

## Tasks

# (Morgen) Tasks

Morgen allows you to manage tasks through a unified API. The following documentations explains how to operate on first-party Morgen tasks. Tasks follow a data model inspired by the [JSCalendar standard](#) ↗.

## List tasks

Returns a list of tasks, optionally filtered by update time.

```
fetch(  
  "https://api.morgen.so/v3/tasks/list?limit=<LIMIT>&updatedAfter=<UPDATED_AFTER>",  
  {  
    method: "GET",  
    headers: {  
      accept: "application/json",  
      Authorization: "ApiKey <API_KEY>",  
    },  
  },  
);
```

Parameter	Type	Default	Required	Description
<code>limit</code>	Query	100	No	Maximum number of tasks to return (max 100).
<code>updatedAfter</code>	Query	-	No	Only return tasks updated/created after this ISO 8601 datetime.



The Tasks API is primarily designed for creating tasks in Morgen from unsupported integrations using apps like Zapier. Listing tasks costs 10 points per request. See [Rate Limits](#) for details.

## Task schema

Here is an example response for the `/tasks/list` endpoint:

```
{
  "data": {
    "tasks": [
      {
        "@type": "Task",

        // Morgen ID for task
        "id": "WyJBUU1rQURaa1lXWXp0e1...",

        // Morgen ID for the account
        "accountId": "640a62c9aa5b7e06cf420000",

        "integrationId": "morgen",

        // Task list this task belongs to
        "taskListId": "default",

        "created": "2023-02-28T11:50:57",
        "updated": "2023-02-28T17:56:44",

        "title": "Review quarterly report",
        "description": "Review and provide feedback on Q4 report",
        "descriptionContentType": "text/plain",

        // Due date in LocalDateTime format
        "due": "2023-03-15T17:00:00",
        "timeZone": "Europe/Berlin",

        // Estimated time to complete (ISO 8601 duration)
        "estimatedDuration": "PT2H",

        // Priority: 0 = undefined, 1 = highest, 9 = lowest
        "priority": 1,

        // Task completion status
        "progress": "needs-action",

        // Position within the task list (for ordering)
        "position": 0,

        // Relations to other tasks (for subtasks)
        "relatedTo": {
          "parent-task-id": {
            "@type": "Relation",
            "relation": { "parent": true }
          }
        }
      },
    ],
  },
}
```

```
// Array of tag IDs assigned to this task
"tags": ["550e8400-e29b-41d4-a716-446655440000"],

// Morgen-specific derived fields (read-only)
"morgen.so:derived": {
  "scheduled": true
}
},
// Label definitions for task metadata
"labelDefs": [...],
// Spaces/workspaces the tasks belong to
"spaces": [...]
}
}
```

## Task Fields Reference

Field	Type	Description
@type	String	Always "Task"
id	String	Morgen's unique identifier for the task
accountId	String	ID of the account the task belongs to
integrationId	String	Integration provider ( "morgen" )
taskListId	String	ID of the task list containing this task
title	String	Task title/summary
description	String	Task description (plain text or HTML)
descriptionContentType	String	Either "text/plain" or "text/html"
due	String	Due date in LocalDateTime format (e.g., "2023-03-15T17:00:00" )
timeZone	String	IANA timezone for the due date
estimatedDuration	String	Estimated duration in ISO 8601 format (e.g., "PT2H" )
priority	Number	Priority level: 0 (undefined), 1 (highest) to 9 (lowest)

Field	Type	Description
<code>progress</code>	String	Status: <code>"needs-action"</code> , <code>"in-process"</code> , <code>"completed"</code> , <code>"failed"</code> , <code>"cancelled"</code>
<code>position</code>	Number	Sort order within the task list
<code>relatedTo</code>	Object	Relations to other tasks (for subtask hierarchies)
<code>tags</code>	Array	Array of tag IDs assigned to this task (see <a href="#">Tags</a> )
<code>created</code>	String	Creation timestamp (UTC)
<code>updated</code>	String	Last update timestamp (UTC)



Additional fields might be added in the future. If you are storing tasks in your database, please make sure to ignore unknown fields.

## Get a task

Retrieve a single task by its ID.

```
fetch("https://api.morgen.so/v3/tasks?id=<TASK_ID>", {
  method: "GET",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
});
```

Parameter	Type	Default	Required	Description
<code>id</code>	Query	-	Yes	The Morgen ID of the task to retrieve.

## Response

```
{
```

```
"data": {
  "task": {
    "@type": "Task",
    "id": "WyJBUU1rQURaa1lXWxp0e1...",
    "title": "Review quarterly report",
    ...
  },
  "labelDefs": [...]
}
```

## Create a task

Create a new task in Morgen.

```
fetch("https://api.morgen.so/v3/tasks/create", {
  method: "POST",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
  body: JSON.stringify({
    title: "Review quarterly report",
    description: "Review and provide feedback",
    due: "2023-03-15T17:00:00",
    timeZone: "Europe/Berlin",
    priority: 1,
  }),
});
```

## Request Body Fields

Field	Type	Required	Description
<code>title</code>	String	Yes	Task title (minimum 1 character).
<code>description</code>	String	No	Task description.
<code>descriptionContentType</code>	String	No	<code>"text/plain"</code> or <code>"text/html"</code> . Default: <code>"text/plain"</code> .

Field	Type	Required	Description
<code>due</code>	String	No	Due date in LocalDateTime format (exactly 19 characters: <code>YYYY-MM-DDTHH:mm:ss</code> ).
<code>timeZone</code>	String	No	IANA timezone for the due date.
<code>estimatedDuration</code>	String	No	Estimated duration in ISO 8601 format.
<code>taskListId</code>	String	No	ID of the task list to add the task to.
<code>priority</code>	Number	No	Priority: 0 (undefined), 1 (highest) to 9 (lowest).
<code>progress</code>	String	No	Initial status: <code>"needs-action"</code> or <code>"completed"</code> .
<code>relatedTo</code>	Object	No	Relations for creating subtasks.
<code>tags</code>	Array	No	Array of tag IDs to assign to the task (see <a href="#">Tags</a> ).

## Response

```
{
  "data": {
    "id": "WyJBUU1rQURaa1lXWXp0e1..."
  }
}
```

## Creating Subtasks

To create a subtask, use the `relatedTo` field to specify the parent task:

```
{
  "title": "Subtask title",
```

```
"relatedTo": {
  "<PARENT_TASK_ID>": {
    "@type": "Relation",
    "relation": { "parent": true }
  }
}
```

## Update a task

Update an existing task. Only include the fields you want to change.

```
fetch("https://api.morgen.so/v3/tasks/update", {
  method: "POST",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
  body: JSON.stringify({
    id: "<TASK_ID>",
    title: "Updated title",
  }),
});
```

## Request Body Fields

Field	Type	Required	Description
<code>id</code>	String	Yes	The Morgen ID of the task to update.
<code>title</code>	String	No	Updated task title.
<code>description</code>	String	No	Updated description.
<code>due</code>	String	No	Updated due date.
<code>timeZone</code>	String	No	Updated timezone.
<code>taskListId</code>	String	No	Move task to a different task list.
<code>priority</code>	Number	No	Updated priority (0-9).
<code>progress</code>	String	No	Updated status.

Field	Type	Required	Description
<code>tags</code>	Array	No	Array of tag IDs (see <a href="#">Tags</a> ).



Updates are patch updates. Only the fields you provide will be updated. All other fields remain unchanged.

## Response

Returns HTTP 204 No Content on success.

## Move a task

Reorder a task within its list or change its parent (for subtask hierarchies).

```
fetch("https://api.morgen.so/v3/tasks/move", {
  method: "POST",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
  body: JSON.stringify({
    id: "<TASK_ID>",
    previousId: "<PREVIOUS_TASK_ID>",
    parentId: "<PARENT_TASK_ID>",
  }),
});
```

## Request Body Fields

Field	Type	Required	Description
<code>id</code>	String	Yes	The Morgen ID of the task to move.
<code>previousId</code>	String	No	ID of the task this should appear after. Use <code>null</code> to move to first position.
<code>parentId</code>	String	No	ID of the parent task. Use <code>null</code> to move to root level.



## Response

Returns HTTP 204 No Content on success.

## Delete a task

Delete a task permanently.

```
fetch("https://api.morgen.so/v3/tasks/delete", {
  method: "POST",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
  body: JSON.stringify({
    id: "<TASK_ID>",
  }),
});
```

## Request Body Fields

Field	Type	Required	Description
<code>id</code>	String	Yes	The Morgen ID of the task to delete.

## Response

Returns HTTP 204 No Content on success.

## Close a task

Mark a task as completed.

```
fetch("https://api.morgen.so/v3/tasks/close", {
  method: "POST",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
```

```
body: JSON.stringify({
  id: "<TASK_ID>",
}),
});
```

## Request Body Fields

Field	Type	Required	Description
<code>id</code>	String	Yes	The Morgen ID of the task to close.
<code>occurrenceStart</code>	String	No	For recurring tasks: ISO 8601 datetime of the specific occurrence to close.

## Response

Returns HTTP 204 No Content on success.

## Reopen a task

Mark a completed task as not completed.

```
fetch("https://api.morgen.so/v3/tasks/reopen", {
  method: "POST",
  headers: {
    accept: "application/json",
    Authorization: "ApiKey <API_KEY>",
  },
  body: JSON.stringify({
    id: "<TASK_ID>",
  }),
});
```

## Request Body Fields

Field	Type	Required	Description
<code>id</code>	String	Yes	The Morgen ID of the task to reopen.

Field	Type	Required	Description
<code>occurrenceStart</code>	String	No	For recurring tasks: ISO 8601 datetime of the specific occurrence to reopen.

## Response

Returns HTTP 204 No Content on success.

## Common Validation Errors

### Invalid Due Date Format

#### Error Message:

```
due must be a valid ISO 8601 date string
```

**Cause:** The due date is not in the correct format.

**Solution:** Use exactly 19 characters in LocalDateTime format: `YYYY-MM-DDTHH:mm:ss`

**Example:** `"2023-03-15T17:00:00"` (not `"2023-03-15"` or `"2023-03-15T17:00:00Z"`)

### Invalid Priority Value

#### Error Message:

```
priority must not be greater than 9
priority must not be less than 0
```

**Cause:** Priority value is outside the valid range.

**Solution:** Use a value between 0 and 9, where 0 is undefined and 1 is highest priority.

# Missing Required Title

## Error Message:

```
title must be longer than or equal to 1 characters
```

**Cause:** Task title is empty or missing.

**Solution:** Provide a title with at least 1 character when creating a task.

Last updated on February 2, 2026

---

Morgen Developer Documentation - Morgen AG © 2020 - 2026 All rights reserved.