

LinkedIn Learning Search API

The LinkedIn Learning Search API is a set of endpoints you can call to search the catalog of LinkedIn Learning content. The main endpoint in the LinkedIn Learning Search API takes a request with search and relevance criteria for LinkedIn Learning content and returns the content metadata in JSON format. Since the LinkedIn Learning Search API uses the same search functionality as the LinkedIn Learning application, you can call the API to find the same search results as you would find in LinkedIn Learning. This document explains how you can access and use the LinkedIn Learning Search API.

Content

Overview

Getting access

- Obtaining a client id and client secret
- Generating an access token

API terminology

- Asset
- Classification
- URN

Using the API: basics

- Including the access token
- Understanding the endpoints

API endpoints

- GET /v2/learningAssets
- GET /v2/learningClassifications

Using the API: advanced

- Specifying the level of asset details
- Rolling up sub-asset classifications
- Specifying the response fields

Schema reference

- Shared schemas
- Asset schema
- Classification schema

Overview

The LinkedIn Learning Search API complements the LinkedIn Learning Content API. The Content API allows you to bulk import the catalog of LinkedIn Learning content, the Search API allows you to discover a subset of LinkedIn Learning content based on search parameters, sorted by relevance, popularity, or recency. As part of this use case, the Search API also supports searching category and skill metadata.

The LinkedIn Learning Search API and Content API share the same key concepts, terminology, and data models. The Search API builds on the Content API to allow you to query the LinkedIn Learning content library and control how the results appear.

One important difference between the LinkedIn Learning Search API and Content API is how the APIs treat retired content. In the Content API, you can specify a parameter to retrieve retired as well as active content. In the Search API, you can retrieve only active content. When bulk importing the catalog of LinkedIn Learning content using the Content API, you may wish to include retired content for completeness of records - for example, to be able to audit all content that learners have viewed. When searching content using the Search API, however, it makes sense to be able to discover only content that is active and can be viewed.

Getting Access

To access the LinkedIn Learning Search API, you will first need a partnership and signed agreement with LinkedIn Learning. Please work with our business development and technical consulting teams to get started: LLSbd@linkedin.com.

The LinkedIn Learning Search API uses a two-legged OAuth 2.0 flow for access. With a partnership and signed agreement in place, you will need to generate an *access token* to call the API. The access token is a value that must be included in a request header in each call to the API.

Obtaining a client id and client secret

To generate an access token, you will need a *client id* and *client secret*. Depending on your requirements, you can get the client id and client secret in one of two ways:

1. If your organization is a part of the LinkedIn Learning Partner Program, you can reach out to your dedicated Business Development contact to request access to a LinkedIn Learning test instance. Within the LinkedIn Learning admin interface, you can provision API keys for testing and development.
2. Alternatively, work with a specific organization to get a client id and client secret directly from that organization (the organization must first have an account with LinkedIn Learning to log in and generate the client id and client secret). With this client id and client secret, the access token you generate will permit the LinkedIn Learning Search API to return organization-specific metadata in addition to publicly-available metadata. In particular, the API will return the AICC launch URL of LinkedIn Learning content, a URL that includes an identifier for a specific organization.

Generating an access token

Once you have a client id and client secret, you can generate an access token by issuing a GET request to the following endpoint:

GET https://www.linkedin.com/oauth/v2/accessToken

Parameter	Description	Required
grant_type	The value of this parameter should always be: client_credentials .	Yes
client_id	The client id obtained by following the steps above.	Yes
client_secret	The client secret obtained by following the steps above.	Yes

Sample request (using cURL):

curl

'https://www.linkedin.com/oauth/v2/accessToken?grant_type=client_credentials&client_id=sampleClientIdValue&client_secret=sampleClientSecretValue'

Sample response:

```
{  
  "access_token": "AQXt...",  
  "expires_in": 7775999  
}
```

Field	Description
access_token	The access token that must be included in a request header in each call to the LinkedIn Learning Search API. This value must be kept secure.
expires_in	The number of seconds remaining, from the time it was requested, before the token will expire. You can request a new token once your previous token expires.

Once you have generated an access token, you can start using the LinkedIn Learning Search API. See the "Including the access token" section for how to call the API with the access token.

If you call the LinkedIn Learning Search API with an invalid access token, you will receive a "401 Unauthorized" error. An access token could be invalid because it has expired or was revoked. When you integrate with the API, it is important to code your application to properly handle "401 Unauthorized" errors. For example, if you receive the error, you can use the client id and client secret to generate a new access token and retry your request.

API Terminology

Key terms used by the LinkedIn Learning Search API:

"Asset"

A learning *asset* is the unified and extensible representation of LinkedIn Learning content in the API. A learning asset may contain other learning assets: for example, a course contains chapters. In the LinkedIn Learning Search API, all learning content is represented as an asset, which may contain nested sub-assets.

"Classification"

A learning *classification* is category or skill metadata that a learning asset may be tagged with. For example, a course called "Learning Java" may be classified under "Technology" - a category - and teach "Object-Oriented Programming (OOP)" - a skill. In the LinkedIn Learning Search API, learning classifications provide context about a learning asset.

"URN"

An *URN* (Uniform Resource Name) is a unique identifier for an entity in the LinkedIn Learning Search API. All entities in the API - learning assets, learning classifications, as well as others (for example, authors) - are identified by URNs. Some example URNs are "urn:li:lyndaCourse:184457", "urn:li:lyndaCategory:7164", and "urn:li:lyndaAuthor:2975371".

Using the API: Basics

This section explains the basics of using the LinkedIn Learning Search API. For details and the full schema reference, see the next sections.

Including the access token

Each request to the LinkedIn Learning Search API must include a header that contains an access token, generated by following the steps in the "Getting access" section. To include the access token in your request to the API, set an "Authorization" header in your request, with the access token in the header value (note the "Bearer" authorization type preceding the access token; this value is required by the authorization protocol):

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java'
```

Sample response:

```
{
  "elements": [
    {
      "urn": "urn:li:lyndaCourse:375490",
      "contents": [
        {
          "asset": {
            "urn": "urn:li:lyndaChapter:(urn:li:lyndaCourse:375490,415232)",
            "contents": [
              {
                "asset": {
                  "urn": "urn:li:lyndaVideo:(urn:li:lyndaCourse:375490,415233)",
                  "contents": [],
                  "type": "VIDEO",
                  "title": {
                    "locale": {
                      "country": "US",
                      "language": "en"
                    },
                    "value": "Welcome"
                  }
                }
              }
            ]
          },
          ...
        ],
        "metadata": {
          "assetTypeFacetMetadata": [
            {
              "count": 2094,
              "assetType": "VIDEO"
            },
            ...
          ],
          "paging": {
            "total": 2179,
            "count": 20,
            "start": 0,
            "links": [
              {
                "rel": "next",
                "href":
"/v2/learningAssets?assetFilteringCriteria.keyword=java&count=20&q=criteria&start=20",
                "type": "application/json"
              }
            ]
          }
        }
      ]
    }
  ]
}
```

Understanding the endpoints

The LinkedIn Learning Search API provides two endpoints - one for learning assets, and one for learning classifications. You can call these endpoints to:

- Retrieve a page of learning assets, given some search and relevance criteria.
- Retrieve a page of learning classifications, given a keyword.

Each API endpoint and its parameters are documented in the "API endpoints" section. First, a note about pagination, using learning assets as an example (the same applies for learning classifications).

When you call the endpoint that retrieves a page of learning assets, given some search and relevance criteria, you will get a response like the following:

```
{
  "elements": [
    ...
  ],
  "metadata": {
    ...
  },
  "paging": {
    "total": 2179,
    "count": 20,
    "start": 100,
    "links": [
      {
        "rel": "prev",
        "href":
"/v2/learningAssets?assetFilteringCriteria.keyword=java&count=20&q=criteria&start=80",
        "type": "application/json"
      },
      {
        "rel": "next",
        "href":
"/v2/learningAssets?assetFilteringCriteria.keyword=java&count=20&q=criteria&start=220",
        "type": "application/json"
      }
    ]
  }
}
```

The value of the "elements" field is an array of learning assets for the requested page. The value of the "metadata" field, if present, is an object with metadata about the search results.

The value of the "paging" field is an object with paging metadata. In particular, if you wish to retrieve the next page of search results, you can use the "links" field in the paging metadata to help construct a request for the next page. The "links" field simplifies requests for additional pages because you will not need to track the page start and count yourself.

Field	Description
total	The total number (i.e. across all pages) of learning entities matching the requested criteria.
count	The requested number of learning entities for the page.
start	The requested start index of learning entities for the page.
links	An array of link objects. Each link object includes: <ul style="list-style-type: none">• A "rel" field whose value can be prev or next.• An "href" field whose value is the link (relative to the base API URL) for the previous or next page of learning entities matching the requested criteria. If the previous or next page does not exist, the corresponding link object will be omitted.

API Endpoints

Each LinkedIn Learning Search API endpoint and its parameters are documented below. Note that all parameter names and values (unless noted) are case-sensitive.

GET /v2/learningAssets

To retrieve a page of learning assets, given some search and relevance criteria, issue a GET call to the following endpoint:

GET <https://api.linkedin.com/v2/learningAssets>

The parameters of this endpoint are grouped into three logical sets of criteria:

- *assetFilteringCriteria*: criteria that narrows the content you wish to retrieve from the LinkedIn Learning catalog.
- *assetPresentationCriteria*: criteria that controls how the search results appear.
- *assetRetrievalCriteria*: criteria that controls how much information is retrieved for the search results.

Parameter	Description	Required
q	The value of this parameter should always be: criteria .	Yes
assetFilteringCriteria.assetTypes	<p>An <i>array</i> of types of learning assets to search. The search results will include only learning assets of these types. The values of this parameter should be COURSE, LEARNING_PATH, or VIDEO.</p> <p>If omitted, the search results will include learning assets of any type.</p> <p>Since this parameter is an array, you will need to specify a zero-based index per value. For example:</p> <pre>assetFilteringCriteria.assetTypes[0]=COURSE assetFilteringCriteria.assetTypes[0]=LEARNING_PATH& assetFilteringCriteria.assetTypes[1]=VIDEO</pre>	No
assetFilteringCriteria.classifications	<p>An <i>array</i> of learning classification URNs to search learning assets. The search results will include only learning assets tagged with these classifications.</p> <p>The supported URN types are "urn:li:lyndaCategory" and "urn:li:skill". The URNs can be discovered using the /v2/learningClassifications endpoint.</p> <p>If omitted, the search results will include learning assets tagged with any classification.</p> <p>Since this parameter is an array, you will need to specify a zero-based index per value. For example:</p> <pre>assetFilteringCriteria.classifications[0]=urn:li:lynda Category:7164 assetFilteringCriteria.classifications[0]=urn:li:lynda Category:7164&assetFilteringCriteria.classifications[1]= urn:li:skill:1328</pre>	No
assetFilteringCriteria.difficultyLevels	<p>An <i>array</i> of difficulty levels of learning assets to search. The search results will include only learning assets of these difficulty levels. The values of this parameter should be BEGINNER, INTERMEDIATE, or ADVANCED.</p> <p>If omitted, the search results will include learning assets of any difficulty level.</p> <p>Since this parameter is an array, you will need to specify a zero-based index per value. For example:</p> <pre>assetFilteringCriteria.difficultyLevels[0]=BEGINNER assetFilteringCriteria.difficultyLevels[0]=INTERMEDIATE& assetFilteringCriteria.difficultyLevels[1]=ADVANCED</pre>	No

assetFilteringCriteria.keyword	<p>The keyword string to search learning assets. The search results will include only learning assets matching this keyword string, as determined by LinkedIn Learning's relevance algorithm. The value of this parameter is case-insensitive.</p> <p>If omitted, the search results will include learning assets matching any keyword string.</p>	No
assetPresentationCriteria.sortBy	<p>How to sort the learning assets in the search results. The value of this parameter should be RELEVANCE, POPULARITY, or RECENCY.</p> <p><i>Relevance</i> sorts the learning assets by LinkedIn Learning's relevance algorithm. <i>Popularity</i> sorts the learning assets by view count. <i>Recency</i> sorts the learning assets by publish date.</p>	No (default = RELEVANCE)
assetPresentationCriteria.targetLocale.language	<p>The locale language the API will use to try to localize the learning asset. The value of this parameter should be de, en, es, fr, or ja. These values correspond to the locales "de_DE", "en_US", "es_ES", "fr_FR", and "ja_JP".</p> <p>If the learning asset cannot be localized or if the target locale is not set, the API will use the source locale of the learning asset.</p>	No
assetPresentationCriteria.targetLocale.country	<p>The locale country the API will use to try to localize the learning asset. The value of this parameter should be DE, US, ES, FR, or JP. These values correspond to the locales "de_DE", "en_US", "es_ES", "fr_FR", and "ja_JP".</p> <p>If the learning asset cannot be localized or if the target locale is not set, the API will use the source locale of the learning asset.</p>	No
assetRetrievalCriteria.expandDepth	<p>The number of levels in the learning asset hierarchy to include asset details. This parameter is optional; please see the "Specifying the level of asset details" section for an explanation with examples.</p>	No (default = 1)
start	<p>The start index of learning assets for the page.</p>	No (default = 0)
count	<p>The number of learning assets to include in the page. Please choose a reasonable value if overriding the default; the API will return an error if the requested page exceeds the 2 MB response size limit of the framework.</p>	No (default = 20)

Sample request (using cURL; note the "-g" flag since the URL includes brackets):

```
curl -H 'Authorization: Bearer AQXt...' -g  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.assetTypes[0]=COURSE&asset  
FilteringCriteria.assetTypes[1]=VIDEO&assetFilteringCriteria.classifications[0]=urn:li:lyndaCategory:7164&  
assetFilteringCriteria.difficultyLevels[0]=BEGINNER&assetFilteringCriteria.difficultyLevels[1]=INTERMEDIATE  
&assetFilteringCriteria.keyword=java&assetPresentationCriteria.sortBy=RELEVANCE&assetPresentation  
Criteria.targetLocale.language=en&assetPresentationCriteria.targetLocale.country=US&assetRetrieval  
Criteria.expandDepth=1&start=100&count=20'
```

Sample response:

```
{  
  "elements": [  
    {  
      "urn": "urn:li:lyndaCourse:375490",  
      "contents": [  
        {  
          "asset": {  
            "urn": "urn:li:lyndaChapter:(urn:li:lyndaCourse:375490,415232)",  
            "contents": [  
              {  
                "asset": {  
                  "urn": "urn:li:lyndaVideo:(urn:li:lyndaCourse:375490,415233)",  
                  "contents": [],  
                  "type": "VIDEO",  
                  "title": {  
                    "locale": {  
                      "country": "US",  
                      "language": "en"  
                    },  
                    "value": "Welcome"  
                  }  
                }  
              }  
            ],  
            "type": "VIDEO"  
          }  
        },  
        ...  
      ],  
      "metadata": {  
        "assetTypeFacetMetadata": [  
          {  
            "count": 1626,  
            "assetType": "VIDEO"  
          },  
          {  
            "count": 67,  
            "assetType": "COURSE"  
          }  
        ],  
        {  
          {
```

```

    "count": 0,
    "assetType": "LEARNING_PATH"
  }
  ...
},
"paging": {
  "total": 1693,
  "count": 20,
  "start": 100,
  "links": [
    {
      "rel": "prev",
      "href":
"/v2/learningAssets?assetFilteringCriteria.assetTypes[0]=COURSE&assetFilteringCriteria.assetTypes[1]=
VIDEO&assetFilteringCriteria.classifications[0]=urn%3Ali%3AlyndaCategory%3A7164&assetFiltering
Criteria.difficultyLevels[0]=BEGINNER&assetFilteringCriteria.difficultyLevels[1]=INTERMEDIATE&asset
FilteringCriteria.keyword=java&assetPresentationCriteria.sortBy=RELEVANCE&assetPresentationCriteria.
targetLocale.country=US&assetPresentationCriteria.targetLocale.language=en&assetRetrievalCriteria.
expandDepth=1&count=20&q=criteria&start=80",
      "type": "application/json"
    },
    {
      "rel": "next",
      "href":
"/v2/learningAssets?assetFilteringCriteria.assetTypes[0]=COURSE&assetFilteringCriteria.assetTypes[1]=
VIDEO&assetFilteringCriteria.classifications[0]=urn%3Ali%3AlyndaCategory%3A7164&assetFiltering
Criteria.difficultyLevels[0]=BEGINNER&assetFilteringCriteria.difficultyLevels[1]=INTERMEDIATE&asset
FilteringCriteria.keyword=java&assetPresentationCriteria.sortBy=RELEVANCE&assetPresentationCriteria.
targetLocale.country=US&assetPresentationCriteria.targetLocale.language=en&assetRetrievalCriteria.
expandDepth=1&count=20&q=criteria&start=120",
      "type": "application/json"
    }
  ]
}
}

```

GET /v2/learningClassifications

To retrieve a page of learning classifications, given a keyword, issue a GET call to the following endpoint:

GET <https://api.linkedin.com/v2/learningClassifications>

Parameter	Description	Required
q	The value of this parameter should always be: keyword .	Yes
keyword	The keyword string to search learning classifications. The search results will include only learning classifications matching this keyword string, as determined by LinkedIn Learning's relevance algorithm. The value of this parameter is case-insensitive.	Yes
targetLocale.language	The locale language the API will use to try to localize the learning classification. The value of this parameter should be de, en, es, fr, or ja . These values correspond to the locales "de_DE", "en_US", "es_ES", "fr_FR", and "ja_JP". If the learning classification cannot be localized or if the target locale is not set, the API will use the source locale of the learning classification.	No
targetLocale.country	The locale country the API will use to try to localize the learning classification. The value of this parameter should be DE, US, ES, FR, or JP . These values correspond to the locales "de_DE", "en_US", "es_ES", "fr_FR", and "ja_JP". If the learning classification cannot be localized or if the target locale is not set, the API will use the source locale of the learning classification.	No
start	The start index of learning classifications for the page.	No (default = 0)
count	The number of learning classifications to include in the page. Please choose a reasonable value if overriding the default; the API will return an error if the requested page exceeds the 2 MB response size limit of the framework.	No (default = 100)

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'
```

```
'https://api.linkedin.com/v2/learningClassifications?q=keyword&keyword=software&targetLocale.language=en&targetLocale.country=US&start=0&count=20'
```

Sample response:

```
{
  "elements": [
    {
      "urn": "urn:li:skill:242",
      "owner": {
        "urn": "urn:li:organization:1337",
        "name": {
          "locale": {
            "country": "US",
            "language": "en"
          },
          "value": "LinkedIn"
        }
      },
      "name": {
        "locale": {
          "country": "US",
          "language": "en"
        },
        "value": "Software"
      },
      "type": "SKILL"
    },
    {
      "urn": "urn:li:lyndaCategory:7186",
      "owner": {
        "urn": "urn:li:organization:1337",
        "name": {
          "locale": {
            "country": "US",
            "language": "en"
          },
          "value": "LinkedIn"
        }
      },
      "name": {
        "locale": {
          "country": "US",
          "language": "en"
        },
        "value": "Software Development"
      },
      "type": "SUBJECT"
    },
    ...
  ],
  "paging": {
    "total": 15,
    "count": 20,
    "start": 0,
    "links": []
  }
}
```

Using the API: Advanced

This section explains several advanced topics about the LinkedIn Learning Search API.

Specifying the level of asset details

When you call `/v2/learningAssets` to retrieve a page of learning assets, given some search and relevance criteria, you can optionally set an `assetRetrievalCriteria.expandDepth` parameter:

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&assetRetrieval  
Criteria.expandDepth=1'
```

This parameter (defaults to 1) tells the API how many levels in the learning asset hierarchy should include asset *details*. A detailed learning asset has the following structure:

```
{  
  "urn": ...,  
  "type": ...,  
  "title": ...,  
  "details": {  
    ...  
  },  
  "contents": [  
    ...  
  ]  
}
```

The "urn", "type", and "title" fields are the basic metadata about the learning asset.

The "details" field is an object that includes detailed metadata about the learning asset - for example, its associated learning classifications, description, change timestamps, etc. (see the "Asset schema" section for the full schema reference).

The "contents" field is an array of sub-assets of the learning asset, each sub-asset of which is a learning asset itself. For example, a course contains chapters, and a chapter contains videos. The learning asset representing the course would look like the following (focusing on the nested structure and omitting other fields):

```
{  
  "type": "COURSE",  
  "contents": [  
    {  
      "asset": {  
        "type": "CHAPTER",  
        "contents": [  
          {  
            "asset": {  
              "type": "VIDEO",  
              "contents": [],  
            },  
            ...  
          }  
        ],  
      },  
      ...  
    }  
  ]  
}
```

In this learning asset hierarchy, you can think of the course as the first level of learning assets. The chapters, which are sub-assets of the course, are the second level of learning assets. The videos, which are sub-assets of the chapters, are the third level of learning assets.

With this explanation of the learning asset hierarchy and the earlier explanation of asset details in mind, we can refer back to the request we started with:

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&assetRetrieval  
Criteria.expandDepth=1'
```

Setting the "assetRetrievalCriteria.expandDepth" parameter to 1 tells the API to include asset details only for the first level of learning assets. Each learning asset in the "elements" array of the response would look like the following (assuming the learning asset is a course):

```
{  
  "urn": ...,  
  "type": COURSE,  
  "title": ...,  
  "details": {  
    ...  
  },  
  "contents": [  
    {  
      "asset": {  
        "urn": ...,  
        "type": "CHAPTER",  
        "title": ...,  
        "contents": [  
          {  
            "asset": {  
              "urn": ...,  
              "type": "VIDEO",  
              "title": ...,  
              "contents": []  
            }  
          },  
          ...  
        ]  
      }  
    },  
    ...  
  ]  
}
```

If you wish to include asset details for both the course and its chapters, you can tell the API to expand learning assets down to the second level:

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&assetRetrieval  
Criteria.expandDepth=2'
```

With the "assetRetrievalCriteria.expandDepth" parameter set to 2, each learning asset in the "elements" array of the response would look like the following (assuming the learning asset is a course):

```
{
  "urn": ...,
  "type": COURSE,
  "title": ...,
  "details": {
    ...
  },
  "contents": [
    {
      "asset": {
        "urn": ...,
        "type": "CHAPTER",
        "title": ...,
        "details": {
          ...
        },
        "contents": [
          {
            "asset": {
              "urn": ...,
              "type": "VIDEO",
              "title": ...,
              "contents": []
            }
          }
        ]
      }
    },
    ...
  ]
}
```

By setting the "assetRetrievalCriteria.expandDepth" parameter to 3 when calling the API, you can include asset details for all three levels - course, chapters, and videos.

When expanding learning assets down to deeper levels, take care that the larger responses do not exceed the 2 MB response size limit of the API framework. The "assetRetrievalCriteria.expandDepth" parameter is a convenience so that you can retrieve more information with fewer calls. But you can always retrieve the same nested information in multiple steps. For example, leaving the "assetRetrievalCriteria.expandDepth" parameter at its default value of 1, you can retrieve a page of learning assets representing courses, which will include asset details only for each course. Then for each chapter URN and video URN under the courses, you can call the endpoint that retrieves an individual learning asset, which will include asset details for the chapter or video.

Note that you can also set the "assetRetrievalCriteria.expandDepth" parameter to 0 when calling the API. This tells the API to omit all asset details:

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&assetRetrieval  
Criteria.expandDepth=0'
```

With the "assetRetrievalCriteria.expandDepth" parameter set to 0, each learning asset in the "elements" array of the response would look like the following (no asset details at any level):

```
{  
  "urn": ...,  
  "type": COURSE,  
  "title": ...,  
  "contents": [  
    {  
      "asset": {  
        "urn": ...,  
        "type": "CHAPTER",  
        "title": ...,  
        "contents": [  
          {  
            "asset": {  
              "urn": ...,  
              "type": "VIDEO",  
              "title": ...,  
              "contents": []  
            }  
          },  
          ...  
        ]  
      }  
    },  
    ...  
  ]  
}
```

Rolling up sub-asset classifications

A learning asset may be tagged with one or more learning classifications. For example, a course or video may be tagged with category and skill metadata. These learning classifications are included in the asset details (see the "Specifying the level of asset details" section).

When retrieving a learning asset with sub-assets, all the learning classifications that the sub-assets are tagged with are *rolled up* into the asset details for the top-level learning asset. For example, when retrieving a learning asset representing a course, the asset details for the course include the learning classifications that the chapters and videos are tagged with. In this case, if you set the "assetRetrievalCriteria.expandDepth" parameter to include the asset details for the chapter and video sub-assets as well, the asset details for each sub-asset will not include its individual learning classifications.

The API rolls up sub-asset classifications for two reasons. First, it makes sense for a learning asset to inherit all the learning classifications of its sub-assets. Second, rolling up sub-asset classifications lets the API de-duplicate them, reducing the response size if multiple sub-assets are tagged with the same learning classification.

Specifying the response fields

When calling any of the API endpoints, you can include a special request parameter that specifies the fields that will be in the response. Note that if the response fields you wish to specify are part of the asset details (see the "Specifying the level of asset details" section), you will also need to set the "assetRetrievalCriteria.expandDepth" parameter so that the required level of asset details will be included.

Parameter	Description	Required
fields	<p>A list of field <i>projections</i> that specify the fields that will be in the response. The projection syntax is explained below.</p> <p>Please note that if you set the "fields" parameter when calling an endpoint that retrieves a page of learning entities, the previous and next links in the paging metadata in the response (see the "Understanding the endpoints" section) will not include the "fields" parameter. You will need to add the same "fields" parameter to the generated links yourself.</p>	No

Field projection syntax:

- *field_name* to include the entire value of the field.
- *field_name:(sub_field_name)* to include only the value of the sub-field if the parent field is an object.
- *field_name:(\$*:(sub_field_name))* to include only the value of the sub-field if the parent field is an array of objects.
- *field_name,field_name* to include the values of multiple fields.

Here are four sample requests showing an example of each kind of field projection:

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'
```

```
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&fields=title'
```

Sample response:

```
{
  "elements": [
    {
      "title": {
        "locale": {
          "country": "US",
          "language": "en"
        },
        "value": "Java Essential Training for Students"
      }
    },
    {
      "title": {
        "locale": {
          "country": "US",
          "language": "en"
        },
        "value": "Scala Essential Training"
      }
    },
    ...
  ],
  "metadata": {
    ...
  },
  "paging": {
    ...
  }
}
```

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&fields=title:(value)'
```

Sample response:

```
{
  "elements": [
    {
      "title": {
        "value": "Java Essential Training for Students"
      }
    },
    {
      "title": {
        "value": "Scala Essential Training"
      }
    },
    ...
  ],
  "metadata": {
    ...
  },
  "paging": {
    ...
  }
}
```

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&fields=contents:  
($*:(asset:(urn)))'
```

Sample response:

```
{  
  "elements": [  
    {  
      "contents": [  
        {  
          "asset": {  
            "urn": "urn:li:lyndaChapter:(urn:li:lyndaCourse:375490,415232)"  
          }  
        },  
        {  
          "asset": {  
            "urn": "urn:li:lyndaChapter:(urn:li:lyndaCourse:375490,415237)"  
          }  
        },  
        ...  
      ],  
      ...  
    },  
    ...  
  ],  
  "metadata": {  
    ...  
  },  
  "paging": {  
    ...  
  }  
}
```

Sample request (using cURL):

```
curl -H 'Authorization: Bearer AQXt...'  
'https://api.linkedin.com/v2/learningAssets?q=criteria&assetFilteringCriteria.keyword=java&fields=urn,type,  
contents:($*:(asset:(urn,type)))'
```

Sample response:

```
{
  "elements": [
    {
      "urn": "urn:li:lyndaCourse:375490",
      "contents": [
        {
          "asset": {
            "urn": "urn:li:lyndaChapter:(urn:li:lyndaCourse:375490,415232)",
            "type": "CHAPTER"
          }
        },
        {
          "asset": {
            "urn": "urn:li:lyndaChapter:(urn:li:lyndaCourse:375490,415237)",
            "type": "CHAPTER"
          }
        },
        ...
      ],
      "type": "COURSE"
    },
    ...
  ],
  "metadata": {
    ...
  },
  "paging": {
    ...
  }
}
```

Schema Reference

Full reference for the schemas for a learning asset and a learning classification in the LinkedIn Learning Search API. The top-level *Asset* schema and *Classification* schema are first in their sections, followed by schemas for included objects, listed alphabetically. Schemas for objects included in both the *Asset* schema and the *Classification* schema are in the "Shared schemas" section.

Shared schemas

Locale:

Field	Type	Description
country	optional string	If present, an uppercase two-letter country code as defined by ISO-3166.
language	string	A lowercase two-letter language code as defined by ISO-639.
variant	optional string	If present, a vendor- or browser-specific code.

LocaleString:

Field	Type	Description
locale	<i>Locale</i>	The locale of the localized string.
value	string	The localized string.

NamedParty:

Field	Type	Description
name	<i>LocaleString</i>	The name of a person or organization, localized if available.
urn	string	The URN identifying a person or organization.

Asset schema

Asset - top-level schema for objects returned by `/v2/learningAssets:`

Field	Type	Description
contents	<i>SubAsset []</i>	The sub-assets of the learning asset. For example, a learning asset representing a course has sub-assets representing its chapters; a learning asset representing a chapter has sub-assets representing its videos.
details	optional <i>AssetDetails</i>	If present, the details about the learning asset. If this field is not present, it means the request did not specify retrieving the asset details (see the "Specifying the level of asset details" section).
title	<i>LocaleString</i>	The title of the learning asset, localized if available.
type	<i>AssetType</i>	The type of the learning asset.
urn	string	The URN of the learning asset. The URN is a unique identifier whose value should be treated as opaque. Do not use the URN to determine the type of the learning asset; use the "type" field instead.

AssetClassification:

Field	Type	Description
assigner	<i>NamedParty</i>	The person or organization who tagged the learning asset with the learning classification.
associatedClassification	<i>Classification</i>	The learning classification the learning asset is tagged with.
path	<i>Classification []</i>	The parent learning classifications of the associated learning classification.

AssetDetails:

Field	Type	Description
availability	<i>Availability</i>	The availability of the learning asset.
availableLocales	<i>Locale []</i>	The locales the learning asset is available in.
classifications	<i>AssetClassification []</i>	The learning classifications the learning asset is tagged with.
contributors	<i>Contributor []</i>	The contributors involved in the lifecycle of the learning asset - for example, authors or publishers.
description	optional <i>LocaleString</i>	If present, the text-only description of the learning asset, localized if available. Any HTML markup will be stripped from this description.
descriptionIncludingHtml	optional <i>LocaleString</i>	If present, the description - including any HTML markup - of the learning asset, localized if available.
images	<i>AssetImages</i>	The images that can be used to represent the learning asset.
lastUpdatedAt	long	The epoch time in milliseconds indicating when the learning asset was last updated.
level	optional <i>DifficultyLevel</i>	If present, the difficulty level of the learning asset.
publishedAt	long	The epoch time in milliseconds indicating when the learning asset was published.
relationships	<i>[]</i>	The value of this field is currently always an empty array. Future versions of the API may use this field to indicate relationships the learning asset has to other learning assets.
retiredAt	optional long	If present, the epoch time in milliseconds indicating when the learning asset was retired.
timeToComplete	optional <i>TimeSpan</i>	If present, the time span indicating how long the learning asset takes to complete.
urls	<i>AssetUrls</i>	The URLs that can be used to launch the learning asset.

AssetImages:

Field	Type	Description
primary	optional string	If present, the URL of the primary image associated with the learning asset. This image is usually high-resolution.

AssetType:

Symbol	Description
CHAPTER	The learning asset is a chapter.
COURSE	The learning asset is a course.
LEARNING_PATH	The learning asset is a learning path.
VIDEO	The learning asset is a video.

AssetUrls:

Field	Type	Description
aiccLaunch	optional string	If present, the launch URL of the learning asset that can be used to initiate AICC tracking in an AICC-compliant system.
webLaunch	optional string	If present, the launch URL of the learning asset in the LinkedIn Learning web application.

Availability:

Symbol	Description
AVAILABLE	The learning asset is available and active.
RETIRED	The learning asset is retired and no longer available.

ContributionType:

Symbol	Description
AUTHOR	The contributor is an author of the learning asset.
PUBLISHER	The contributor is a publisher of the learning asset.

Contributor:

Field	Type	Description
contributionType	<i>ContributionType</i>	The type of contribution the contributor made to the learning asset.
name	<i>LocaleString</i>	The name of the contributor, localized if available.
urn	string	The URN identifying the contributor.

DifficultyLevel:

Symbol	Description
BEGINNER	The learning asset assumes little to no understanding of the concepts related to the material.
INTERMEDIATE	The learning asset assumes basic understanding of the concepts related to the material.
ADVANCED	The learning asset assumes advanced understanding of the concepts related to the material.

SubAsset:

Field	Type	Description
asset	<i>Asset</i>	The learning asset that is a sub-asset of another learning asset.

TimeSpan:

Field	Type	Description
duration	long	The duration the time span represents.
unit	<i>TimeUnit</i>	The unit of time the duration refers to.

TimeUnit:

Symbol	Description
SECOND	A second.
MINUTE	A minute.
HOUR	An hour.

Classification schema

Classification - top-level schema for objects returned by /v2/learningClassifications:

Field	Type	Description
name	<i>LocaleString</i>	The name of the learning classification, localized if available.
owner	<i>NamedParty</i>	The person or organization who created the learning classification.
type	<i>ClassificationType</i>	The type of the learning classification.
urn	<i>string</i>	The URN of the learning classification. The URN is a unique identifier whose value should be treated as opaque. Do not use the URN to determine the type of the learning classification; use the "type" field instead.

ClassificationType:

Symbol	Description
LIBRARY	The learning classification is a library. <i>Libraries</i> - for example, "Technology" - are broad groups of learning assets.
SKILL	The learning classification is a skill. For example, "Object-Oriented Programming (OOP)" is a skill that may be taught by a course called "Learning Java".
SUBJECT	The learning classification is a subject. <i>Subjects</i> - for example, "Software Development" - are children of libraries and are groups of related topics.
TOPIC	The learning classification is a topic. <i>Topics</i> - for example, "Version Control" - are children of subjects and are focused topics.