieves the ticket identified by the path.

```
public Task<Ticket> GetAsync()
```

Returns

[Task](#)<[Ticket](#)>

A [Ticket](#) object representing the ticket.

# Class TicketsRequestBuilder

Namespace: [TeamDynamix.Api.Tickets](#)

Assembly: TeamDynamix.Api.dll

Provides operations to manage and create tickets within a specific ticketing application in the TeamDynamix platform.

```
public class TicketsRequestBuilder : BaseRequestBuilder
```

## Inheritance

[object](#) ← [BaseRequestBuilder](#) ← TicketsRequestBuilder

## Inherited Members

[BaseRequestBuilder.Client](#) , [BaseRequestBuilder.Path](#) ,  
[BaseRequestBuilder.CreateRequest\(HttpMethod, string, object\)](#) ,  
[BaseRequestBuilder.SendAsync<T>\(HttpRequestMessage\)](#) ,  
[BaseRequestBuilder.SendAsync\(HttpRequestMessage\)](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### TicketsRequestBuilder(TdxBaseClient, string)

Initializes a new instance of the [TicketsRequestBuilder](#) class.

```
public TicketsRequestBuilder(TdxBaseClient client, string appId)
```

## Parameters

**client** [TdxBaseClient](#)

The TDX base client used to send HTTP requests.

**appId** [string](#)

The ID of the ticketing application.

# Exceptions

## [ArgumentException](#)

Thrown if `appId` is null or empty.

# Properties

## AppId

Gets the ID of the ticketing application associated with this request builder.

```
public string AppId { get; }
```

## Property Value

[string](#)

## this[string]

Provides operations for a specific ticket by its ID.

```
public TicketItemRequestBuilder this[string ticketId] { get; }
```

## Parameters

`ticketId` [string](#)

The ID of the ticket to manage.

## Property Value

[TicketItemRequestBuilder](#)

A [TicketItemRequestBuilder](#) for the specified ticket.

# Methods

## Create()

Provides operations for creating new tickets using request parameters.

```
public TicketCreateRequestBuilder Create()
```

Returns

[TicketCreateRequestBuilder](#)

A [TicketCreateRequestBuilder](#) for ticket creation.

## Create(Ticket)

Provides operations for creating a new ticket using a [Ticket](#) object.

```
public TicketCreateObjectRequestBuilder Create(Ticket ticket)
```

Parameters

**ticket** [Ticket](#)

The ticket to create.

Returns

[TicketCreateObjectRequestBuilder](#)

A [TicketCreateObjectRequestBuilder](#) to execute the ticket creation.

# Namespace TeamDynamix.Api.Tickets.Models

## Classes

### [Ticket](#)

Represents a ticket with all associated details in TeamDynamix.

### [TicketTask](#)

A ticket task or activity.

## Enums

### [ConflictType](#)

Describes the different types of conflicts that can be detected for a scheduled maintenance activity for a CI.

### [StatusClass](#)

Describes the different classes of statuses.

### [TicketClass](#)

Describes the different classifications of tickets.

### [TicketTaskType](#)

Describes the different types of items that can be backended by ticket tasks.

# Enum ConflictType

Namespace: [TeamDynamix.Api.Tickets.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different types of conflicts that can be detected for a scheduled maintenance activity for a CI.

[Flags]

```
public enum ConflictType
```

## Fields

**DuringBlackoutWindow = 2**

Indicates that an activity takes place during a blackout window.

**ExistingActivity = 4**

Indicates that an activity conflicts with a pre-existing activity scheduled for the CI.

**ExistingChildActivity = 16**

Indicates that an activity conflicts with a pre-existing activity on one of the CI's operational children.

**ExistingParentActivity = 32**

Indicates that an activity conflicts with a pre-existing activity on one of the CI's operational parents.

**None = 0**

Indicates that no conflicts were detected.

**OutsideChildMaintenanceWindow = 8**

Indicates that an activity takes place outside the maintenance window of one or more of the CI's operational children.

**OutsideMaintenanceWindow = 1**

Indicates that an activity takes place outside of the CI's maintenance window.

## Remarks

This is an enumeration that can be treated as a bit field (i.e. a series of flags), and thus the fields can be combined.

# Enum StatusClass

Namespace: [TeamDynamix.Api.Tickets.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different classes of statuses.

```
public enum StatusClass
```

## Fields

**Cancelled = 4**

Used for items that are cancelled.

**Completed = 3**

Used for statuses that indicate completion such as "Closed" or "Closed and Approved".

**InProcess = 2**

Used for statuses that are somewhere in the pipeline, but not yet completed.

**New = 1**

Used for new statuses.

**None = 0**

Used when a status class could not be determined. Should not be used in normal operations.

**OnHold = 5**

Used for items that are on hold.

**Requested = 6**

Used for items that have been requested and not yet assigned a status.

# Class Ticket

Namespace: [TeamDynamix.Api.Tickets.Models](#)

Assembly: TeamDynamix.Api.dll

Represents a ticket with all associated details in TeamDynamix.

```
public class Ticket
```

## Inheritance

[object](#) ← Ticket

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AccountId

The ID of the account/department associated with the ticket.

```
[JsonProperty("AccountID")]
public int AccountId { get; set; }
```

### PropertyValue

[int](#)

### AccountName

The name of the account/department associated with the ticket.

```
[JsonProperty("AccountName")]
public string AccountName { get; set; }
```

## Property Value

[string](#)

## ActualMinutes

The time, in minutes, entered against the ticket or associated tasks/activities.

```
[JsonProperty("ActualMinutes")]
public int ActualMinutes { get; set; }
```

## Property Value

[int](#)

## AppId

The ID of the ticketing application associated with the ticket.

```
[JsonProperty("AppID")]
public int AppId { get; set; }
```

## Property Value

[int](#)

## ArticleAppId

The ID of the client portal application associated with the ticket's Knowledge Base article.

```
[JsonProperty("ArticleAppID")]
public int ArticleAppId { get; set; }
```

## Property Value

[int](#)

## ArticleCategoryPathNames

A delimited string of the category hierarchy associated with the ticket's Knowledge Base article.

```
[JsonProperty("ArticleCategoryPathNames")]
public string ArticleCategoryPathNames { get; set; }
```

Property Value

[string](#)

## ArticleId

The ID of the Knowledge Base article associated with the ticket.

```
[JsonProperty("ArticleID")]
public int ArticleId { get; set; }
```

Property Value

[int](#)

## ArticleShortcutId

The ID of the shortcut that is used when viewing the ticket's Knowledge Base article. This is set when the ticket is associated with a cross client portal article shortcut. Nullable.

```
[JsonProperty("ArticleShortcutID")]
public int? ArticleShortcutId { get; set; }
```

Property Value

[int](#)?

## ArticleStatus

The ID of the Knowledge Base article associated with the ticket.

```
[JsonProperty("ArticleStatus")]
public ArticleStatus ArticleStatus { get; set; }
```

Property Value

[ArticleStatus](#)

## ArticleSubject

The subject of the Knowledge Base article associated with the ticket.

```
[JsonProperty("ArticleSubject")]
public string ArticleSubject { get; set; }
```

Property Value

[string](#)

## Attachments

The attachments associated with the ticket. Nullable.

```
[JsonProperty("Attachments")]
public List<Attachment>? Attachments { get; set; }
```

Property Value

[List](#) <[Attachment](#)>

## Attributes

The custom attributes associated with the ticket. Nullable.

```
[JsonProperty("Attributes")]
```

```
public List<CustomAttribute>? Attributes { get; set; }
```

Property Value

[List ↗ <CustomAttribute>](#)

## Classification

The classification associated with the ticket.

```
[JsonProperty("Classification")]
public TicketClass Classification { get; set; }
```

Property Value

[TicketClass](#)

## ClassificationName

The application-defined name of the classification associated with the ticket.

```
[JsonProperty("ClassificationName")]
public string ClassificationName { get; set; }
```

Property Value

[string ↗](#)

## CompletedDate

The completed/closed date of the ticket.

```
[JsonProperty("CompletedDate")]
public DateTime CompletedDate { get; set; }
```

Property Value

[DateTime](#)

## CompletedFullName

The full name of the user who completed/closed the ticket.

```
[JsonProperty("CompletedFullName")]
public string CompletedFullName { get; set; }
```

Property Value

[string](#)

## CompletedUid

The UID of the user who completed/closed the ticket. Nullable because it may not be provided.

```
[JsonProperty("CompletedUid")]
public string? CompletedUid { get; set; }
```

Property Value

[string](#)

## ConvertedToDate

The date the ticket was converted to a project task.

```
[JsonProperty("ConvertedToDate")]
public DateTime ConvertedToDate { get; set; }
```

Property Value

[DateTime](#)

## ConvertedToTaskFullName

The full name of the user who converted the ticket to a project task.

```
[JsonProperty("ConvertedToTaskFullName")]
public string ConvertedToTaskFullName { get; set; }
```

### Property Value

[string](#)

## ConvertedToTaskUid

The UID of the user who converted the ticket to a project task. Nullable because it may not be provided.

```
[JsonProperty("ConvertedToTaskUid")]
public string? ConvertedToTaskUid { get; set; }
```

### Property Value

[string](#)

## CreatedDate

The created date of the ticket.

```
[JsonProperty("CreatedDate")]
public DateTime CreatedDate { get; set; }
```

### Property Value

[DateTime](#)

## CreatedEmail

The email address of the user who created the ticket.

```
[JsonProperty("CreatedEmail")]
public string CreatedEmail { get; set; }
```

Property Value

[string](#)

## CreatedFullName

The full name of the user who created the ticket.

```
[JsonProperty("CreatedFullName")]
public string CreatedFullName { get; set; }
```

Property Value

[string](#)

## CreatedUid

The UID of the user who created the ticket.

```
[JsonProperty("CreatedUid")]
public string? CreatedUid { get; set; }
```

Property Value

[string](#)

## DaysOld

The age of the ticket, in days.

```
[JsonProperty("DaysOld")]
public int DaysOld { get; set; }
```

## Property Value

[int ↗](#)

## Description

The description of the ticket.

```
[JsonProperty("Description")]
public string? Description { get; set; }
```

## Property Value

[string ↗](#)

## EndDate

The end date of the ticket. Nullable because it may not be provided.

```
[JsonProperty("EndDate")]
public DateTime? EndDate { get; set; }
```

## Property Value

[DateTime ↗?](#)

## EstimatedMinutes

The estimated minutes of the ticket.

```
[JsonProperty("EstimatedMinutes")]
public int EstimatedMinutes { get; set; }
```

## Property Value

[int ↗](#)

## ExpensesBudget

The expense budget of the ticket.

```
[JsonProperty("ExpensesBudget")]
public double ExpensesBudget { get; set; }
```

Property Value

[double](#)

## ExpensesBudgetUsed

The used expense budget of the ticket.

```
[JsonProperty("ExpensesBudgetUsed")]
public double ExpensesBudgetUsed { get; set; }
```

Property Value

[double](#)

## FormId

The ID of the form associated with the ticket.

```
[JsonProperty("FormID")]
public int FormId { get; set; }
```

Property Value

[int](#)

## FormName

The name of the form associated with the ticket.

```
[JsonProperty("FormName")]
public string FormName { get; set; }
```

Property Value

[string](#)

## GoesOffHoldDate

The date the ticket goes off hold.

```
[JsonProperty("GoesOffHoldDate")]
public DateTime? GoesOffHoldDate { get; set; }
```

Property Value

[DateTime](#)?

## Id

The ID of the ticket.

```
[JsonProperty("ID")]
public int Id { get; set; }
```

Property Value

[int](#)

## ImpactId

The ID of the impact associated with the ticket.

```
[JsonProperty("ImpactID")]
public int ImpactId { get; set; }
```

Property Value

[int](#)

## ImpactName

The name of the impact associated with the ticket.

```
[JsonProperty("ImpactName")]
public string ImpactName { get; set; }
```

Property Value

[string](#)

## IsConvertedToTask

Whether the ticket has been converted to a project task.

```
[JsonProperty("IsConvertedToTask")]
public bool IsConvertedToTask { get; set; }
```

Property Value

[bool](#)

## IsOnHold

The on hold status of the ticket.

```
[JsonProperty("IsOnHold")]
public bool IsOnHold { get; set; }
```

Property Value

[bool](#)

## IsRichHtml

Indicates if the ticket description is rich-text or plain-text.

```
[JsonProperty("IsRichHtml")]
public bool IsRichHtml { get; set; }
```

Property Value

[bool](#)

## IsSlaResolveByViolated

Whether the Service Level Agreement (SLA) associated with the ticket has its "Resolve By" component violated.

```
[JsonProperty("IsSlaResolveByViolated")]
public bool? IsSlaResolveByViolated { get; set; }
```

Property Value

[bool](#)?

## IsSlaRespondByViolated

Whether the Service Level Agreement (SLA) associated with the ticket has its "Respond By" component violated.

```
[JsonProperty("IsSlaRespondByViolated")]
public bool? IsSlaRespondByViolated { get; set; }
```

Property Value

[bool](#)?

## IsSlaViolated

Whether the Service Level Agreement (SLA) associated with the ticket has been violated.

```
[JsonProperty("IsSlaViolated")]
public bool IsSlaViolated { get; set; }
```

Property Value

[bool](#)

## LocationId

The ID of the location associated with the ticket.

```
[JsonProperty("LocationID")]
public int LocationId { get; set; }
```

Property Value

[int](#)

## LocationName

The name of the location associated with the ticket.

```
[JsonProperty("LocationName")]
public string LocationName { get; set; }
```

Property Value

[string](#)

## LocationRoomId

The ID of the location room associated with the ticket.

```
[JsonProperty("LocationRoomID")]
```

```
public int LocationRoomId { get; set; }
```

Property Value

[int](#)

## LocationRoomName

The name of the location room associated with the ticket.

```
[JsonProperty("LocationRoomName")]
public string LocationRoomName { get; set; }
```

Property Value

[string](#)

## ModifiedDate

The last modified date of the ticket.

```
[JsonProperty("ModifiedDate")]
public DateTime ModifiedDate { get; set; }
```

Property Value

[DateTime](#)

## ModifiedFullName

The full name of the user who last modified the ticket.

```
[JsonProperty("ModifiedFullName")]
public string ModifiedFullName { get; set; }
```

Property Value

[string](#)

## ModifiedUid

The UID of the user who last modified the ticket.

```
[JsonProperty("ModifiedUid")]
public string? ModifiedUid { get; set; }
```

## Property Value

[string](#)

## Notify

The list of people who can be notified for the ticket. Nullable.

```
[JsonProperty("Notify")]
public List<ResourceItem>? Notify { get; set; }
```

## Property Value

[List](#)<[ResourceItem](#)>

## ParentClass

The classification of the parent associated with the ticket.

```
[JsonProperty("ParentClass")]
public TicketClass ParentClass { get; set; }
```

## Property Value

[TicketClass](#)

## ParentId

The ID of the parent associated with the ticket.

```
[JsonProperty("ParentID")]
public int ParentId { get; set; }
```

### Property Value

[int](#)

## ParentTitle

The title of the parent associated with the ticket.

```
[JsonProperty("ParentTitle")]
public string ParentTitle { get; set; }
```

### Property Value

[string](#)

## PlacedOnHoldDate

The date the ticket was placed on hold.

```
[JsonProperty("PlacedOnHoldDate")]
public DateTime? PlacedOnHoldDate { get; set; }
```

### Property Value

[DateTime](#)?

## PriorityId

The ID of the priority associated with the ticket.

```
[JsonProperty("PriorityID")]
public int PriorityId { get; set; }
```

Property Value

[int](#)

## PriorityName

The name of the priority associated with the ticket.

```
[JsonProperty("PriorityName")]
public string PriorityName { get; set; }
```

Property Value

[string](#)

## PriorityOrder

The order of the priority associated with the ticket.

```
[JsonProperty("PriorityOrder")]
public double PriorityOrder { get; set; }
```

Property Value

[double](#)

## RefCode

The reference code of the ticket.

```
[JsonProperty("RefCode")]
public string RefCode { get; set; }
```

Property Value

[string](#)

## RequestorEmail

The email address of the requestor associated with the ticket.

```
[JsonProperty("RequestorEmail")]
public string RequestorEmail { get; set; }
```

Property Value

[string](#)

## RequestorFirstName

The first name of the requestor associated with the ticket.

```
[JsonProperty("RequestorFirstName")]
public string RequestorFirstName { get; set; }
```

Property Value

[string](#)

## RequestorLastName

The last name of the requestor associated with the ticket.

```
[JsonProperty("RequestorLastName")]
public string RequestorLastName { get; set; }
```

Property Value

[string](#)

## RequestorName

The full name of the requestor associated with the ticket.

```
[JsonProperty("RequestorName")]
public string RequestorName { get; set; }
```

### Property Value

[string](#)

## RequestorPhone

The phone number of the requestor associated with the ticket.

```
[JsonProperty("RequestorPhone")]
public string RequestorPhone { get; set; }
```

### Property Value

[string](#)

## RequestorUid

The UID of the requestor associated with the ticket. Nullable because it may not be provided in some cases.

```
[JsonProperty("RequestorUID")]
public string? RequestorUid { get; set; }
```

### Property Value

[string](#)

## ResolveByDate

The "Resolve By" deadline of the Service Level Agreement (SLA) associated with the ticket.

```
[JsonProperty("ResolveByDate")]
public DateTime? ResolveByDate { get; set; }
```

Property Value

[DateTime](#)?

## RespondByDate

The "Respond By" deadline of the Service Level Agreement (SLA) associated with the ticket.

```
[JsonProperty("RespondByDate")]
public DateTime? RespondByDate { get; set; }
```

Property Value

[DateTime](#)?

## RespondedDate

The date the ticket was responded to.

```
[JsonProperty("RespondedDate")]
public DateTime RespondedDate { get; set; }
```

Property Value

[DateTime](#)

## RespondedFullName

The full name of the user who responded to the ticket.

```
[JsonProperty("RespondedFullName")]
public string RespondedFullName { get; set; }
```

Property Value

[string](#)

## RespondedUid

The UID of the user who responded to the ticket. Nullable because it may not be provided.

```
[JsonProperty("RespondedUid")]
public string? RespondedUid { get; set; }
```

Property Value

[string](#)

## ResponsibleEmail

The email address of the responsible user associated with the ticket.

```
[JsonProperty("ResponsibleEmail")]
public string ResponsibleEmail { get; set; }
```

Property Value

[string](#)

## ResponsibleFullName

The full name of the responsible user associated with the ticket.

```
[JsonProperty("ResponsibleFullName")]
public string ResponsibleFullName { get; set; }
```

Property Value

[string](#)

## ResponsibleGroupId

The ID of the responsible group associated with the ticket.

```
[JsonProperty("ResponsibleGroupID")]
public int ResponsibleGroupId { get; set; }
```

### Property Value

[int](#)

## ResponsibleGroupName

The name of the responsible group associated with the ticket.

```
[JsonProperty("ResponsibleGroupName")]
public string ResponsibleGroupName { get; set; }
```

### Property Value

[string](#)

## ResponsibleUid

The UID of the responsible user associated with the ticket. Nullable because it may not be provided.

```
[JsonProperty("ResponsibleUid")]
public string? ResponsibleUid { get; set; }
```

### Property Value

[string](#)

## ReviewerEmail

The email address of the reviewing user associated with the ticket.

```
[JsonProperty("ReviewerEmail")]
public string ReviewerEmail { get; set; }
```

Property Value

[string](#)

## ReviewerFullName

The full name of the reviewing user associated with the ticket.

```
[JsonProperty("ReviewerFullName")]
public string ReviewerFullName { get; set; }
```

Property Value

[string](#)

## ReviewerUid

The UID of the reviewing user associated with the ticket. Nullable because it may not be provided.

```
[JsonProperty("ReviewerUid")]
public string? ReviewerUid { get; set; }
```

Property Value

[string](#)

## ReviewingGroupId

The ID of the reviewing group associated with the ticket.

```
[JsonProperty("ReviewingGroupID")]
public int ReviewingGroupId { get; set; }
```

Property Value

[int](#)

## ReviewingGroupName

The name of the reviewing group associated with the ticket.

```
[JsonProperty("ReviewingGroupName")]
public string ReviewingGroupName { get; set; }
```

Property Value

[string](#)

## ServiceCategoryId

The ID of the category associated with the ticket's service.

```
[JsonProperty("ServiceCategoryID")]
public int ServiceCategoryId { get; set; }
```

Property Value

[int](#)

## ServiceCategoryName

The name of the category associated with the ticket's service.

```
[JsonProperty("ServiceCategoryName")]
public string ServiceCategoryName { get; set; }
```

Property Value

[string](#)

## ServiceId

The ID of the service associated with the ticket.

```
[JsonProperty("ServiceID")]
public int ServiceId { get; set; }
```

### Property Value

[int](#)

## ServiceName

The name of the service associated with the ticket.

```
[JsonProperty("ServiceName")]
public string ServiceName { get; set; }
```

### Property Value

[string](#)

## ServiceOfferingId

The ID of the service offering associated with the ticket.

```
[JsonProperty("ServiceOfferingID")]
public int ServiceOfferingId { get; set; }
```

### Property Value

[int](#)

## ServiceOfferingName

The name of the service offering associated with the ticket.

```
[JsonProperty("ServiceOfferingName")]
public string ServiceOfferingName { get; set; }
```

Property Value

[string](#)

## SlaBeginDate

The date the Service Level Agreement (SLA) associated with the ticket was started.

```
[JsonProperty("SlaBeginDate")]
public DateTime? SlaBeginDate { get; set; }
```

Property Value

[DateTime](#)?

## SlaId

The ID of the Service Level Agreement (SLA) associated with the ticket.

```
[JsonProperty("SlaID")]
public int SlaId { get; set; }
```

Property Value

[int](#)

## SlaName

The name of the Service Level Agreement (SLA) associated with the ticket.

```
[JsonProperty("SlaName")]
public string SlaName { get; set; }
```

## PropertyValue

[string](#) ↗

## SourceId

The ID of the ticket source associated with the ticket.

```
[JsonProperty("SourceID")]
public int SourceId { get; set; }
```

## PropertyValue

[int](#) ↗

## SourceName

The name of the ticket source associated with the ticket.

```
[JsonProperty("SourceName")]
public string SourceName { get; set; }
```

## PropertyValue

[string](#) ↗

## StartDate

The start date of the ticket. Nullable because it may not be provided.

```
[JsonProperty("StartDate")]
public DateTime? StartDate { get; set; }
```

## PropertyValue

[DateTime](#) ↗?

## StatusClass

The class of the ticket status associated with the ticket.

```
[JsonProperty("StatusClass")]
public StatusClass StatusClass { get; set; }
```

### Property Value

[StatusClass](#)

## StatusId

The ID of the ticket status associated with the ticket.

```
[JsonProperty("StatusID")]
public int StatusId { get; set; }
```

### Property Value

[int](#)

## StatusName

The name of the ticket status associated with the ticket.

```
[JsonProperty("StatusName")]
public string StatusName { get; set; }
```

### Property Value

[string](#)

## TaskEndDate

The end date of the associated project task when the ticket has been converted to a project task.

```
[JsonProperty("TaskEndDate")]
public DateTime TaskEndDate { get; set; }
```

Property Value

[DateTime](#)

## TaskId

The ID of the associated project task when the ticket has been converted to a project task.

```
[JsonProperty("TaskID")]
public int TaskId { get; set; }
```

Property Value

[int](#)

## TaskPercentComplete

The percent complete of the associated project task when the ticket has been converted to a project task.

```
[JsonProperty("TaskPercentComplete")]
public int TaskPercentComplete { get; set; }
```

Property Value

[int](#)

## TaskPlanId

The ID of the associated project plan when the ticket has been converted to a project task.

```
[JsonProperty("TaskPlanID")]
public int TaskPlanId { get; set; }
```

Property Value

[int](#)

## TaskPlanName

The name of the associated project plan when the ticket has been converted to a project task.

```
[JsonProperty("TaskPlanName")]
public string TaskPlanName { get; set; }
```

Property Value

[string](#)

## TaskProjectId

The ID of the associated project when the ticket has been converted to a project task.

```
[JsonProperty("TaskProjectID")]
public int TaskProjectId { get; set; }
```

Property Value

[int](#)

## TaskProjectName

The name of the associated project when the ticket has been converted to a project task.

```
[JsonProperty("TaskProjectName")]
public string TaskProjectName { get; set; }
```

Property Value

[string](#)

## TaskStartDate

The start date of the associated project task when the ticket has been converted to a project task.

```
[JsonProperty("TaskStartDate")]
public DateTime TaskStartDate { get; set; }
```

Property Value

[DateTime](#)

## TaskTitle

The title of the associated project task when the ticket has been converted to a project task.

```
[JsonProperty("TaskTitle")]
public string TaskTitle { get; set; }
```

Property Value

[string](#)

## Tasks

The ticket tasks associated with the ticket. Nullable.

```
[JsonProperty("Tasks")]
public List<TicketTask>? Tasks { get; set; }
```

Property Value

[List](#)<[TicketTask](#)>

## TimeBudget

The time budget of the ticket.

```
[JsonProperty("TimeBudget")]
public double TimeBudget { get; set; }
```

Property Value

[double](#)

## TimeBudgetUsed

The used time budget of the ticket.

```
[JsonProperty("TimeBudgetUsed")]
public double TimeBudgetUsed { get; set; }
```

Property Value

[double](#)

## Title

The title of the ticket.

```
[JsonProperty("Title")]
public string Title { get; set; }
```

Property Value

[string](#)

## TypeCategoryId

The ID of the category associated with the ticket's type.

```
[JsonProperty("TypeCategoryId")]
public int TypeCategoryId { get; set; }
```

Property Value

[int](#)

## TypeCategoryName

The name of the category associated with the ticket's type.

```
[JsonProperty("TypeCategoryName")]
public string TypeCategoryName { get; set; }
```

Property Value

[string](#)

## TypeId

The ID of the ticket type associated with the ticket.

```
[JsonProperty("TypeID")]
public int TypeId { get; set; }
```

Property Value

[int](#)

## TypeName

The name of the ticket type associated with the ticket.

```
[JsonProperty("TypeName")]
public string TypeName { get; set; }
```

Property Value

[string](#)

## UrgencyId

The ID of the urgency associated with the ticket.

```
[JsonProperty("UrgencyID")]
public int UrgencyId { get; set; }
```

Property Value

[int](#)

## UrgencyName

The name of the urgency associated with the ticket.

```
[JsonProperty("UrgencyName")]
public string UrgencyName { get; set; }
```

Property Value

[string](#)

## Uri

The URI to retrieve the full details of the ticket via the web API.

```
[JsonProperty("Uri")]
public string Uri { get; set; }
```

Property Value

[string](#)

## WorkflowConfigurationId

The ID of the configuration workflow associated with the ticket.

```
[JsonProperty("WorkflowConfigurationID")]
public int WorkflowConfigurationId { get; set; }
```

Property Value

[int](#)

## WorkflowId

The ID of the workflow associated with the ticket.

```
[JsonProperty("WorkflowID")]
public int WorkflowId { get; set; }
```

Property Value

[int](#)

## WorkflowName

The name of the workflow associated with the ticket. Nullable.

```
[JsonProperty("WorkflowName")]
public string? WorkflowName { get; set; }
```

Property Value

[string](#)

# Enum TicketClass

Namespace: [TeamDynamix.Api.Tickets.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different classifications of tickets.

```
public enum TicketClass
```

## Fields

**Change** = 34

A change.

**Incident** = 32

An incident.

**MajorIncident** = 77

A major incident.

**None** = 0

An unknown or indeterminate type of ticket.

**Problem** = 33

A problem.

**Release** = 35

A release.

**ServiceRequest** = 46

A service request.

**Ticket** = 9

An "all tickets" classification for filtering purposes. Tickets should not be created or edited with this class.

```
TicketTemplate = 36
```

A ticket template. Tickets should not be created or edited with this class.

# Class TicketTask

Namespace: [TeamDynamix.Api.Tickets.Models](#)

Assembly: TeamDynamix.Api.dll

A ticket task or activity.

```
public class TicketTask
```

## Inheritance

[object](#) ← TicketTask

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### ActualMinutes

The time, in minutes, entered against the task.

```
[JsonProperty("ActualMinutes")]
public int ActualMinutes { get; set; }
```

### Property Value

[int](#)

### CompleteWithinMinutes

The expected duration, in operational minutes, of the task.

```
[JsonProperty("CompleteWithinMinutes")]
public int? CompleteWithinMinutes { get; set; }
```

Property Value

[int](#)?

## CompletedDate

The completed date of the task.

```
[JsonProperty("CompletedDate")]
public DateTime CompletedDate { get; set; }
```

Property Value

[DateTime](#)

## CompletedFullName

The full name of the user who completed the task.

```
[JsonProperty("CompletedFullName")]
public string CompletedFullName { get; set; }
```

Property Value

[string](#)

## CompletedUid

The UID of the user who completed the task.

```
[JsonProperty("CompletedUID")]
public Guid? CompletedUid { get; set; }
```

Property Value

[Guid](#)?

## CreatedDate

The created date of the task.

```
[JsonProperty("CreatedDate")]
public DateTime CreatedDate { get; set; }
```

### Property Value

[DateTime](#)

## CreatedEmail

The email address of the user who created the task.

```
[JsonProperty("CreatedEmail")]
public string CreatedEmail { get; set; }
```

### Property Value

[string](#)

## CreatedFullName

The full name of the user who created the task.

```
[JsonProperty("CreatedFullName")]
public string CreatedFullName { get; set; }
```

### Property Value

[string](#)

## CreatedUid

The UID of the user who created the task.

```
[JsonProperty("CreatedUID")]
public Guid CreatedUid { get; set; }
```

Property Value

[Guid](#)

## Description

The description of the task.

```
[JsonProperty("Description")]
public string? Description { get; set; }
```

Property Value

[string](#)

## DetectedConflictCount

The number of detected conflicts for this task.

```
[JsonProperty("DetectedConflictCount")]
public int DetectedConflictCount { get; set; }
```

Property Value

[int](#)

## DetectedConflictTypes

The type of detected conflicts for this task.

```
[JsonProperty("DetectedConflictTypes")]
public ConflictType DetectedConflictTypes { get; set; }
```

Property Value

[ConflictType](#)

## EndDate

The end date of the task.

```
[JsonProperty("EndDate")]
public DateTime? EndDate { get; set; }
```

Property Value

[DateTime](#)?

## EstimatedMinutes

The estimated minutes of the task.

```
[JsonProperty("EstimatedMinutes")]
public int EstimatedMinutes { get; set; }
```

Property Value

[int](#)

## Id

The ID of the task.

```
[JsonProperty("ID")]
public int Id { get; set; }
```

Property Value

[int](#)

## IsActive

The active status of the ticket task.

```
[JsonProperty("IsActive")]
public bool IsActive { get; set; }
```

Property Value

[bool](#)

## LastConflictScanDateUtc

The date the task was last scanned for conflicts.

```
[JsonProperty("LastConflictScanDateUtc")]
public DateTime LastConflictScanDateUtc { get; set; }
```

Property Value

[DateTime](#)

## ModifiedDate

The last modified date of the task.

```
[JsonProperty("ModifiedDate")]
public DateTime ModifiedDate { get; set; }
```

Property Value

[DateTime](#)

## ModifiedFullName

The full name of the user who last modified the task.

```
[JsonProperty("ModifiedFullName")]
public string ModifiedFullName { get; set; }
```

Property Value

[string](#)

## ModifiedUid

The UID of the user who last modified the task.

```
[JsonProperty("ModifiedUID")]
public Guid ModifiedUid { get; set; }
```

Property Value

[Guid](#)

## NotifyResponsible

Whether to notify responsible party. Defaults to true.

```
[JsonProperty("NotifyResponsible")]
public bool NotifyResponsible { get; set; }
```

Property Value

[bool](#)

## Order

The order in which the task should be displayed in the list of the ticket's tasks/activities.

```
[JsonProperty("Order")]
public int Order { get; set; }
```

Property Value

[int](#)

## PercentComplete

The percent complete of the task.

```
[JsonProperty("PercentComplete")]
public int PercentComplete { get; set; }
```

Property Value

[int](#)

## PredecessorId

The ID of the predecessor associated with the task.

```
[JsonProperty("PredecessorID")]
public int PredecessorId { get; set; }
```

Property Value

[int](#)

## PredecessorTitle

The title of the predecessor associated with the task.

```
[JsonProperty("PredecessorTitle")]
public string PredecessorTitle { get; set; }
```

Property Value

[string](#)

## ResponsibleEmail

The email address of the user responsible for the task.

```
[JsonProperty("ResponsibleEmail")]
public string ResponsibleEmail { get; set; }
```

### Property Value

[string](#)

## ResponsibleFullName

The full name of the user responsible for the task.

```
[JsonProperty("ResponsibleFullName")]
public string ResponsibleFullName { get; set; }
```

### Property Value

[string](#)

## ResponsibleGroupId

The ID of the group responsible for the task.

```
[JsonProperty("ResponsibleGroupID")]
public int ResponsibleGroupId { get; set; }
```

### Property Value

[int](#)

## ResponsibleGroupName

The name of the group responsible for the task.

```
[JsonProperty("ResponsibleGroupName")]
public string ResponsibleGroupName { get; set; }
```

Property Value

[string](#)?

## ResponsibleUid

The UID of the user responsible for the task.

```
[JsonProperty("ResponsibleUID")]
public Guid? ResponsibleUid { get; set; }
```

Property Value

[Guid](#)?

## StartDate

The start date of the task.

```
[JsonProperty("StartDate")]
public DateTime? StartDate { get; set; }
```

Property Value

[DateTime](#)?

## TicketId

The ID of the ticket associated with the task.

```
[JsonProperty("TicketID")]
public int TicketId { get; set; }
```

## Property Value

[int](#)

## Title

The title of the task.

```
[JsonProperty("Title")]
public string Title { get; set; }
```

## Property Value

[string](#)

## TypeId

The type ID of the task.

```
[JsonProperty("TypeID")]
public TicketTaskType TypeId { get; set; }
```

## Property Value

[TicketTaskType](#)

## Uri

The URI to retrieve the full details of the ticket task via the web API.

```
[JsonProperty("Uri")]
public string Uri { get; set; }
```

## Property Value

[string](#)

# Enum TicketTaskType

Namespace: [TeamDynamix.Api.Tickets.Models](#)

Assembly: TeamDynamix.Api.dll

Describes the different types of items that can be backended by ticket tasks.

```
public enum TicketTaskType
```

## Fields

**MaintenanceActivity** = 2

A scheduled maintenance activity.

**None** = 0

An unknown or indeterminate type of task.

**TicketTask** = 1

A standard, workable ticket task.

**WorkflowTask** = 3

A task that is used as a work step in a workflow.

# Namespace TeamDynamix.Api.Tickets.Options

## Classes

### [TicketCreateOptions](#)

Represents optional parameters that can be applied when creating a ticket via the TeamDynamix API. These options are serialized into query string parameters.

# Class TicketCreateOptions

Namespace: [TeamDynamix.Api.Tickets.Options](#)

Assembly: TeamDynamix.Api.dll

Represents optional parameters that can be applied when creating a ticket via the TeamDynamix API. These options are serialized into query string parameters.

```
public class TicketCreateOptions
```

## Inheritance

[object](#) ← TicketCreateOptions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) ,  
[object.ToString\(\)](#)

## Properties

### AllowRequestorCreation

Gets or sets a value indicating whether to allow creation of the requestor if they do not already exist.

```
public bool AllowRequestorCreation { get; set; }
```

#### Property Value

[bool](#)

### ApplyDefaults

Gets or sets a value indicating whether to apply default values to the ticket.

```
public bool ApplyDefaults { get; set; }
```

Property Value

[bool](#)

## EnableNotifyReviewer

Gets or sets a value indicating whether to enable notifications to the reviewer.

```
public bool EnableNotifyReviewer { get; set; }
```

Property Value

[bool](#)

## NotifyRequestor

Gets or sets a value indicating whether to notify the requestor.

```
public bool NotifyRequestor { get; set; }
```

Property Value

[bool](#)

## NotifyResponsible

Gets or sets a value indicating whether to notify the person responsible for the ticket.

```
public bool NotifyResponsible { get; set; }
```

Property Value

[bool](#)

## PreferRequestorAccountAndPriority

Gets or sets a value indicating whether to prefer the requestor's account and priority over defaults.

```
public bool PreferRequestorAccountAndPriority { get; set; }
```

Property Value

[bool](#)

## Methods

### ToQueryString()

Converts the current options to a URL query string suitable for appending to an HTTP request.

```
public string ToQueryString()
```

Returns

[string](#)

A query string representation of the ticket creation options.