

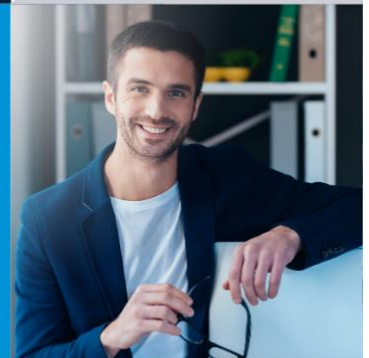
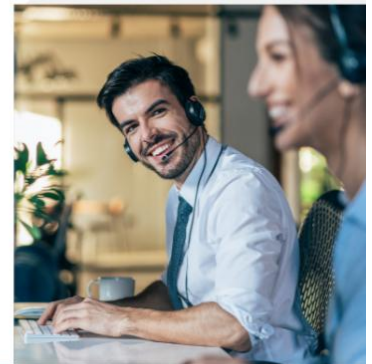
PINNACLE SERIES

by  EAGLE POINT
SOFTWARE

Pinnacle Series API Guide

A Technical Reference for Accessing Learning,
User, and Assessment Data

Version 5.0 – September 2026



Overview

The Pinnacle Series API enables secure, programmatic access to user activity, content usage, learning progress, and assessment results. It empowers organizations to:

- Build advanced dashboards
- Automate reporting
- Derive actionable insights to optimize user engagement and learning ROI

This guide explains how to use Pinnacle Series APIs to extract data for reporting on users, content, assignments, KnowledgeSmart results, and usage activity.

Who is this guide for?

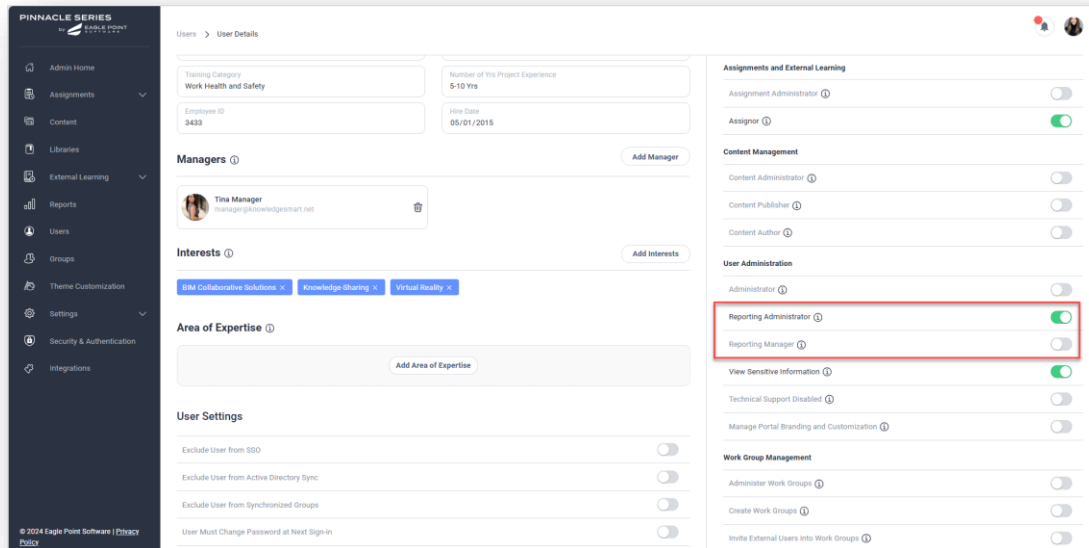
This guide supports developers, data analysts, and LMS administrators in accessing and using Pinnacle Series APIs effectively.

Contents

Overview	2
Tips & Best Practices	8
Summary Table of API Calls	8
Registered Users API - POST /reporting/registeredUsers	8
Logins API - POST /reporting/logins	9
Searches API - POST /reporting/searches	10
Resources Accessed API - POST /reporting/resourcesAccessed	11
Enrollments API - POST /reporting/courseEnrollments	12
Content API - POST /reporting/content	14
External Learning API - POST /reporting/externalLearning	16
KS Logins - POST /reporting/ksLogins	17
KS Assessments - POST /reporting/assessments	18
KS Assignments - POST reporting/assessmentAssignments	18
Models Section	20

Permissions

- To action API calls you will need either the reporting administrator or reporting manager permission.



Base URL

<https://reporting.pinnacleapis.com/api/swagger/ui>

Authentication

All API calls are documented on the [Swagger](#) page.

Authentication Flow:

1. **API Key: Tenant Info:** [GET /authinfo?email={yourEmail}](#)
2. **Bearer Token:** [POST /authorize](#) with:
 - Email (required)
 - Password (required)
 - TenantId (required)
 - Language (optional)
 - TimeOffset (optional)
 - TimezoneName (optional)
 - ReturnLongLivedToken (optional)

Required Headers:

authorization: Bearer {token}

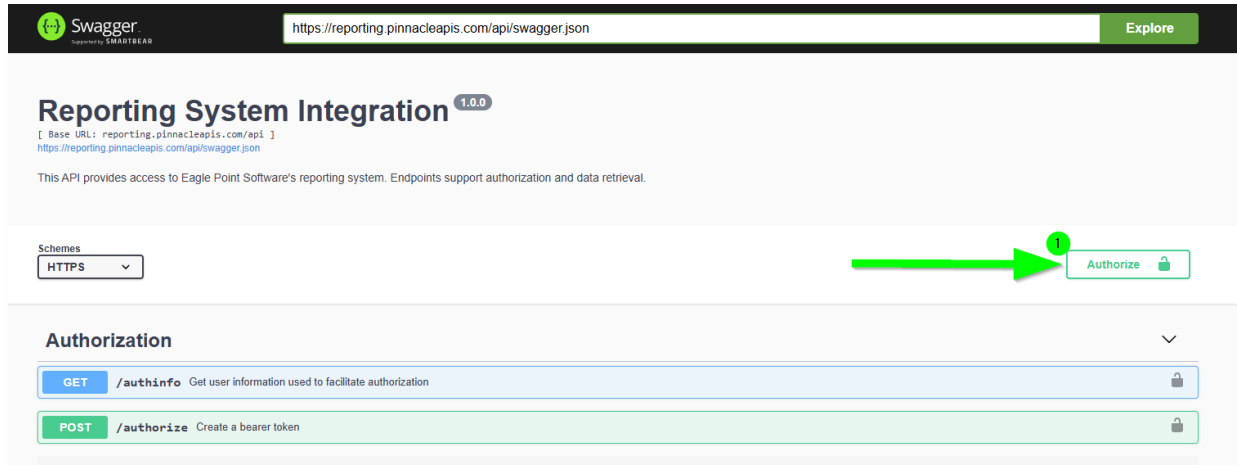
Bearer tokens are session-based and may expire. Store securely during session lifetime.

Note: Responses follow a consistent data model across endpoints, typically returning an array of values along with a totalCount property for pagination.

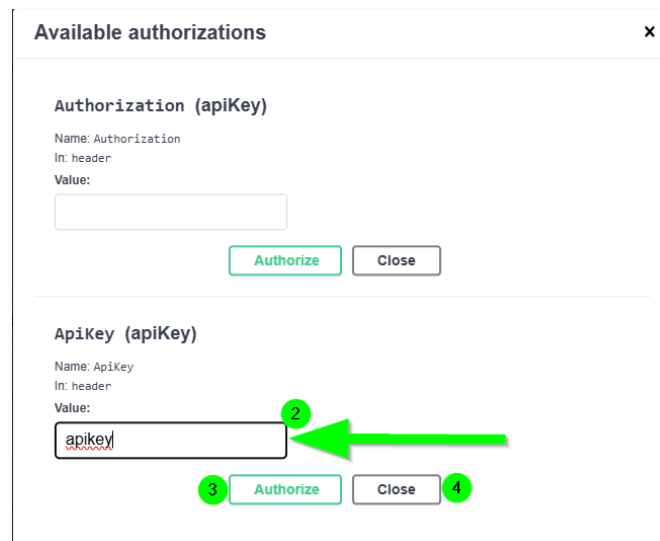
Trying Out the API in Swagger

Use the built-in Swagger interface to test authentication and reporting endpoints. Follow the steps below to simulate a typical workflow:

1. Select the 'Authorize' button



2. Pass the apiKey issued by the Eagle Point Software Team in the ApiKey (apiKey) > Value field (ref 2 below)



3. Select 'Authorize' (ref 3 above)
4. Click 'Close' (ref 4 above)
5. Click the **/authinfo** endpoint to expand it.
6. Click **Try it out**.
7. Enter your email address in the input field.

Authorization

GET /authinfo Get user information used to facilitate authorization

Get information for a user based on the provided email address

Parameters Try it out

Name	Description
email • required string (query)	The email address for which to retrieve user information.

email - The email address for which to retri

Authorization

GET /authinfo Get user information used to facilitate authorization

Get information for a user based on the provided email address

Parameters Cancel

Name	Description
email • required string (query)	The email address for which to retrieve user information.

lisa.taylor@eaglepoint.com

Execute

8. Click **Execute**.
9. Observe '200' response and retrieve the "tenantID". You'll need this in the next step.
10. Expand the /authorize endpoint.
11. Click **Try it out**.
12. Paste the "tenantId" and other required values into the request body. Below is an example of required formatting. (see ref 9 below)

Authorization

POST /authorize Create a bearer token

A bearer token is used to authorize use for most api calls. The bearer token should be placed in the Authorization header of the request.

Parameters Cancel

Name	Description
body object (body)	Edit Value Model

```
{
  "email": "your email address",
  "password": "password",
  "tenantId": "52a13ec2-4c85-1a85a488e8bf",
  "language": "en"
}
```

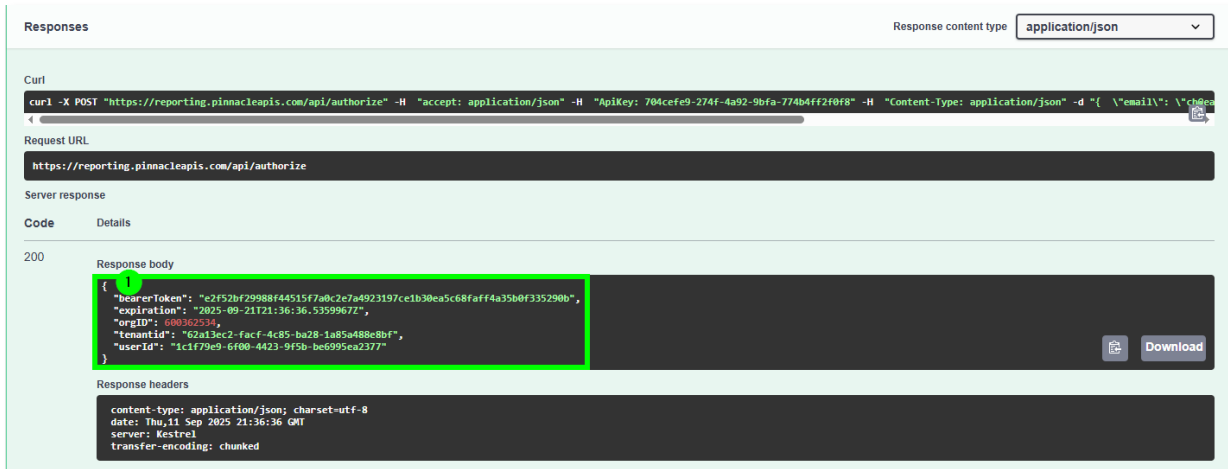
Cancel

Parameter content type
application/json

Execute Clear

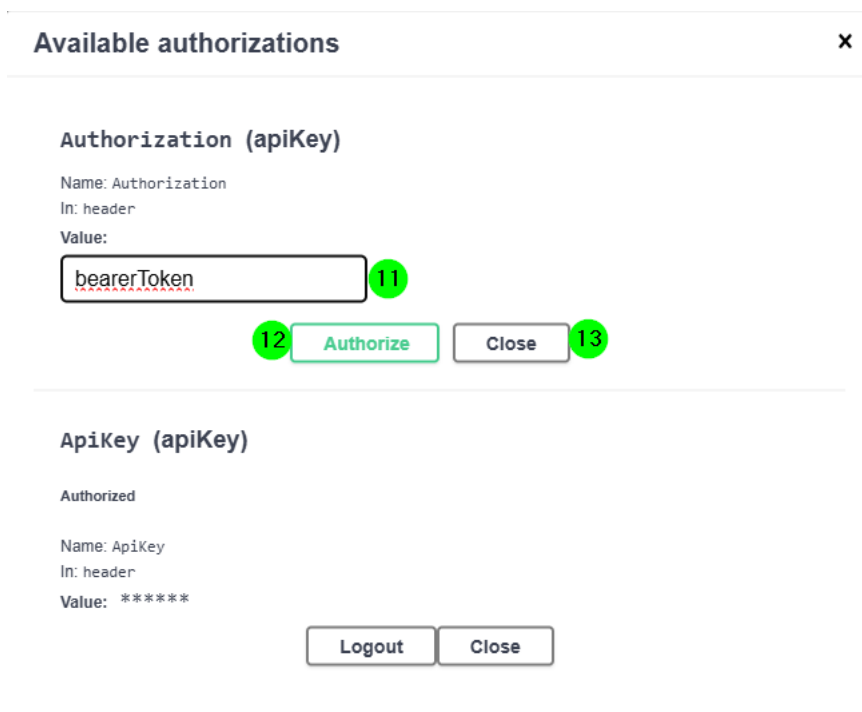
13. Click **Execute**.

14. Copy the bearerToken value from the response (see ref 1 below)



15. At the top right of the Swagger page, click the **Authorize** button again.

16. For **Authorization (apiKey)**, paste in your bearerToken and click **Authorize**. (ref 11 and 12 below). Then click close (ref 13 below).



17. **Reporting endpoints:** With the above authorizations you are now ready to expand and execute the variety of Reporting endpoints.

18. Navigate to any /reporting endpoint.

19. Click **Try it out**.

20. Enter the request body parameters as needed.

21. Click **Execute** to test the request and view the results.

**While the Swagger interface is a useful tool for testing and exploration, this manual process is not recommended for long-term or production use. It's best suited for quick validation or familiarizing yourself with the API endpoints.*

Responses

HTTP Status Code	Meaning	Recommended Action
200 OK	The request was successful.	
400 Bad Request	The API request was malformed or invalid.	Check for JSON syntax errors, missing required fields, or incorrect filter parameters.
403 Forbidden	The requester is authenticated but does not have permission to retrieve search data (e.g., insufficient API role or tenant restriction).	Verify API credentials and tenant scope.
500 Internal Server Error	An unexpected error occurred on the server side.	Retry later; if recurring, validate payload size and parameters or contact support.

Common Parameters

The **Parameters** section defines how API consumers can control, filter, and shape the data returned from each API call. Parameters like `PageSize` and `PageNumber` allow efficient handling of large datasets through pagination, while `OrderBy` and `OrderAsc` help sort results consistently for better readability and analysis. The `Filters` array enables precise targeting of records using conditions that support a range of operators (e.g., `=`, `!=`, `<`, `>`) and values, empowering developers and analysts to retrieve only the most relevant data for their reporting needs.

Example request body:

```
{
  "Paging": {
    "PageSize": 1000,
    "PageNumber": 1,
    "OrderBy": "assigned_to_display_name",
    "OrderAsc": true
  },
  "Filters": [
    { "ColumnName": "assigned_to_display_name", "Operator": "=", "ColumnValue": "John Doe" },
    { "ColumnName": "has_due_date", "Operator": "!=", "ColumnValue": true }
  ]
}
```

Examples:

- `PageSize: 100` (optimize payload size)
- `OrderBy: "created_date"` (sort chronologically)
- `Filters: { "ColumnName": "is_completed", "Operator": "=", "ColumnValue": true }` (show completed courses only)

Parameter	Required	Type	Description	Value
PageSize	Yes	Int	Records per page	Manage payload size
PageNumber	Yes	Int	Page index	Paginate results
OrderBy	No	String	Sort column	Clean reporting
OrderAsc	No	Bool	True=asc	Sort order
Filters	No	Array	Conditions combined with AND	Refine data

Important Notes:

- **Case-insensitive matching**
- **AND logic for filters**
- **No filtering on array fields**
- **Valid operators: =, !=, <, >, <=, >=**

Tips & Best Practices

- Use small PageSize for dev/testing, larger in production.
- Cache bearer tokens securely.
- Use OrderBy and OrderAsc for clean results.
- Filters: case-insensitive, AND logic, no array filtering.

Summary Table of API Calls

API	Category	Use Case
/reporting/registeredUsers	User Management	Identify active vs. deleted users
/reporting/searches	Discoverability	See what users are searching for
/reporting/resourcesAccessed	Item Interaction Analytics	Measure which resources are being used
/reporting/courseEnrollments	Learning Engagement	Track assignment and completion progress
/reporting/content	Content Analytics	Returns all content metadata for inventory, categorization, and audit reporting.

◆ Registered Users API - **POST /reporting/registeredUsers**

Returns all user records, including creation, deletion, anonymization, and login counts. For user lifecycle reporting and adoption analysis. **Example response below:**

```
{
  "data": [{
    "user_id": "string",
    "tenant_id": "string",
    "display_name": "string",
    "email": "string",
    "attributes": ["string"],
    "groups": ["string"],
    "managers": ["string"],
    "created_date": "2025-09-11T10:23:46.902Z",
    "deleted_date": "2025-09-11T10:23:46.902Z",
    "is_deleted": true,
    "is_anonymized": true,
    "number_of_logins": 0
  }],
  "totalCount": 0
}
```

Data Point	Data Type	Value
user_id	Text	Unique identifier for the user; use as the primary key to join with activity (logins), enrollments, or content access logs.
tenant_id	Text	Organization/tenant identifier; use to filter multi-tenant datasets.
display_name	Text	User's full display name (for UI/reporting).
email	Text	User's email; useful for lookups, communication, and ad-hoc filtering. Prefer user_id for joins.
attributes	Array<Text>	Custom user attributes (e.g., ["Job Profile: Architect","Region: EMEA"]). Use for segmentation, row-level security, and slicers. May be empty.
groups	Array<Text>	User's group memberships (e.g., ["EMEA > UK > London","BIM Managers"]). Use for access/visibility filters and group-based reporting. May be empty.
managers	Array<Text>	Manager identifiers/names/emails as strings (e.g., ["sarah.jones@org.com"]). Use for rollups and org hierarchy filtering. May be empty.
created_date	DateTime (ISO 8601)	When the user account was created (e.g., 2025-09-11T10:23:46.902Z). Use to count new registrations in a period.
deleted_date	DateTime (ISO 8601, nullable)	When the account was deleted/marked for deletion (e.g., 2025-09-11T10:23:46.902Z). Blank/null if the user is active.
is_deleted	Boolean	True if the user is deleted/inactive. Use to include/exclude from active-user metrics.
is_anonymized	Boolean	True if PII has been anonymized (e.g., GDPR/data retention). Handle joins/PII accordingly.
number_of_logins	Int	Lifetime login session count for this user as of extract time. Use with created_date and your Date table for rate metrics (e.g., logins per user).
totalCount	Int (top-level)	Total number of registered users matching the query. The totalCount value is not part of each user record. Instead, it appears outside the list of users and simply tells you how many users matched your search or filter. It helps with paging through large sets of results.

◇ Logins API - POST /reporting/logins

Returns user login/logout records with device info and Pinnacle version. For auditing access patterns and usage trends. **Example response below:**

```
{
  "data": [{
    "login_id": "string",
    "tenant_id": "string",
    "user_id": "string",
    "display_name": "string",
    "email": "string",
    "attributes": ["string"],
    "groups": ["string"],
    "managers": ["string"],
    "login_date": "2025-07-29T09:42:02.195Z",
    "logout_date": "2025-07-29T09:42:02.195Z",
    "pinnacle_version": "string",
    "device": "string",
    "ip_address": "string"
  }],
  "totalCount": 0
}
```

Data Point	Data Type	Value
login_id	Text	Unique identifier for this login session; supports user auditing and login tracking
tenant_id	Text	Organization ID; used to filter login activity by customer
user_id	Text	Internal user ID; allows joining login activity to user metadata
display_name	Text	Full name of the user; useful for dashboards and user recognition
email	Text	User's email address; allows filtering or reporting by user
attributes	Array of Text	Metadata describing user profile (e.g., department, job role)
groups	Array of Text	Lists user groups during login; supports group-level engagement reporting
managers	Array of Text	Associated managers at time of login; useful for roll-up reporting
login_date	DateTime (ISO 8601)	Timestamp of user login (e.g., 2025-07-16T15:59:49.166Z)
logout_date	DateTime (ISO 8601)	Timestamp of logout, useful for tracking session duration and logout patterns.
pinnacle_version	Text	Version of the Pinnacle platform accessed; helps track adoption and upgrade impact
device	Text	User device or browser details; helpful for usage and troubleshooting analysis
ip_address	Text	IP address at login; supports location insights, auditing, and security
TotalCount	Int	Total Number of Logins. totalCount is returned outside the data array. It indicates the total number of matching records for paging.

◇ Searches API - POST /reporting/searches

Returns user search history. For analyzing behavior and discoverability. **Example Response:**

```
{
  "data": [
    {
      "search_id": "string",
      "tenant_id": "string",
      "user_id": "string",
      "display_name": "string",
      "email": "string",
      "attributes": ["string"],
      "groups": ["string"],
      "managers": ["string"],
      "search_date": "2025-07-16T15:50:12.146Z",
      "search_term": "string"
    }
  ],
  "totalCount": 0
}
```

Data Point	Data Type	Value
search_id	Text	Unique identifier for the search event; useful for auditing and tracking search patterns
tenant_id	Text	Organization ID; allows filtering by customer account or tenant
user_id	Text	Unique ID of the user performing the search
display_name	Text	Full name of the user; useful for reporting and admin dashboards

email	Text	User's email address; used for filtering or user-specific reporting
attributes	Array of Text	Metadata tags like department, job role, or skill set; useful for analyzing search behavior
groups	Array of Text	User's assigned groups at time of search; allows group-based usage reporting
managers	Array of Text	Associated managers; enables filtering and hierarchy-based reporting
search_date	DateTime (ISO 8601)	Timestamp of when the search occurred (e.g., 2025-07-25T09:05:51.405Z)
search_term	Text	The exact term or phrase entered in the search field
totalCunt	Int	Total number of search records matching the query. totalCount is returned outside the data array. It indicates the total number of matching records for paging.

◇ Resources Accessed API – [POST /reporting/resourcesAccessed](#)

Returns individual content access events with metadata, user info, and referral sources. **Example response:**

```
{
  "data": [{
    "access_id": "string",
    "tenant_id": "string",
    "user_id": "string",
    "display_name": "string",
    "email": "string",
    "attributes": ["string"],
    "groups": ["string"],
    "managers": ["string"],
    "accessed_date": "2025-07-29T10:17:41.167Z",
    "content_id": "string",
    "content_name": "string",
    "content_type": "string",
    "publisher": "string",
    "topics": ["string"],
    "subtopics": ["string"],
    "libraries": ["string"],
    "referred_by": "string"
  }],
  "totalCount": 0
}
```

Data Point	Data Type	Value
access_id	Text	Unique identifier for this access event; critical for auditing and troubleshooting
tenant_id	Text	Organization ID; used to segment or filter access data by customer
user_id	Text	Unique ID of the user accessing the content
display_name	Text	Full name of the user for reporting and analysis
email	Text	User's email address; useful for communication and filtering
attributes	Array of Text	User metadata (e.g., job role, department); enables segmentation and advanced reporting
groups	Array of Text	Lists user group memberships at the time of access

managers	Array of Text	IDs or names of the user's managers; supports hierarchy-based reporting
accessed_date	DateTime (ISO 8601)	Date and time of access; format: YYYY-MM-DDTHH:mm:ss.sssZ (e.g., 2025-07-25T09:05:51.412Z)
content_id	Text	Unique ID of the accessed content
content_name	Text	Name or title of the accessed content
content_type	Text	Type of content (e.g., Document, Video, Workflow)
publisher	Text	Publisher or owner of the content
topics	Array of Text	High-level categories for the content
subtopics	Array of Text	More granular content categorization
libraries	Array of Text	Libraries that house the content; used for segmentation and filtering
referred_by	Text	Indicates the source of referral (e.g., search, direct link)
totalCount	Int	Total number of item interactions. It indicates the total number of matching records for paging.

Enrollments API - [POST /reporting/courseEnrollments](#)

Enrollments Returns all course enrollments with details, progress, quiz status. Example response:

```
{
  "data": [{
    "enrollment_id": "string",
    "tenant_id": "string",
    "assigned_to_user_id": "string",
    "assigned_to_display_name": "string",
    "assigned_to_email": "string",
    "assigned_date": "2025-07-29T09:43:58.376Z",
    "assigned_by_user_id": "string",
    "assigned_by_display_name": "string",
    "attributes": ["string"],
    "groups": ["string"],
    "managers": ["string"],
    "course_id": "string",
    "course_version": "string",
    "topics": ["string"],
    "subtopics": ["string"],
    "libraries": ["string"],
    "course_duration_minutes": 0,
    "assigned_minutes": 0,
    "viewed_minutes": 0,
    "publisher": "string",
    "percent_reviewed": 0,
    "learning_path_id": "string",
    "learning_path": "string",
    "status": "string",
    "quiz_status": "string",
    "due_date": "2025-07-29T09:43:58.376Z",
    "completed_date": "2025-07-29T09:43:58.376Z",
    "dropped_date": "2025-07-29T09:43:58.376Z",
    "is_self_enrolled": true,
    "has_due_date": true,
    "is_completed": true,
    "is_dropped": true,
    "is_scorm": true,
    "is_personalized": true,
    "assessment": "string"
  }],
  "totalCount": 0
}
```

Data Point	Data Type	Value
enrollment_id	Text	Unique identifier for each enrollment record; used for tracking and referencing
tenant_id	Text (was Int)	Organization ID; enables filtering by customer organization
assigned_to_user_id	Text	Unique ID of the user receiving the assignment
assigned_to_display_name	Text	Full display name of the assigned user; improves reporting readability
Assigned_to_email	Text	Email address of the assigned user; used for communication and filtering
Assigned_date	DateTime (ISO 8601)	Timestamp of assignment; format: YYYY-MM-DDTHH:mm:ss.sssZ (e.g., 2025-07-25T08:44:33.849Z)
assigned_by_user_id	Text	Unique ID of the assignor
assigned_by_display_name	Text	Full name of the person who made the assignment; provides audit trail context
attributes	Array of Text	Custom user metadata (e.g., job role, discipline); enables filtering and personalized reporting
groups	Array of Text	Groups the user belongs to; used for managing assignments and visibility
managers	Array of Text	Names or IDs of assigned managers; allows filtering by reporting structure
course_id	Text	Unique identifier for the assigned course
course_version	Text	Indicates the version of the assigned course
topics	Array of Text	Course-related topics; used for filtering, recommendations, and analytics
subTopics	Array of Text	More granular categorization under topics
libraries	Array of Text	Lists which libraries the course belongs to; used to segment and organize content
course_duration_minutes	Int	Total course length in minutes; supports time investment tracking
assigned_minutes	Int	Total minutes assigned to the user; helps gauge training workload
viewed_minutes	Int	Minutes the user has spent viewing the content;
publisher	Text	The name of the publisher
percent_reviewed	Int	The percentage progress the user has made with the enrolment.
learning_path_id	Text	The unique ID of the learning path in the database
learning_path	Text	The name of the learning path
status	Text	Current enrollment status (e.g., Assigned, In Progress, Completed)

quiz_status	Text	Pass/Fail
due_date	DateTime (ISO 8601)	Due date for enrollment completion
completed_date	DateTime (ISO 8601)	The date the user marked the enrolment as completed
dropped_date	DateTime (ISO 8601)	The date the user dropped the enrolment
Is_self_enrolled	Boolean	Indicator for self-enrollment
has_due_date	Boolean	Indicator for whether the enrolment had a due date or not.
Is_completed	Boolean	Indicator for whether the enrolment was marked as completed.
Is_dropped	Boolean	Indicator for whether the enrolment was dropped.
Is_scorm	Boolean	Indicator for whether the enrolment was for a SCORM course
Is_personalized	Boolean	Indicator for personalized learning enrolment
assessment	Text	The assessment name linked to the enrolment
totalcount	Int	Total count of enrollments. totalCount is returned outside the data array. It indicates the total number of matching records for paging.

Execute

Clear

Responses

Response content type application/json

Curl

```
curl -X POST "https://reporting.pinnacleapis.com/api/reporting/courseenrollments" -H "accept: application/json" -H "Authorization: e2f52bf29988f44515" -H "Content-Type: application/json"
```

Request URL

https://reporting.pinnacleapis.com/api/reporting/courseenrollments

Server response

Code Details

200

Response body

```
{
  "data": [
    {
      "enrollment_id": "B4C33FE6-B7FC-4716-83750BF92355",
      "tenant_id": "62A13EC2-4CB5-4AB5A488E8BF",
      "assigned_to_user_id": "D217D9EC-F4F6-8422-57B417B63216",
      "assigned_to_display_name": "Alyce Ofer",
      "assigned_to_email": "aofer3s1@fakeemaildomain.com",
      "assigned_date": "2024-09-19 14:54:06.000",
      "assigned_by_user_id": "1C1F79E9-6F90-4423-9F5B-BE6995EA2377",
      "assigned_by_display_name": "Allias:aofer3s1",
      "attributes": [
        "Test Attribute:2021-11-05T20:56:10Z"
      ],
      "groups": [
        "Alpha",
        "Echo",
        "Golf"
      ],
      "managers": [],
      "course_id": "F57EE832-547E-BE30-ACB0594620C5",
      "course": " ",
      "course_version": "1.0",
      "topics": [],
      "subtopics": [],
      "libraries": []
    }
  ]
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Thu, 11 Sep 2025 22:05:30 GMT
server: Kestrel
transfer-encoding: chunked
```

◇ Content API - POST /reporting/content

Returns all content metadata for inventory, categorization, and audit reporting.

```
{
  "data": [
    {
      "content_id": "string",
      "content_name": "string",
      "content_type": "string",
      "courses": ["string"],
      "learning_paths": ["string"],
      "topics": ["string"],
    }
  ]
}
```

```

"subtopics": ["string"],
"libraries": ["string"],
"url": "string",
"duration_minutes": 0,
"calculated_duration": 0,
"video_duration": "string",
"difficulty": "string",
"publisher": "string",
"created_date": "2025-07-25T11:19:45.838Z",
"created_by": "string",
"modified_date": "2025-07-25T11:19:45.838Z",
"modified_by": "string",
"last_accessed_date": "2025-07-25T11:19:45.838Z",
"last_accessed_by": "string",
"learning_tags": ["string"],
"command_tags": ["string"],
"keywords": ["string"],
"folder": "string",
"parent_folder": "string",
"folder_path": "string"
}],
"totalCount": 0
}

```

Data Point	Data Type	Value
content_id	Text	Unique identifier for the content item; used for joins and tracking
content_name	Text	Title of the content (e.g., course name, video title)
content_type	Text	Type of content (e.g., Document, Video, Workflow, Course)
courses	Array of Text	Lists courses this content is associated with; supports training structure analysis
learning_paths	Array of Text	Lists learning paths this content belongs to; helps track structured learning
topics	Array of Text	High-level categorization of content
subtopics	Array of Text	More granular categories under each topic
libraries	Array of Text	Libraries where this content is stored; helps control visibility and access
url	Text	Direct link to launch or preview the content
duration_minutes	Int	Manually entered duration (if provided by the content author)
calculated_duration	Int	System-estimated duration (based on length of video or workflow steps)
video_duration	Text	Actual video duration in hh:mm:ss format (if applicable); useful for media analysis.
difficulty	Text	Content difficulty level (e.g., Beginner, Intermediate, Advanced); useful for tagging
publisher	Text	Name of the person or team who published the content
created_date	DateTime (ISO 8601)	When the content was created (e.g., 2025-07-25T11:19:45.838Z)
created_by	Text	Username or ID of the user who created the content
modified_date	DateTime (ISO 8601)	When the content was last modified
modified_by	Text	Username or ID of the person who last modified the content
last_accessed_date	DateTime (ISO 8601)	Last date/time the content was accessed by a user
last_accessed_by	Text	Name or ID of the last user to access this content
learning_tags	Array of Text	Tags that link content to skill areas or job profiles; enhances search and automation
command_tags	Array of Text	Used for linking content to specific software commands or tools
keywords	Array of Text	Searchable keywords to improve discoverability
folder	Text	Immediate folder where content is stored
parent_folder	Text	Parent folder of the current folder
folder_path	Text	Full path of the folder hierarchy (e.g., Training > Civil 3D > Workflows)

totalCount	Int	Total number of content records returned. totalCount is returned outside the data array. It indicates the total number of matching records for paging.
------------	-----	--

◇ External Learning API - POST /reporting/externalLearning

Returns records of learning activity completed outside of Pinnacle Series (e.g., external courses, certifications, or third-party training), tied to a user profile. For consolidated reporting across internal and external learning sources.

```
{
  "data": [
    {
      "external_learning_id": "string",
      "tenant_id": "string",
      "user_id": "string",
      "display_name": "string",
      "email": "string",
      "user_type": "string",
      "groups": ["string"],
      "managers": ["string"],
      "attributes": ["string"],
      "status": "string",
      "created_date": "2026-09-09T15:34:39.537Z",
      "created_type": "string",
      "record_attributes": ["string"]
    }
  ],
  "totalCount": 0
}
```

Data Point	Data Type	Value
external_learning_id	Text	Unique identifier for the external learning record; use as the primary key for joining with other reporting endpoints.
tenant_id	Text	Organization/tenant identifier; use to filter multi-tenant datasets.
user_id	Text	Unique identifier for the user associated with this external learning record.
display_name	Text	User's full display name.
email	Text	User's email address.
user_type	Text	Type or classification of the user (e.g., Employee, Contractor, External).
groups	Array of Text	User's group memberships. May be empty.
managers	Array of Text	Manager identifiers, names, or emails as strings. May be empty.
attributes	Array of Text	Custom user attributes used for segmentation and filtering. May be empty.
status	Text	Current status of the external learning record (e.g., Active, Completed, Archived).
created_date	DateTime (ISO 8601)	When the external learning record was created.
created_type	Text	How the record was created (e.g., Manual, Import, API, Integration).
record_attributes	Array of Text	Additional custom attributes tied to this specific learning record. May be empty.
totalCount	Int (top-level)	Total number of external learning records matching the query. Returned outside the data array; used for paging.

KnowledgeSmart Assessment Data API Endpoints

These endpoints introduce access to KnowledgeSmart assessment data through the Pinnacle Reporting API layer. This is the first time assessment insights and learning assignment data can be

joined consistently in reporting. The goal is to help organizations identify skill variability, align targeted learning to real capability gaps, and measure improvement over time.

This update also enables a new **Assessment Reports** section within Peak Reporting, which provides visualizations for assessment activity, completion status, score trends, and triggered learning tags. All filters are aligned with the existing Pinnacle reporting model (attributes, groups, managers, user metadata), allowing assessment data to be analyzed alongside enrollments, content usage, and learning paths.

While the original KS endpoints remain available, adding these new endpoints directly within the Pinnacle API streamlines data access and enables clearer mapping of the *full personalized learning journey* — from assessment → assignment → engagement → completion.

There are still some data elements that remain outside the Pinnacle reporting layer (e.g., module-level performance, question-level analysis, training requests). Those will continue to be accessed through KnowledgeSmart's reporting tools and APIs until we expand the unified data model further.

◇ KS Logins - POST /reporting/ksLogins

Returns KnowledgeSmart user login activity data for reporting and analytics. Tracks login timestamps, user identifiers, and associated metadata (e.g., region, business line, job role).

```
{
  "data": [
    {
      "login_id": "string",
      "tenant_id": "string",
      "user_id": "string",
      "display_name": "string",
      "email": "string",
      "attributes": ["string"],
      "groups": ["string"],
      "managers": ["string"],
      "login_date": "2025-07-29T09:43:58.376Z",
      "page_visited": "string"
    }
  ],
  "totalCount": 0
}
```

Data Point	Data Type	Value
login_id	Text	Unique login record ID
tenant_id	Text	Organization ID
user_id	Text	Unique user identifier
display_name	Text	Display name of the user who logged in
email	Text	Email address of the user
attributes	Array of Text	Custom user metadata (job role, discipline, etc.)
groups	Array of Text	Groups the user belongs to
managers	Array of Text	Managers tied to the user (reporting visibility)
login_date	DateTime (ISO 8601)	Timestamp of login
page_visited (optional)	Text	Page or tool accessed during the session — usage analytics insight
totalCount	Int	Total record count (paging support)

◇ KS Assessments - POST /reporting/assessments

The **Assessments** endpoint enables customers to manage and analyze all available assessments, providing visibility into their configuration, status, and training tags for better reporting, automation, and learning alignment.

```
{
  "data": [
    {
      "assessment_id": "string",
      "assessment": "string",
      "description": "string",
      "off_the_shelf": true,
      "published": true,
      "enabled": true,
      "exam_mode": true,
      "clock_on": true,
      "time_limit_minutes": 0,
      "created_by_user_id": "string",
      "created_by": "string",
      "modified_date": "2025-07-29T09:43:58.376Z",
      "total_questions": 0,
      "training_tags": ["string"]
    }
  ],
  "totalCount": 0
}
```

Data Point	Data Type	Value
assessment_id	Text	Unique identifier for the assessment
assessment	Text	Assessment name
description	Text	Short description of the assessment
off_the_shelf	Boolean	Indicates whether this is Pinnacle/Eagle Point content
published	Boolean	Indicates visibility to end users
enabled	Boolean	Indicates if the assessment is active
exam_mode	Boolean	Forces pass/fail testing mode
clock_on	Boolean	Indicates if time-tracking is required
time_limit_minutes	Int	Assessment time limit
created_by_user_id	Text	User ID of the creator
created_by	Text	Display name of the creator
modified_date	DateTime (ISO 8601)	When the assessment was last updated
total_questions	Int	Number of questions included
training_tags	Array of Text	Tags for filtering, categorization
totalCount	Int	Total matching record count

◇ KS Assignments - POST /reporting/assessmentAssignments

Returns data on KnowledgeSmart assessment assignments, including user, status, and completion details. Enables reporting on assigned assessments, participation rates, and progress tracking. Focuses on what users have been assigned, regardless of completion status.

```

{
  "data": [
    {
      "invite_id": "string",
      "tenant_id": "string",
      "assessment_id": "string",
      "assessment": "string",
      "sent_to_user_id": "string",
      "sent_to_display_name": "string",
      "sent_to_email": "string",
      "attributes": ["string"],
      "groups": ["string"],
      "managers": ["string"],
      "sent_date": "2025-07-29T09:43:58.376Z",
      "sent_by_user_id": "string",
      "sent_by_display_name": "string",
      "expiry_date": "2025-07-29T09:43:58.376Z",
      "last_activity_date": "2025-07-29T09:43:58.376Z",
      "assessment_start_date": "2025-07-29T09:43:58.376Z",
      "assessment_completed_date": "2025-07-29T09:43:58.376Z",
      "number_of_logins": 0,
      "answered_questions": 0,
      "total_questions": 0,
      "percent_complete": 0,
      "score": 0,
      "time_taken_minutes": 0,
      "is_exam": true,
      "training_tags": ["string"],
      "is_pending": true,
      "is_completed": true,
      "is_deleted": false
    }
  ],
  "totalCount": 0
}

```

Data Point	Data Type	Value
invite_id	Text	Assignment record ID
tenant_id	Text	Customer organization ID
assessment_id	Text	Assessment being assigned
assessment	Text	Assessment name
sent_to_user_id	Text	ID of the user assigned
sent_to_display_name	Text	Display name of the assigned user
sent_to_email	Text	Email of the assigned user
attributes / groups / managers	Arrays of Text	Organizational context filters
sent_date	DateTime	Time assignment was sent
sent_by_user_id	Text	User ID of assignor
sent_by_display_name	Text	Assignor display name
expiry_date	DateTime	If assessment expires
last_activity_date	DateTime	Track recent engagement
assessment_start_date	DateTime	When user first opened exam
assessment_completed_date	DateTime	When user finished
number_of_logins	Int	Attempts / engagement signal
answered_questions	Int	Progress indicator
total_questions	Int	Max question count
percent_complete	Int	Completion percentage
score	Int	Score (if graded)
time_taken_minutes	Int	Duration of attempt
is_exam	Boolean	Mode indicator
training_tags	Array of Text	Useful for filtering / rollup dashboards
is_pending / is_completed / is_deleted	Boolean	Workflow state flags
totalCount	Int	Total count

Models Section

The Models section in Swagger UI documents the exact structure of the data objects that underpin these API responses. These models reflect the way Pinnacle Series stores this information and serve as a reference so technical teams can plan integrations with confidence, knowing exactly what fields, data types, and structures to expect.

Why this matters:

- These models help developers understand exactly what information will come back in API responses.
- They serve as a schema blueprint so teams can efficiently parse, store, and integrate this data into their own systems (e.g., databases, reporting tools, dashboards).
- The models also ensure that integrations handle fields like arrays (e.g., topics, groups, managers) and dates (accessed_date) properly.

How you might use this:

- To design integrations that expect consistent field names and data types.
- To map API response fields to internal reporting structures or analytics platforms
- To identify what data points are available for building dashboards or reports before writing queries or building connectors.
- You don't need to reference or use these model names directly — they serve as documentation aids, not API input requirements.
- Field names and types in these models match what you receive in API responses, making it easier to plan integrations, parsing, and data joins.