

API docs

• • •

目錄

1. Get Specific Web Chat
2. Get LLM Usage Statistics
3. Get AI Assistant Statistics
4. Enable or Disable Evaluation for Specific AI Assistant
5. Send Message (Streaming)
6. Send Message (Create)
7. Create New Conversation
8. Get Message List
9. Get Specific Message
10. Get Conversation List
11. Get Specific Conversation
12. Get All Text Nodes of an AI Assistant's Files
13. Retrieve a specific text node of an AI assistant's file
14. Search Test
15. List Parsers Supported by File Type
16. List Contacts
17. Create Contact
18. Get Contact Details
19. Update Contact
20. Update Contact
21. Delete Contact
22. Create New Attachment (Integration)
23. Get a Pre-signed URL for File Upload (Create/Get)
24. Presigned Attachment Upload
25. Create Conversation Attachment

API Error: Claude's response exceeded the 32000 output token maximum. To configure this behavior, set the CLAUDE_CODE_MAX_OUTPUT_TOKENS environment variable.

API call example (Shell)

Get Specific Web Chat

GET

/api/web-chats/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this web chat.

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/web-chats/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request
data before execution.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/web-chats/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/web-chats/550e8400-e29b-41d4-
a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/web-
chats/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "name": string
  "avatar"??: string (uri) // Optional
  "logo"??: string (uri) // Optional
  "description"??: string // Optional
```

```
"cover"?: string (uri) // Optional
"backUrl"?: string // Optional
"isActive"?: boolean // Optional
"enableSpeech": boolean
"displayGreeting": string
"displayConversationStarters": [
  string
]
"theme": {
{
  "primaryColor"?: string // Optional
  "navbarTextColor"?: string // Optional
  "conversationBackgroundColor"?: string // Optional
  "chatbotMessageTextColor"?: string // Optional
  "userMessageTextColor"?: string // Optional
  "chatbotMessageBackgroundColor"?: string // Optional
  "userMessageBackgroundColor"?: string // Optional
  "chatbotMessageBackgroundShadowEnabled"?: boolean // Optional
  "userMessageBackgroundShadowEnabled"?: boolean // Optional
}
}
"darkTheme": {
{
  "primaryColor"?: string // Optional
  "navbarTextColor"?: string // Optional
  "conversationBackgroundColor"?: string // Optional
  "chatbotMessageTextColor"?: string // Optional
  "userMessageTextColor"?: string // Optional
  "chatbotMessageBackgroundColor"?: string // Optional
  "userMessageBackgroundColor"?: string // Optional
  "chatbotMessageBackgroundShadowEnabled"?: boolean // Optional
  "userMessageBackgroundShadowEnabled"?: boolean // Optional
}
}
"enableFileUpload"?: boolean // Optional
"enableDisplayCitations"?: boolean // Optional
"enableDisplayToolResults"?: boolean // Optional
"enableShareConversation"?: boolean // Optional
"enableLocation"?: boolean // Optional
"enableShowPoweredBy"?: boolean // Optional
"poweredByLogo"?: string (uri) // Optional
"enableAnonymous"?: boolean // Optional
"enableDisplayPreviousConversation"?: boolean // Optional
"enableDisplayConversationHistory"?: boolean // Optional
"enableConversationTimer"?: boolean // Optional
"conversationTimerDuration"?: integer // Timer timeout duration, range
3-60 minutes (Optional)
"enableAutoNewConversationOnTimeout"?: boolean // Optional
```

```
"accessType":  
{  
}  
"enableThemeModeToggle"??: boolean // Optional  
"enableDisplayFontSizeSwitch"??: boolean // Optional  
"defaultThemeMode"??: // Optional  
{  
}  
"enableDownloadCitations"??: boolean // Optional  
"customDomain"??: // Optional  
{  
    "id": string (uuid)  
    "domain": string  
    "status":  
    {  
    }  
}  
"enableDisplayChatbotAvatar"??: boolean // Optional  
"enableDisplayChatbotName"??: boolean // Optional  
"inbox": object  
}
```

Response Example Value

```
{  
    "id": "123e4567-e89b-12d3-a456-426614174000",  
    "name": "Test doc",  
    "avatar": "https://example.com/static/images/default-chatbot-  
    avatar.png",  
    "logo": null,  
    "description": "",  
    "cover": null,  
    "backUrl": "",  
    "isActive": true,  
    "enableSpeech": false,  
    "displayGreeting": null,  
    "displayConversationStarters": [],  
    "theme": {  
        "primaryColor": "#3854d8",  
        "dialogColor": "#ddeaf6",  
        "navbarTextColor": "#ffffff"  
    },  
    "enableFileUpload": true,  
    "enableDisplayCitations": true,  
    "enableShareConversation": false,  
    "enableLocation": false,  
    "enableShowPoweredBy": true,
```

```
"poweredByLogo": null,  
"accessType": "web",  
"inbox": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Test doc",  
    "signAuth": {  
        "id": "6ba7b810-9dad-11d1-80b4-00c04fd430c8",  
        "signSource": {  
            "id": "7c9e6679-7425-40de-944b-e07fc1f90ae7",  
            "accessType": "keycloak"  
        },  
        "signParams": {  
            "keycloak": {  
                "url": "example.com",  
                "realm": "example",  
                "clientId": "example"  
            }  
        }  
    }  
}
```

Get LLM Usage Statistics

GET

/api/chatbots/{chatbotPk}/llm-usage-statistics/

Parameters

Parameter Name	Required	Type	Description
chatbotPk	V	string	A UUID string identifying this Chatbot ID
endDate	X	string	End date, format: YYYY-MM-DD
startDate	V	string	Start date, format: YYYY-MM-DD
timeGranularity	V	string	Time granularity (day/month)

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/chatbots/550e8400-e29b-
41d4-a716-446655440000/llm-usage-statistics/?
endDate=example&startDate=example&timeGranularity=day"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request
data before execution.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-446655440000/llm-usage-statistics/?endDate=example&startDate=example&timeGranularity=day", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/llm-usage-statistics/?
endDate=example&startDate=example&timeGranularity=day"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/llm-usage-statistics/?
endDate=example&startDate=example&timeGranularity=day", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
[
{
    "date": string (date)
```

```
    }
]
```

Response Example Value

```
[
  [
    [
      {
        "totalWordsCount": 511,
        "month": "2025-03-01",
        "llmDetails": [
          {
            "chatbotId": "5b646600-f6e0-4f07-aec7-0207a26c0a9b",
            "wordsCount": 511,
            "llmModelId": "d4282719-f50e-4a82-88d0-953e202a3383",
            "llmmodelName": "GPT-4o-2024-08-06",
            "llmModelProvider": "openai"
          }
        ]
      }
    ]
]
```

Status Code: **400**

Response Schema Example

```
{
  "detail"??: string // Optional
  "valid_types"??: [ // Optional
    string
  ]
}
```

Response Example Value

```
[
  [
    [
      {
        "totalWordsCount": 511,
        "month": "2025-03-01",
        "llmDetails": [
          {

```

```
        "chatbotId": "5b646600-f6e0-4f07-aec7-0207a26c0a9b",
        "wordsCount": 511,
        "llmModelId": "d4282719-f50e-4a82-88d0-953e202a3383",
        "llmmodelName": "GPT-4o-2024-08-06",
        "llmModelProvider": "openai"
    }
]
}
]
]
```

Get AI Assistant Statistics

GET

/api/chatbots/{chatbotPk}/statistics/

Parameters

Parameter Name	Required	Type	Description
chatbotPk	V	string	A UUID string identifying this Chatbot ID
datetimeFrom	X	string	Start time
datetimeTo	X	string	End time
timeGranularity	X	string	Time granularity (hour/day/month)

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/chatbots/550e8400-e29b-
41d4-a716-446655440000/statistics/?
datetimeFrom=example&datetimeTo=example&timeGranularity=example"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request
data before execution.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-
a716-446655440000/statistics/?  
datetimeFrom=example&datetimeTo=example&timeGranularity=example",
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/statistics/?
datetimeFrom=example&datetimeTo=example&timeGranularity=example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/statistics/?
datetimeFrom=example&datetimeTo=example&timeGranularity=example", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
[
{
    "datetime": string (timestamp)
    "messagesCount": integer // Optional
```

```
"conversationsCount?: integer // Optional  
"avgMessagesPerConversation": number (double)  
"avgLikesPerFeedback": number (double)  
}  
]
```

Response Example Value

```
[  
 {  
 "datetime": "Response string",  
 "messagesCount": 456,  
 "conversationsCount": 456,  
 "avgMessagesPerConversation": 456,  
 "avgLikesPerFeedback": 456  
 }  
]
```

Enable or Disable Evaluation for Specific AI Assistant

PATCH </api/chatbots/{id}/enable-evaluation/>

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Chatbot.

Request Body

Request Parameters

Field	Type	Required	Description
enableEvaluation	boolean	No	

Request Schema Example

```
{  
  "enableEvaluation"??: boolean // Optional  
}
```

Request Example Value

```
{  
  "enableEvaluation": true  
}
```

Code Examples

Shell/Bash

```
# API call example (Shell)  
curl -X PATCH "https://api.maiagent.ai/api/chatbots/550e8400-e29b-  
41d4-a716-446655440000/enable-evaluation/"  
  -H "Authorization: Api-Key YOUR_API_KEY"  
  -H "Content-Type: application/json"  
  -d '{  
    "enableEvaluation": true  
  }'  
  
# Please make sure to replace YOUR_API_KEY and verify the request  
data before execution.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request body (payload)
const data = {
  "enableEvaluation": true
};

axios.patch("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-446655440000/enable-evaluation/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/enable-evaluation/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request body (payload)
data = {
    "enableEvaluation": true
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>patch("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-
a716-446655440000/enable-evaluation/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "enableEvaluation": true
    ]
]);

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
{  
  "enableEvaluation"??: boolean // Optional  
}
```

Response Example Value

```
{  
  "enableEvaluation": false  
}
```

Knowledge Base (New Version)

Create Knowledge Base

POST [/api/knowledge-bases/](#)

Request Body

Request Parameters

Field	Type	Required	Description
embeddingModel	string (uuid)	Yes	
rerankerModel	string (uuid)	Yes	
name	string	Yes	
description	string	No	
numberOfRetrievedChunks	integer	No	Number of retrieved reference chunks, default is 12, minimum is 1
sentenceWindowSize	integer	No	RAG augmented sentence window size, default is 2, minimum is 0
enableHyde	boolean	No	Enable HyDE, default is False

Field	Type	Required	Description
similarityCutoff	number (double)	No	Similarity cutoff, default is 0.0, range is 0.0-1.0
enableRerank	boolean	No	Enable reranking, default is True
chatbots	array[IdName]	No	
groups	array[IdName]	No	

Request Structure Example

```
{
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description"?: string // Optional
  "numberOfRetrievedChunks"?: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize"?: integer // RAG sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde"?: boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff"?: number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank"?: boolean // Enable rerank, default is True (Optional)
  "chatbots"?: [ // Optional
    {
      "id": string (uuid)
    }
  ]
  "groups"?: [ // Optional
    {
      "id": string (uuid)
    }
  ]
}
```

Request Sample Value

```
{
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example string",
```

```
"numberOfRetrievedChunks": 123,  
"sentenceWindowSize": 123,  
"enableHyde": true,  
"similarityCutoff": 123,  
"enableRerank": true,  
"chatbots": [  
  {  
    "id": "550e8400-e29b-41d4-a716-446655440000"  
  }  
,  
  "groups": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000"  
    }  
  ]  
}
```

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Sample Name",
  "description": "Sample String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
};
```



Python

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
            "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Example Name",
            "description": "Example String",
            "numberOfRetrievedChunks": 123,
            "sentenceWindowSize": 123,
            "enableHyde": true,
            "similarityCutoff": 123,
            "enableRerank": true,
            "chatbots": [
                [
                    [
                        "id": "550e8400-e29b-41d4-a716-446655440000"
                    ]
                ],
                "groups": [
                    [
                        [
                            "id": "550e8400-e29b-41d4-a716-446655440000"
                        ]
                    ]
                ]
            ]
        ]
    ]);
}

$data = json_decode($response→getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e→getMessage();
}
?>
```

Response Body

Status Code: 201

Example Response Structure

```
{
  "id": string (uuid)
  "user"?: string (uuid) // Optional
  "organization"?: string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description"?: string // Optional
  "numberOfRetrievedChunks"?: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize"?: integer // RAG sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde"?: boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff"?: number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank"?: boolean // Enable rerank, default is True (Optional)
  "labels": [
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "chatbots"?: [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "groups"?: [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "filesCount": integer // Returns the number of ChatbotFiles under this knowledge base (excluding processed files)
  "createdAt": string (timestamp)
```

```
    "updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "filesCount": 456,
  "createdAt": "Response string",
  "updatedAt": "Response string"
}
```

- ### List all knowledge bases**

GET </api/knowledge-bases/>

Parameters

Parameter Name	Required	Type	Description
embeddingModel	X	string	
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	
rerankerModel	X	string	

Code Example

Shell/Bash

```
# Call API Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/?
embeddingModel=550e8400-e29b-41d4-a716-
44665544000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-
41d4-a716-44665544000"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/?embeddingModel=550e8400-e29b-41d4-a716-44665544000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-41d4-a716-44665544000", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/?embeddingModel=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-41d4-a716-446655440000"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/?embeddingModel=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-
41d4-a716-446655440000", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Structure Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "user"??: string (uuid) // Optional
      "organization"??: string (uuid) // Optional
      "embeddingModel": string (uuid)
    }
  ]
}
```

```

    "rerankerModel": string (uuid)
    "name": string
    "description?": string // Optional
    "numberOfRetrievedChunks?": integer // Number of retrieved reference
    chunks, default is 12, minimum is 1 (Optional)
    "sentenceWindowSize?": integer // RAG augmented sentence window size,
    default is 2, minimum is 0 (Optional)
    "enableHyde?": boolean // Enable HyDE, default is False (Optional)
    "similarityCutoff?": number (double) // Similarity cutoff, default is
    0.0, range is 0.0-1.0 (Optional)
    "enableRerank?": boolean // Whether to enable reranking, default is True
    (Optional)
    "labels": [
        {
            "id": string (uuid)
            "name": string
        }
    ]
    "chatbots?": [ // Optional
        {
            "id": string (uuid)
            "name": string
        }
    ]
    "groups?": [ // List of groups that can access the knowledge base
    (Optional)
        {
            "id": string (uuid)
            "name": string
        }
    ]
    "filesCount": integer // Returns the number of ChatbotFiles under this
    knowledge base (excluding processed files)
    "createdAt": string (timestamp)
    "updatedAt": string (timestamp)
}
]
}

```

Example Response Value

```
{
    "count": 123,
    "next": "http://api.example.org/accounts/?page=4",
    "previous": "http://api.example.org/accounts/?page=2",
    "results": [
        {

```

```

"id": "550e8400-e29b-41d4-a716-446655440000",
"user": "550e8400-e29b-41d4-a716-446655440000",
"organization": "550e8400-e29b-41d4-a716-446655440000",
"embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
"rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
"name": "Response string",
"description": "Response string",
"numberOfRetrievedChunks": 456,
"sentenceWindowSize": 456,
"enableHyde": false,
"similarityCutoff": 456,
"enableRerank": false,
"labels": [
  {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
  }
],
"chatbots": [
  {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
  }
],
"groups": [
  {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
  }
],
"filesCount": 456,
"createdAt": "Response string",
"updatedAt": "Response string"
}
]
}

```

- ### Get Specific Knowledge Base Details

GET `/api/knowledge-bases/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Knowledge Base.

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

JavaScript

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully retrieved response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Example Response Structure

```
{
  "id": string (uuid)
  "user?: string (uuid) // Optional
  "organization?: string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?: string // Optional
  "numberOfRetrievedChunks?: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?: integer // RAG sentence window size, default is 2, minimum is 0 (Optional)
}
```

```

"enableHyde"??: boolean // Enable HyDE, default is False (Optional)
"similarityCutoff"??: number (double) // Similarity cutoff, default is 0.0,
range is 0.0-1.0 (Optional)
"enableRerank"??: boolean // Enable rerank, default is True (Optional)
"labels": [
  {
    "id": string (uuid)
    "name": string
  }
]
"chatbots"??: [ // Optional
  {
    "id": string (uuid)
    "name": string
  }
]
"groups"??: [ // List of groups that can access the knowledge base (Optional)
  {
    "id": string (uuid)
    "name": string
  }
]
"filesCount": integer // Returns the number of ChatbotFiles under this
knowledge base (excluding processed files)
"createdAt": string (timestamp)
"updatedAt": string (timestamp)
}

```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ]
}
```

```

    }
],
"chatbots": [
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string"
},
],
"groups": [
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string"
},
],
"filesCount": 456,
"createdAt": "Response string",
"updatedAt": "Response string"
}

```

- ### Update Knowledge Base

PUT /api/knowledge-bases/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Knowledge Base.

Request Body

Request Parameters

Field	Type	Required	Description
user	string (uuid)	No	
organization	string (uuid)	No	
embeddingModel	string (uuid)	Yes	
rerankerModel	string (uuid)	Yes	
name	string	Yes	
description	string	No	

Field	Type	Required	Description
numberOfRetrievedChunks	integer	No	Number of retrieved reference chunks, default is 12, minimum is 1
sentenceWindowSize	integer	No	RAG augmented sentence window size, default is 2, minimum is 0
enableHyde	boolean	No	Enable HyDE, default is False
similarityCutoff	number (double)	No	Similarity cutoff, default is 0.0, range is 0.0-1.0
enableRerank	boolean	No	Whether to enable reranking, default is True
chatbots	array[IdName]	No	
groups	array[IdName]	No	List of groups that can access the knowledge base

Request Structure Example

```
{
  "user"?: string (uuid) // Optional
  "organization"?: string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description"?: string // Optional
  "numberOfRetrievedChunks"?: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize"?: integer // RAG augmented sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde"?: boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff"?: number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank"?: boolean // Whether to enable rerank, default is True (Optional)
  "chatbots"?: [ // Optional
    {
      "id": string (uuid)
    }
  ]
  "groups"?: [ // List of groups that can access the knowledge base (Optional)
    {
      ...
    }
  ]
}
```

```
        "id": string (uuid)
    }
]
}
```

Request Sample Value

```
{
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example string",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
}
```

Code Example

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}'.

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Sample Name",
  "description": "Sample String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
  })
```

```
    console.error(error.response?.data || error.message);
});
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Sample Name",
    "description": "Sample String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```



```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "user": "550e8400-e29b-41d4-a716-446655440000",
            "organization": "550e8400-e29b-41d4-a716-446655440000",
            "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
            "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Example Name",
            "description": "Example String",
            "numberOfRetrievedChunks": 123,
            "sentenceWindowSize": 123,
            "enableHyde": true,
            "similarityCutoff": 123,
            "enableRerank": true,
            "chatbots": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "groups": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
```

```
}
```

```
?>
```

Response Content

Status Code: 200

Response Structure Example

```
{
  "id": string (uuid)
  "user?": string (uuid) // Optional
  "organization?": string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?": string // Optional
  "numberOfRetrievedChunks?": integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?": integer // RAG expansion sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde?": boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff?": number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank?": boolean // Enable rerank, default is True (Optional)
  "labels": [
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "chatbots?": [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "groups?": [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "filesCount": integer // Returns the number of ChatbotFiles under this
}
```

knowledge base (excluding processed files)

```
"createdAt": string (timestamp)  
"updatedAt": string (timestamp)  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "user": "550e8400-e29b-41d4-a716-446655440000",  
  "organization": "550e8400-e29b-41d4-a716-446655440000",  
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",  
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response string",  
  "description": "Response string",  
  "numberOfRetrievedChunks": 456,  
  "sentenceWindowSize": 456,  
  "enableHyde": false,  
  "similarityCutoff": 456,  
  "enableRerank": false,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "chatbots": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "groups": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "filesCount": 456,  
  "createdAt": "Response string",  
  "updatedAt": "Response string"  
}
```

- ### Partially Update a Knowledge Base

PATCH [/api/knowledge-bases/{id}/](#)

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Knowledge Base.

Request Body

Request Parameters

Field	Type	Required	Description
<code>user</code>	string (uuid)	No	
<code>organization</code>	string (uuid)	No	
<code>embeddingModel</code>	string (uuid)	No	
<code>rerankerModel</code>	string (uuid)	No	
<code>name</code>	string	No	
<code>description</code>	string	No	
<code>numberOfRetrievedChunks</code>	integer	No	Number of retrieved chunks, default is 12, minimum is 1
<code>sentenceWindowSize</code>	integer	No	RAG augmented sentence window size, default is 2, minimum is 0
<code>enableHyde</code>	boolean	No	Enable HyDE, default is False
<code>similarityCutoff</code>	number (double)	No	Similarity cutoff, default is 0.0, range is 0.0-1.0
<code>enableRerank</code>	boolean	No	Enable reranking, default is True
<code>chatbots</code>	array[IdName]	No	
<code>groups</code>	array[IdName]	No	List of groups that can access the knowledge base

Request Structure Example

```
{  
  "user": string (uuid) // Optional
```

```

"organization"??: string (uuid) // Optional
"embeddingModel"??: string (uuid) // Optional
"rerankerModel"??: string (uuid) // Optional
"name"??: string // Optional
"description"??: string // Optional
"numberOfRetrievedChunks"??: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
"sentenceWindowSize"??: integer // RAG sentence window size, default is 2, minimum is 0 (Optional)
"enableHyde"??: boolean // Enable HyDE, default is False (Optional)
"similarityCutoff"??: number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
"enableRerank"??: boolean // Enable rerank, default is True (Optional)
"chatbots"??: [ // Optional
{
  "id": string (uuid)
}
]
"groups"??: [ // List of groups that can access the knowledge base (Optional)
{
  "id": string (uuid)
}
]
}

```

Request Sample Value

```
{
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
}
```

```
        }
    ]
}
```

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Sample Name",
  "description": "Sample String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
  })
```

```
    console.error(error.response?.data || error.message);
});
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "範例名稱",
    "description": "範例字串",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```



```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "user": "550e8400-e29b-41d4-a716-446655440000",
            "organization": "550e8400-e29b-41d4-a716-446655440000",
            "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
            "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Example Name",
            "description": "Example String",
            "numberOfRetrievedChunks": 123,
            "sentenceWindowSize": 123,
            "enableHyde": true,
            "similarityCutoff": 123,
            "enableRerank": true,
            "chatbots": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "groups": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
```

```
}
```

Response Body

Status Code: 200

Example Response Structure

```
{
  "id": string (uuid)
  "user?": string (uuid) // Optional
  "organization?": string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?": string // Optional
  "numberOfRetrievedChunks?": integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?": integer // RAG augmented sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde?": boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff?": number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank?": boolean // Enable rerank, default is True (Optional)
  "labels": [
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "chatbots?": [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "groups?": [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "filesCount": integer // Returns the number of ChatbotFiles under this
}
```

```
knowledge base (excluding processed files)
```

```
  "createdAt": string (timestamp)
  "updatedAt": string (timestamp)
}
```

Response Example

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "filesCount": 456,
  "createdAt": "Response string",
  "updatedAt": "Response string"
}
```

- ### Delete Knowledge Base

DELETE /api/knowledge-bases/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Knowledge Base.

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code	Description
204	No response body

*

Create a new label

POST /api/knowledge-bases/{knowledgeBasePk}/labels/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
name	string	Yes	

Request Structure Example

```
{  
  "name": string  
}
```

Request Sample Value

```
{  
  "name": "Sample Name"  
}
```

Code Example

Shell/Bash

```
# API Call Example (Shell)  
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/labels/"  
  -H "Authorization: Api-Key YOUR_API_KEY"  
  -H "Content-Type: application/json"  
  -d '{  
    "name": "Example Name"  
  }'  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "name": "Example Name"
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "name": "Example Name"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "name": "Sample Name"
        ]
    ]);

    $data = json_decode($response→getBody(), true);
    echo "Successfully received response:\n";
    print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e→getMessage();
}
?>

```

Response Body

Status Code: 201

Example Response Structure

```
{
  "id": string (uuid)
  "name"??: string // Optional
}
```

Response Example

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response String"  
}
```

- ### List labels in the knowledge base**

GET `/api/knowledge-bases/{knowledgeBasePk}/labels/`

Parameters

Parameter Name	Required	Type	Description
<code>knowledgeBasePk</code>	V	string	
<code>page</code>	X	integer	A page number within the paginated result set.
<code>pageSize</code>	X	integer	Number of results to return per page.

Code Example

Shell/Bash

```
# API Call Example (Shell)  
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/labels/?page=1&pageSize=1"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/?page=1&pageSize=1", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/?page=1&pageSize=1"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/?page=1&pageSize=1", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Content

Status Code: 200

Response Structure Example

```
{
  "count": integer
  "next"??: string (uri) // optional
  "previous"??: string (uri) // optional
  "results": [
    {
      "id": string (uuid)
      "name"??: string // optional
    }
  ]
}
```

Response Example Value

```
{  
  "count": 123,  
  "next": "http://api.example.org/accounts/?page=4",  
  "previous": "http://api.example.org/accounts/?page=2",  
  "results": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response String"  
    }  
  ]  
}
```

- ### Get Specific Label Details**

GET `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)  
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Example Response Structure

```
{
  "id": string (uuid)
  "name": string // Optional
}
```

Response Example

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
```

```
        "name": "Response String"  
    }
```

- *Update Label*

PUT `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>name</code>	string	Yes	

Request Structure Example

```
{  
  "name": string  
}
```

Request Sample Value

```
{  
  "name": "Sample Name"  
}
```

Code Example

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "name": "Example Name"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "name": "Example Name"
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "name": "Example Name"
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "name": "Sample Name"
    ]
]);

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Structure Example

```
{
  "id": string (uuid)
  "name?: string // Optional
}
```

Example Response Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response String"  
}
```

- ### Partially Update Label**

PATCH `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>name</code>	string	No	

Request Structure Example

```
{  
  "name"??: string // Optional  
}
```

Request Sample Value

```
{  
  "name": "Example Name"  
}
```

Code Example

```
# Call API Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "name": "Example Name"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "name": "Example Name"
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/",
  data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "name": "Sample Name"
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "name": "Sample Name"
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Structure Example

```
{
  "id": string (uuid)
  "name?: string // Optional
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response String"  
}
```

- ### Delete Label

DELETE /api/knowledge-bases/{knowledgeBasePk}/labels/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this knowledge base label.
knowledgeBasePk	V	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)  
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/",
  data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code	Description
204	No response body

*

Upload File to Knowledge Base

POST /api/knowledge-bases/{knowledgeBasePk}/files/

Parameters

Parameter Name	Required	Type	Description
<code>knowledgeBasePk</code>	V	string	
<code>fileType</code>	X	string	File type filter, e.g., pdf, docx, txt, xlsx, etc.
<code>knowledgeBase</code>	X	string	Knowledge Base ID
<code>page</code>	X	integer	A page number within the paginated result set.
<code>pageSize</code>	X	integer	Number of results to return per page.
<code>query</code>	X	string	Search keyword, can search for file name or file ID.
<code>status</code>	X	string	File status filter. Possible values: <code>initial</code> , <code>processing</code> , <code>done</code> , <code>deleting</code> , <code>deleted</code> , <code>failed</code> .

Request Body

Request Parameters

Field	Type	Required	Description
<code>files</code>	array[<code>ChatbotFileCreate</code>]	Yes	Supports batch upload of multiple files, with a limit of 50 files per request

Request Structure Example

```
{
  "files": [ // Supports batch upload of multiple files, with a limit of 50
    files per request
    {
      "filename": string // File name
      "file": string (uri) // The file to be uploaded
      "knowledgeBase?": // Optional
      {
        "id": string (uuid)
      }
      "parser?": string (uuid) // Optional
      "labels?": [ // Optional
        {
          "id": string (uuid)
        }
      ]
      "rawUserDefineMetadata?": object // Optional
    }
  ]
}
```

```
]  
}
```

Request Sample Value

```
{  
  "files": [  
    {  
      "filename": "my-file.pdf",  
      "file": "string",  
      "knowledgeBase": {  
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  
        "name": "my-knowledge-base"  
      },  
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  
      "labels": [  
        {  
          "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  
          "name": "my-label"  
        }  
      ],  
      "rawUserDefineMetadata": "string"  
    },  
    {  
      "filename": "you-can-upload-multiple-files.pdf",  
      "file": "string",  
      "knowledgeBase": {  
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",  
        "name": "string"  
      },  
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"  
    }  
  ]  
}
```

Code Example

```
# Call API Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "files": [
    {
      "filename": "my-file.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "my-knowledge-base"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "labels": [
        {
          "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
          "name": "my-label"
        }
      ],
      "rawUserDefineMetadata": "string"
    },
    {
      "filename": "you-can-upload-multiple-files.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "string"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
    }
  ]
}'.

# Please replace YOUR_API_KEY and verify the request data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "files": [
    {
      "filename": "my-file.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "my-knowledge-base"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "labels": [
        {
          "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
          "name": "my-label"
        }
      ],
      "rawUserDefineMetadata": "string"
    },
    {
      "filename": "you-can-upload-multiple-files.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "string"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
    }
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-
```

```
e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", data,
config)
.then(response => {
  console.log('Successfully received response:');
  console.log(response.data);
})
.catch(error => {
  console.error('An error occurred with the request:');
  console.error(error.response ?.data || error.message);
});
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "files": [
        {
            "filename": "my-file.pdf",
            "file": "string",
            "knowledgeBase": {
                "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                "name": "my-knowledge-base"
            },
            "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
            "labels": [
                {
                    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                    "name": "my-label"
                }
            ],
            "rawUserDefineMetadata": "string"
        },
        {
            "filename": "you-can-upload-multiple-files.pdf",
            "file": "string",
            "knowledgeBase": {
                "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                "name": "string"
            },
            "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
        }
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
```

```
print("Successfully received response:")
print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/?
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "files": [
            {
                "filename": "my-file.pdf",
                "file": "string",
                "knowledgeBase": {
                    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                    "name": "my-knowledge-base"
                },
                "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                "labels": [
                    {
                        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                        "name": "my-label"
                    }
                ],
                "rawUserDefineMetadata": "string"
            },
            {
                "filename": "you-can-upload-multiple-files.pdf",
                "file": "string",
                "knowledgeBase": {
                    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                    "name": "string"
                },
                "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
            }
        ]
    }
}]
```

```
]);  
  
    $data = json_decode($response->getBody(), true);  
    echo "Successfully received response:\n";  
    print_r($data);  
} catch (Exception $e) {  
    echo 'Request failed: ' . $e->getMessage();  
}  
?>
```

Response Body

Status Code: 201

Sample Response Structure

```
{  
    "count": integer  
    "next"??: string (uri) // Optional  
    "previous"??: string (uri) // Optional  
    "results": [  
        {  
            "id": string (uuid)  
            "filename": string // File name  
            "file": string (uri) // File to upload  
            "fileType": string  
            "knowledgeBase"??: // Optional  
            {  
                "id": string (uuid)  
                "name": string  
            }  
            "size": integer  
            "status":  
            {  
            }  
            "parser": {  
                "id": string (uuid)  
                "name": string  
                "provider":  
                {  
                }  
                "order"??: integer // Optional  
            }  
        }  
    ]  
}
```

```
        }
      "labels"? : [ // Optional
        {
          "id": string (uuid)
          "name": string
        }
      ]
    "rawUserDefineMetadata"? : object // Optional
    "createdAt": string (timestamp)
  }
]
}
```

Response Example

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "filename": "Response String",
      "file": "https://example.com/file.jpg",
      "fileType": "Response String",
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String"
      },
      "size": 456,
      "status": {},
      "parser": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String",
        "provider": {},
        "order": 456
      },
      "labels": [
        {
          "id": "550e8400-e29b-41d4-a716-446655440000",
          "name": "Response String"
        }
      ],
      "rawUserDefineMetadata": null,
      "createdAt": "Response String"
    }
  ]
}
```

```
]  
}
```

- ### Batch Delete Files**

POST /api/knowledge-bases/{knowledgeBasePk}/files/batch-delete/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
ids	array[string]	Yes	

Request Structure Example

```
{  
  "ids": [  
    string (uuid)  
  ]  
}
```

Request Sample Value

```
{  
  "ids": [  
    "550e8400-e29b-41d4-a716-446655440000"  
  ]  
}
```

Code Example

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/batch-delete/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/batch-delete/" , data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/batch-delete/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "ids": [
        "550e8400-e29b-41d4-a716-446655440000"
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/batch-delete/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "ids": [
                "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response→getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Error during request: ' . $e→getMessage();
}
?>

```

Response Content

Status Code	Description
204	No response body

List Files in a Knowledge Base

GET </api/knowledge-bases/{knowledgeBasePk}/files/>

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
fileType	X	string	File type filter, e.g., pdf, docx, txt, xlsx, etc.
knowledgeBase	X	string	Knowledge Base ID
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	Search keyword, can search by file name or file ID.
status	X	string	File status filter. Possible values: <code>initial</code> , <code>processing</code> , <code>done</code> , <code>deleting</code> , <code>deleted</code> , <code>failed</code> . <code>initial</code> :Initial; <code>processing</code> : Processing; <code>done</code> :Done; <code>deleting</code> :Deleting ; <code>deleted</code> :Deleted ; `faile...

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-
e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted",
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/?
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Response received successfully:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Example Response Structure

```
{
  "count": integer
  "next?: string (uri) // Optional
  "previous?: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "filename": string // File name
      "file": string (uri) // File to upload
      "fileType": string
    }
  ]
}
```

```
"knowledgeBase?": // Optional
{
  "id": string (uuid)
  "name": string
}
"size": integer
"status":
{
}
"parser": {
{
  "id": string (uuid)
  "name": string
  "provider":
  {
  }
  "order?": integer // Optional
}
}
"labels?": [ // Optional
{
  "id": string (uuid)
  "name": string
}
]
"rawUserDefineMetadata?": object // Optional
"createdAt": string (timestamp)
}
]
}
```

Example Response Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "filename": "Response String",
      "file": "https://example.com/file.jpg",
      "fileType": "Response String",
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String"
      },
    }
  ]
}
```

```

    "size": 456,
    "status": {},
    "parser": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String",
        "provider": {},
        "order": 456
    },
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Response String"
        }
    ],
    "rawUserDefineMetadata": null,
    "createdAt": "Response String"
}
]
}

```

- ### Get Specific File Details**

GET </api/knowledge-bases/{knowledgeBasePk}/files/{id}/>

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Chatbot File.
knowledgeBasePk	V	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

JavaScript

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully retrieved response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Example Response Structure

```
{
  "id": string (uuid)
  "filename": string // File name
  "file": string (uri) // File to upload
  "fileType": string
  "knowledgeBase"??: // Optional
  {
    "id": string (uuid)
    "name": string
  }
  "size": integer
}
```

```
"status":  
{  
}  
}  
"parser{  
    "id": string (uuid)  
    "name": string  
    "provider":  
    {  
    }  
    "order"?: integer // Optional  
}  
}  
}  
"labels??: [ // Optional  
{  
    "id": string (uuid)  
    "name": string  
}  
]  
"rawUserDefineMetadata??: object // Optional  
"createdAt: string (timestamp)  
}
```

Response Example Value

```
{  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "filename": "Response String",  
    "file": "https://example.com/file.jpg",  
    "fileType": "Response String",  
    "knowledgeBase": {  
        "id": "550e8400-e29b-41d4-a716-446655440000",  
        "name": "Response String"  
    },  
    "size": 456,  
    "status": {},  
    "parser        "id": "550e8400-e29b-41d4-a716-446655440000",  
        "name": "Response String",  
        "provider": {},  
        "order": 456  
    },  
    "labels        {  
            "id": "550e8400-e29b-41d4-a716-446655440000",  
            "name": "Response String"  
        }  
    ]  
}
```

```

],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}

```

- ### Update File Information

PUT /api/knowledge-bases/{knowledgeBasePk}/files/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Chatbot File.
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
filename	string	Yes	File name
file	string (uri)	Yes	The file to upload
knowledgeBase	object (contains 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	
parser	string (uuid)	No	
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	

Request Structure Example

```

{
  "filename": string // File name
  "file": string (uri) // The file to upload
  "knowledgeBase?": // Optional
  {
    "id": string (uuid)
  }
}

```

```
}

"parser"??: string (uuid) // Optional
"labels"??: [ // Optional
{
  "id": string (uuid)
}
]
"rawUserDefineMetadata"??: object // Optional
}
```

Request Sample Value

```
{
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
}
```

Code Example

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "filename": "document.pdf",
            "file": "https://example.com/file.jpg",
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ],
            "parser": "550e8400-e29b-41d4-a716-446655440000",
            "labels": [
                [
                    {
                        "id": "550e8400-e29b-41d4-a716-446655440000"
                    }
                ],
                "rawUserDefineMetadata": null
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Response received successfully:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Status Code: 200

Response Structure Example

```
{  
  "id": string (uuid)  
  "filename": string // File name  
  "file": string (uri) // File to upload  
  "fileType": string  
  "knowledgeBase"??: // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
  "size": integer  
  "status":  
  {  
  }  
  "parser": {  
    "id": string (uuid)  
    "name": string  
    "provider":  
    {  
    }  
    "order"??: integer // Optional  
  }  
  }  
  "labels"??: [ // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
]  
  "rawUserDefineMetadata"??: object // Optional  
  "createdAt": string (timestamp)  
}
```

Example Response Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "filename": "Response String",  
  "file": "https://example.com/file.jpg",  
  "fileType": "Response String",  
  "knowledgeBase": {
```

```

    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
},
"size": 456,
"status": {},
"parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String",
    "provider": {},
    "order": 456
},
"labels": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
}
],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}

```

- ### Partially Update a File

PATCH [/api/knowledge-bases/{knowledgeBasePk}/files/{id}/](#)

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Chatbot File.
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
filename	string	No	File name
file	string (uri)	No	The file to be uploaded
knowledgeBase	object (contains 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	

Field	Type	Required	Description
parser	string (uuid)	No	
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	

Request Structure Example

```
{
  "filename"??: string // File name (Optional)
  "file"??: string (uri) // File to upload (Optional)
  "knowledgeBase"??: // Optional
  {
    "id": string (uuid)
  }
  "parser"??: string (uuid) // Optional
  "labels"??: [ // Optional
  {
    "id": string (uuid)
  }
]
  "rawUserDefineMetadata"??: object // Optional
}
```

Request Example Value

```
{
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
}
```

Code Example

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": None
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "filename": "document.pdf",
            "file": "https://example.com/file.jpg",
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ],
            "parser": "550e8400-e29b-41d4-a716-446655440000",
            "labels": [
                [
                    {
                        "id": "550e8400-e29b-41d4-a716-446655440000"
                    }
                ],
                "rawUserDefineMetadata": null
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Status Code: 200

Response Structure Example

```
{  
  "id": string (uuid)  
  "filename": string // File name  
  "file": string (uri) // File to upload  
  "fileType": string  
  "knowledgeBase"??: // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
  "size": integer  
  "status":  
  {  
  }  
  "parser": {  
    "id": string (uuid)  
    "name": string  
    "provider":  
    {  
    }  
    "order"??: integer // Optional  
  }  
  }  
  "labels"??: [ // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
]  
  "rawUserDefineMetadata"??: object // Optional  
  "createdAt": string (timestamp)  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "filename": "Response String",  
  "file": "https://example.com/file.jpg",  
  "fileType": "Response String",  
  "knowledgeBase": {
```

```

    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
},
"size": 456,
"status": {},
"parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String",
    "provider": {},
    "order": 456
},
"labels": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
}
],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}

```

- ### Batch Reparse Files

PATCH [/api/knowledge-bases/{knowledgeBasePk}/files/batch-reparse/](#)

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
fileType	X	string	File type filter, e.g., pdf, docx, txt, xlsx, etc.
knowledgeBase	X	string	Knowledge Base ID
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	Search keyword, can search by file name or file ID.
status	X	string	File status filter. Possible values: initial , processing , done , deleting , deleted , failed . initial :Initial; processing : Processing; done :Done; deleting :Deleting ; deleted :Deleted ; `faile...

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "filename": "document.pdf",
            "file": "https://example.com/file.jpg",
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ],
            "parser": "550e8400-e29b-41d4-a716-446655440000",
            "labels": [
                [
                    {
                        "id": "550e8400-e29b-41d4-a716-446655440000"
                    }
                ],
                "rawUserDefineMetadata": null
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Status Code: 200

Response Structure Example

```
{  
  "count": integer  
  "next"??: string (uri) // Optional  
  "previous"??: string (uri) // Optional  
  "results": [  
    {  
      "id": string (uuid)  
      "filename": string // File name  
      "file": string (uri) // File to upload  
      "fileType": string  
      "knowledgeBase"??: // Optional  
      {  
        "id": string (uuid)  
        "name": string  
      }  
      "size": integer  
      "status":  
      {  
      }  
      "parser": {  
        {  
          "id": string (uuid)  
          "name": string  
          "provider":  
          {  
          }  
          "order"??: integer // Optional  
        }  
      }  
      "labels"??: [ // Optional  
        {  
          "id": string (uuid)  
          "name": string  
        }  
      ]  
      "rawUserDefineMetadata"??: object // Optional  
      "createdAt": string (timestamp)  
    }  
  ]  
}
```

Response Example Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "filename": "Response String",
      "file": "https://example.com/file.jpg",
      "fileType": "Response String",
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String"
      },
      "size": 456,
      "status": {},
      "parser": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String",
        "provider": {},
        "order": 456
      },
      "labels": [
        {
          "id": "550e8400-e29b-41d4-a716-446655440000",
          "name": "Response String"
        }
      ],
      "rawUserDefineMetadata": null,
      "createdAt": "Response String"
    }
  ]
}
```

- ### Delete File

DELETE `/api/knowledge-bases/{knowledgeBasePk}/files/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Chatbot File.
<code>knowledgeBasePk</code>	V	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response Content

Status Code	Description
204	No response body

*

Create a new FAQ

POST /api/knowledge-bases/{knowledgeBasePk}/faqs/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
question	string	Yes	
answer	string	Yes	
answerMediaUrls	object	No	URLs for media files such as images and videos in the answer.
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	
knowledgeBase	object (with 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	

Request Structure Example

```
{
  "question": string
  "answer": string
  "answerMediaUrls"??: object // Stores the URLs of media files such as images, videos, etc., in the answer (optional)
  "labels"??: [ // (optional)
    {
      "id": string (uuid)
    }
  ]
  "rawUserDefineMetadata"??: object // (optional)
  "knowledgeBase"??: // (optional)
  {
    "id": string (uuid)
  }
}
```

Request Sample Value

```
{  
    "question": "Sample string",  
    "answer": "Sample string",  
    "answerMediaUrls": null,  
    "labels": [  
        {  
            "id": "550e8400-e29b-41d4-a716-446655440000"  
        }  
    ],  
    "rawUserDefineMetadata": null,  
    "knowledgeBase": {  
        "id": "550e8400-e29b-41d4-a716-446655440000"  
    }  
}
```

Code Example

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "question": "Sample string",
  "answer": "Sample string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "question": "Sample string",
            "answer": "Sample string",
            "answerMediaUrls": null,
            "labels": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "rawUserDefineMetadata": null,
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 201

Example Response Structure

```
{  
  "id": string (uuid)  
  "question": string  
  "answer": string  
  "answerMediaUrls"??: object // Stores the URLs of media files such as images,  
videos, etc. in the answer (optional)  
  "hitsCount": integer  
  "embeddingTokensCount": integer // The number of embedding tokens used when  
creating this FAQ  
  "labels"??: [ // optional  
    {  
      "id": string (uuid)  
      "name": string  
    }  
  ]  
  "rawUserDefineMetadata"??: object // optional  
  "knowledgeBase"??: // optional  
  {  
    "id": "string (uuid)",  
    "name": "string"  
  }  
}
```

Response Example

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "question": "Response string",  
  "answer": "Response string",  
  "answerMediaUrls": null,  
  "hitsCount": 456,  
  "embeddingTokensCount": 456,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Response string"
```

```
    }  
}
```

- ### Batch Delete FAQs**

POST `/api/knowledge-bases/{knowledgeBasePk}/faqs/batch-delete/`

Parameters

Parameter Name	Required	Type	Description
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>ids</code>	array[string]	No	

Request Structure Example

```
{  
  "ids"??: [ // Optional  
    string (uuid)  
  ]  
}
```

Request Sample Value

```
{  
  "ids": [  
    "550e8400-e29b-41d4-a716-446655440000"  
  ]  
}
```

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/batch-delete/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/batch-delete/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/batch-delete/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "ids": [
        "550e8400-e29b-41d4-a716-446655440000"
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/batch-delete/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "ids": [
                "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response→getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e→getMessage();
}
?>

```

Response Content

Status Code	Description
204	No response body

List all FAQs for a specific knowledge base

GET </api/knowledge-bases/{knowledgeBasePk}/faqs/>

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/?page=1&pageSize=1&query=example"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/?page=1&pageSize=1&query=example", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/?page=1&pageSize=1&query=example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/?
page=1&pageSize=1&query=example", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Structure Example

```
{
  "count": integer
  "next"?: string (uri) // Optional
  "previous"?: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "question": string
      "answer": string
      "answerMediaUrls"?: object // Stores URLs for media files like images,
                                 videos, etc., in the answer (Optional)
    }
  ]
}
```

```
"hitsCount": integer
"embeddingTokensCount": integer // The number of embedding tokens used
when creating this FAQ
"labels"??: [ // Optional
{
  "id": string (uuid)
  "name": string
}
]
"rawUserDefineMetadata"??: object // Optional
"knowledgeBase"??: // Optional
{
  "id": string (uuid)
  "name": string
}
}
]
}
```

Response Example

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "question": "Response string",
      "answer": "Response string",
      "answerMediaUrls": null,
      "hitsCount": 456,
      "embeddingTokensCount": 456,
      "labels": [
        {
          "id": "550e8400-e29b-41d4-a716-446655440000",
          "name": "Response string"
        }
      ],
      "rawUserDefineMetadata": null,
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string"
      }
    }
  ]
}
```

```
]  
}
```

- ### Get Specific FAQ Details**

GET </api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/>

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this FAQ.
<code>knowledgeBasePk</code>	V	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)  
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully retrieved response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Structure Example

```
{
  "id": string (uuid)
  "question": string
  "answer": string
  "answerMediaUrls"??: object // Stores the URLs of media files such as images,
videos, etc. in the answer (optional)
  "hitsCount": integer
  "embeddingTokensCount": integer // The number of embedding tokens used when
creating this FAQ
  "labels"??: [ // optional
  {
    "label": string
  }
]
```

```

        "id": string (uuid)
        "name": string
    }
]
"rawUserDefineMetadata"??: object // optional
"knowledgeBase"??: // optional
{
    "id": string (uuid)
    "name": string
}
}

```

Example Response Value

```
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "question": "Response string",
    "answer": "Response string",
    "answerMediaUrls": null,
    "hitsCount": 456,
    "embeddingTokensCount": 456,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Response string"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string"
    }
}
```

- ### Update FAQ

PUT /api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this FAQ.
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
question	string	Yes	
answer	string	Yes	
answerMediaUrls	object	No	URLs for media files such as images and videos in the answer
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	
knowledgeBase	object (with 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	

Request Structure Example

```
{
  "question": string
  "answer": string
  "answerMediaUrls"??: object // Stores the URLs of media files such as images, videos, etc. in the answer (optional)
  "labels"??: [ // (optional)
    {
      "id": string (uuid)
    }
  ]
  "rawUserDefineMetadata"??: object // (optional)
  "knowledgeBase"??: // (optional)
  {
    "id": string (uuid)
  }
}
```

Request Sample Value

```
{  
  "question": "Sample string",  
  "answer": "Sample string",  
  "answerMediaUrls": null,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000"  
  }  
}
```

Code Example

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "question": "Sample string",
  "answer": "Sample string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "question": "Sample string",
            "answer": "Sample string",
            "answerMediaUrls": null,
            "labels": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "rawUserDefineMetadata": null,
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Status Code: 200

Example Response Structure

```
{  
  "id": string (uuid)  
  "question": string  
  "answer": string  
  "answerMediaUrls"??: object // Stores the URLs of media files such as images,  
  videos, etc. in the answer (optional)  
  "hitsCount": integer  
  "embeddingTokensCount": integer // The number of embedding tokens used when  
  creating this FAQ  
  "labels"??: [ // optional  
    {  
      "id": string (uuid)  
      "name": string  
    }  
  ]  
  "rawUserDefineMetadata"??: object // optional  
  "knowledgeBase"??: // optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
}
```

Response Example Value

```
{  
  "id": "50e8400-e29b-41d4-a716-446655440000",  
  "question": "Response string",  
  "answer": "Response string",  
  "answerMediaUrls": null,  
  "hitsCount": 456,  
  "embeddingTokensCount": 456,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Response string"  
  }  
}
```

```
    }
}
```

- ### Partially Update FAQ

PATCH `/api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this FAQ.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>question</code>	string	No	
<code>answer</code>	string	No	
<code>answerMediaUrls</code>	object	No	URLs for media files such as images and videos in the answer
<code>labels</code>	array[IdName]	No	
<code>rawUserDefineMetadata</code>	object	No	
<code>knowledgeBase</code>	object (with 2 properties: id, name)	No	
<code>knowledgeBase.id</code>	string (uuid)	Yes	

Request Structure Example

```
{
  "question"??: string // Optional
  "answer"??: string // Optional
  "answerMediaUrls"??: object // Stores URLs of media files like images, videos, etc., in the answer (Optional)
  "labels"??: [ // Optional
```

```
{  
    "id": string (uuid)  
}  
]  
"rawUserDefineMetadata"??: object // Optional  
"knowledgeBase"??: // Optional  
{  
    "id": string (uuid)  
}  
}
```

Request Sample Value

```
{  
    "question": "example string",  
    "answer": "example string",  
    "answerMediaUrls": null,  
    "labels": [  
        {  
            "id": "550e8400-e29b-41d4-a716-446655440000"  
        }  
    ],  
    "rawUserDefineMetadata": null,  
    "knowledgeBase": {  
        "id": "550e8400-e29b-41d4-a716-446655440000"  
    }  
}
```

Code Example

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "question": "Example String",
    "answer": "Example String",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "question": "Sample string",
  "answer": "Sample string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "question": "Sample string",
            "answer": "Sample string",
            "answerMediaUrls": null,
            "labels": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "rawUserDefineMetadata": null,
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Status Code: 200

Example Response Structure

```
{  
  "id": string (uuid)  
  "question": string  
  "answer": string  
  "answerMediaUrls"??: object // Stores the URLs of media files such as images,  
  videos, etc. in the answer (optional)  
  "hitsCount": integer  
  "embeddingTokensCount": integer // The number of embedding tokens used when  
  creating this FAQ  
  "labels"??: [ // optional  
    {  
      "id": string (uuid)  
      "name": string  
    }  
  ]  
  "rawUserDefineMetadata"??: object // optional  
  "knowledgeBase"??: // optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "question": "Response string",  
  "answer": "Response string",  
  "answerMediaUrls": null,  
  "hitsCount": 456,  
  "embeddingTokensCount": 456,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Response string"  
  }  
}
```

```
    }  
}
```

- ### Delete FAQ

DELETE /api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this FAQ.
knowledgeBasePk	V	string	

Code Example

Shell/Bash

```
# API Call Example (Shell)  
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code	Description
204	No response body

*

對話-widget

I have successfully translated the complete Traditional Chinese Markdown file to English. The translation includes:

Key translations made:

- "取得特定 Web Chat" → "Get Specific Web Chat"

- "獲取 LLM 使用統計" → "Get LLM Usage Statistics"
- "取得 AI 助理統計數據" → "Get AI Assistant Statistics"
- "啟用或停用特定 AI 助理的評估功能" → "Enable or Disable Evaluation for Specific AI Assistant"
- "參數" → "Parameters"
- "程式碼範例" → "Code Examples"
- "回應內容" → "Response"
- "請求內容" → "Request Body"
- "呼叫 API 示例" → "API call example"
- "設定請求標頭" → "Configure request headers"
- "請求發生錯誤" → "Request error occurred"
- "成功取得回應" → "Successfully received response"
- "非必填" → "Optional"
- All comments in code examples were translated to English

The entire HTML structure, CSS styling, GitBook tabs syntax, and all code examples have been preserved exactly as in the original file. The file has been saved to [/home/runner/work/skills-for-Agents/skills-for-Agents/source_repo/en/api-reference/conversation-widget.md](#).

API call example (Shell)

I'll translate this Traditional Chinese Markdown document to English completely, preserving all the structure and content.

Send Message (Streaming)

POST

[/api/chatbots/{chatbotId}/completions/](#)

Parameters

Parameter Name	Required	Type	Description
chatbotId	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
conversation	string (uuid)	No	Unique identifier for the conversation, will create a new conversation if empty (optional)
message	object (with 6 properties: content, contentPayload, queryMetadata...)	Yes	Message content to send
message.content	string	Yes	Text content of the message
message.contentPayload	object	No	Additional content payload of the message, JSON format (optional)
message.queryMetadata	object	No	Query metadata, JSON format (optional)
message.metadata	object	No	Environment metadata of the message, such as timezone information, JSON format (optional)

Field	Type	Required	Description
message.attachments	array[AttachmentInput]	No	List of message attachments (optional)
message.sender	string (uuid)	No	Sender's Contact ID (optional)
isStreaming	boolean	No	Whether to use streaming mode for response, defaults to false (optional)

Request Schema Example

```
{
  "conversation"??: string (uuid) // Unique identifier for the conversation, will create a new conversation if empty (optional) (non-required)
  "message": // Message content to send
  {
    "content": string // Text content of the message
    "contentPayload"??: object // Additional content payload of the message, JSON format (optional) (non-required)
    "queryMetadata"??: object // Query metadata, JSON format (optional) (non-required)
    "metadata"??: object // Environment metadata of the message, such as timezone information, JSON format (optional) (non-required)
    "attachments"??: [ // List of message attachments (optional) (non-required)
      {
        "id": string (uuid) // Unique identifier for the attachment
        "type": // Attachment type, possible values: image (Image), video (Video, under development, not yet supported), audio (Audio), sticker (Sticker, under development, not yet supported), other (Other)

        * `image` - Image
        * `video` - Video
        * `audio` - Audio
        * `sticker` - Sticker
        * `other` - Other
      }
    ]
  }
}
```

```
    }
    "filename": string // Filename of the attachment
    "file": string (uri) // URL address of the attachment file
  }
]
"sender"??: string (uuid) // Sender's Contact ID (optional) (non-required)
}
"isStreaming"??: boolean // Whether to use streaming mode for response, defaults to false (optional) (non-required)
}
```

Request Example Value

```
{
  "message": {
    "content": "Hello, please introduce yourself"
  },
  "is_streaming": true
}
```

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X POST "https://api.maiagent.ai/api/chatbots/550e8400-e29b-
41d4-a716-446655440000/completions/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "message": {
        "content": "Hello, please introduce yourself"
    },
    "is_streaming": true
}'

# Please make sure to replace YOUR_API_KEY and verify the request
# data before executing.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "message": {
    "content": "Hello, please introduce yourself"
  },
  "is_streaming": true
};

axios.post("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-446655440000/completions/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/completions/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "message": {
        "content": "Hello, please introduce yourself"
    },
    "is_streaming": true
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>post("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/completions/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "message": [
            "content": "Hello, please introduce yourself"
        ],
        "is_streaming": true
    ]
]);

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code	Description
-------------	-------------

200	Conversation response content, event stream if streaming
400	Request parameter error, possible reasons include: Contact ID does not belong to the organization

Send Message (Create)

POST [/api/messages/](#)

Request Body

Request Parameters

Field	Type	Required	Description
conversation	string (uuid)	Yes	
type	string	No	
content	string	No	
contentPayload	object	No	
attachments	array[AttachmentCreateInput]	No	
canvas	object	No	
canvas.name	string	Yes	
canvas.canvasType	object	Yes	
canvas.title	string	Yes	
canvas.content	string	Yes	
queryMetadata	object	No	

Request Schema Example

```
{  
  "conversation": string (uuid)  
  "type"??: string // non-required  
  "content"??: string // non-required  
  "contentPayload"??: object // non-required  
  "attachments"??: [ // non-required  
    {  
      "type"??: // non-required  
      {  
      }  
      "filename": string  
      "file": string (uri)  
      "conversation"??: string (uuid) // non-required  
    }  
  ]  
  "canvas"??: { // non-required  
  {  
    "name": string  
    "canvasType":  
    {  
    }  
    "title": string  
    "content": string  
  }  
  }  
  "queryMetadata"??: object // non-required  
}
```

Request Example Value

```
{  
  "conversation": "c96a0fb9-d106-4b16-8706-3906533bafa2",  
  "sender": {  
    "id": 2.992461161002561e+38,  
    "name": "jessiekuo",  
    "avatar": "https://autox-media-dev.playma.app/static/images/default-  
user-avatar.png"  
  },  
  "type": "incoming",  
  "content": "This is a test message",  
  "contentPayload": {  
    "content": "This is a test message",  
    "items": [  
      {  
        "type": "text",  
        "text": "This is a test message",  
      }  
    ]  
  }  
}
```

```
        "startTimestamp": null,
        "citations": []
    },
],
"metadata": null
},
"feedback": "like",
"createdAt": "1748314926000",
"attachments": [],
"citations": [],
"citationNodes": [],
"canvas": {
    "id": "034eecd7-e0f4-46e9-88cb-4fefdb5b7ddd",
    "name": "system-architecture-diagram",
    "canvasType": "markdown",
    "title": "System Architecture Diagram",
    "content": "←!— MERMAID_PLACEHOLDER_START —>
<div class="mermaid-diagram">
    <div class="mermaid">
\ngraph TD\n    A[User] —> B[Frontend Interface]\n    B —> C[API Layer]\n    C —> D[Database]\n
    </div>
    <p class="mermaid-caption">Mermaid 圖表</p>
</div>
←!— MERMAID_PLACEHOLDER_END —>",
    "createdAt": "1748314926000"
}
}
```

Code Examples

```
# API call example (Shell)
curl -X POST "https://api.maiagent.ai/api/messages/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "conversation": "c96a0fb9-d106-4b16-8706-3906533bafa2",
    "sender": {
        "id": 2.992461161002561e+38,
        "name": "jessiekuo",
        "avatar": "https://autox-media-
dev.playma.app/static/images/default-user-avatar.png"
    },
    "type": "incoming",
    "content": "This is a test message",
    "contentPayload": {
        "content": "This is a test message",
        "items": [
            {
                "type": "text",
                "text": "This is a test message",
                "startTimestamp": null,
                "citations": []
            }
        ],
        "metadata": null
    },
    "feedback": "like",
    "createdAt": "1748314926000",
    "attachments": [],
    "citations": [],
    "citationNodes": [],
    "canvas": {
        "id": "034eeecd7-e0f4-46e9-88cb-4fefdb5b7ddd",
        "name": "system-architecture-diagram",
        "canvasType": "markdown",
        "title": "System Architecture Diagram",
        "content": "
\ngraph TD\n    A[User] --> B[Frontend Interface]\n    B --> C[API Layer]\n    C --> D[Database]\n

```

```
",  
    "createdAt": "1748314926000"  
}  
}'  
  
# Please make sure to replace YOUR_API_KEY and verify the request  
data before executing.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "conversation": "c96a0fb9-d106-4b16-8706-3906533bafa2",
  "sender": {
    "id": 2.992461161002561e+38,
    "name": "jessiekuo",
    "avatar": "https://autox-media-
dev.playma.app/static/images/default-user-avatar.png"
  },
  "type": "incoming",
  "content": "This is a test message",
  "contentPayload": {
    "content": "This is a test message",
    "items": [
      {
        "type": "text",
        "text": "This is a test message",
        "startTimestamp": null,
        "citations": []
      }
    ],
    "metadata": null
  },
  "feedback": "like",
  "createdAt": "1748314926000",
  "attachments": [],
  "citations": [],
  "citationNodes": [],
  "canvas": {
    "id": "034eeecd7-e0f4-46e9-88cb-4fefdb5b7ddd",
    "name": "system-architecture-diagram",
    "canvasType": "markdown",
    "title": "System Architecture Diagram",
  }
};
```

```
"content": "  
  
\ngraph TD\n    A[User] --> B[Frontend Interface]\n    B --> C[API Layer]\n    C --> D[Database]\n  
  
Mermaid 圖表  
  
",  
    "createdAt": "1748314926000"  
}  
};  
  
axios.post("https://api.maiagent.ai/api/messages/", data, config)  
.then(response => {  
    console.log('Successfully received response:');  
    console.log(response.data);  
})  
.catch(error => {  
    console.error('Request error occurred:');  
    console.error(error.response?.data || error.message);  
});
```

```
import requests

url = "https://api.maiagent.ai/api/messages/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "conversation": "c96a0fb9-d106-4b16-8706-3906533bafa2",
    "sender": {
        "id": 2.992461161002561e+38,
        "name": "jessiekuo",
        "avatar": "https://autox-media-
dev.playma.app/static/images/default-user-avatar.png"
    },
    "type": "incoming",
    "content": "This is a test message",
    "contentPayload": {
        "content": "This is a test message",
        "items": [
            {
                "type": "text",
                "text": "This is a test message",
                "startTimestamp": null,
                "citations": []
            }
        ],
        "metadata": null
    },
    "feedback": "like",
    "createdAt": "1748314926000",
    "attachments": [],
    "citations": [],
    "citationNodes": [],
    "canvas": {
        "id": "034eeecd7-e0f4-46e9-88cb-4fefdb5b7ddd",
        "name": "system-architecture-diagram",
        "canvasType": "markdown",
        "title": "System Architecture Diagram",
        "content": "

```

```
\ngraph TD\n    A[User] --> B[Frontend Interface]\n    B --> C[API Layer]\n    C --> D[Database]\n
```

Mermaid 圖表

```
",\n        "createdAt": "1748314926000"\n    }\n}\n\nresponse = requests.post(url, json=data, headers=headers)\ntry:\n    print("Successfully received response:")\n    print(response.json())\nexcept Exception as e:\n    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>post("https://api.maiagent.ai/api/messages/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "conversation": "c96a0fb9-d106-4b16-8706-3906533bafa2",
        "sender": {
            "id": 2.992461161002561e+38,
            "name": "jessiekuo",
            "avatar": "https://autox-media-
dev.playma.app/static/images/default-user-avatar.png"
        },
        "type": "incoming",
        "content": "This is a test message",
        "contentPayload": {
            "content": "This is a test message",
            "items": [
                {
                    "type": "text",
                    "text": "This is a test message",
                    "startTimestamp": null,
                    "citations": []
                }
            ],
            "metadata": null
        },
        "feedback": "like",
        "createdAt": "1748314926000",
        "attachments": [],
        "citations": [],
        "citationNodes": [],
        "canvas": {
            "id": "034eeecd7-e0f4-46e9-88cb-4fefdb5b7ddd",
            "name": "system-architecture-diagram",
            "canvasType": "markdown".
        }
    ]
]
```

```
        "title": "System Architecture Diagram",
        "content": "

\ngraph TD\n    A[User] —> B[Frontend Interface]\n    B —> C[API Layer]\n    C —> D[Database]\n

Mermaid 圖表

",
        "createdAt": "1748314926000"
    }
}
]);

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 201

Response Schema Example

```
{
    "id": string (uuid)
    "conversation": string (uuid)
    "sender":
    {
        "id": string (uuid)
        "name": string
        "avatar": string
        "email"??: string (email) // non-required
        "phoneNumber"??: string // non-required
```

```
    }
    "type"??: string // non-required
    "content"??: string // non-required
    "contentPayload"??: object // non-required
    "feedback":
    {
        "id": string (uuid)
        "type":
        {
        }
        "suggestion"??: string // non-required
        "updatedAt": string (timestamp)
    }
    "createdAt": string (timestamp)
    "attachments"??: [ // non-required
    {
        "id": string (uuid)
        "type"??: // non-required
        {
        }
        "filename": string
        "file": string (uri)
        "conversation"??: string (uuid) // non-required
    }
]
"citations": [
{
    "id": string (uuid)
    "filename": string // Filename
    "file": string (uri) // File to upload
    "fileType": string
    "knowledgeBase"??: // non-required
    {
        "id": string (uuid)
        "name": string
    }
    "size": integer
    "status":
    {
    }
    "parser": {
    {
        "id": string (uuid)
        "name": string
        "provider":
        {
        }
        "order"??: integer // non-required
    }
}
```

```
        }
    }
    "labels"??: [ // non-required
    {
        "id": string (uuid)
        "name": string
    }
]
"rawUserDefineMetadata"??: object // non-required
"vectorStorageSize": integer // Size of vectors for this file in Elasticsearch (bytes)
"chunksCount": integer // Number of chunks/nodes generated from this file
"createdAt": string (timestamp)
}
]
"citationNodes": [
{
    "chatbotTextNode": {
    {
        "id": string (uuid)
        "charactersCount": integer
        "hitsCount": integer
        "text": string
        "updatedAt": string (timestamp)
        "filename": string
        "chatbotFile": string (uuid)
        "knowledgeBaseFile": string (uuid) // Same as get_chatbot_file, provides backward compatibility
        "pageNumber": integer
    }
    }
}
{
    "score"??: number (double) // non-required
    "displayScore": integer
}
]
"canvas"??: { // non-required
{
    "id": string (uuid)
    "name": string
    "canvasType":
    {
    }
    "title": string
    "content": string
    "createdAt": string (timestamp)
}
}
```

```
"queryMetadata"??: object // non-required
  "recordId": string // Returns the ChatbotRecord ID corresponding to the
Message
}
```

Response Example Value

```
{
  "conversation": "c96a0fb9-d106-4b16-8706-3906533bafa2",
  "sender": {
    "id": 2.992461161002561e+38,
    "name": "jessiekuo",
    "avatar": "https://autox-media-dev.playma.app/static/images/default-
user-avatar.png"
  },
  "type": "incoming",
  "content": "This is a test message",
  "contentPayload": {
    "content": "This is a test message",
    "items": [
      {
        "type": "text",
        "text": "This is a test message",
        "startTimestamp": null,
        "citations": []
      }
    ],
    "metadata": null
  },
  "feedback": "like",
  "createdAt": "1748314926000",
  "attachments": [],
  "citations": [],
  "citationNodes": [],
  "canvas": {
    "id": "034eeecd7-e0f4-46e9-88cb-4fefdb5b7ddd",
    "name": "system-architecture-diagram",
    "canvasType": "markdown",
    "title": "System Architecture Diagram",
    "content": "<!— MERMAID_PLACEHOLDER_START —>
<div class=\"mermaid-diagram\">
  <div class=\"mermaid\">
\ngraph TD\n    A[User] —> B[Frontend Interface]\n    B —> C[API
Layer]\n    C —> D[Database]\n
  </div>
  <p class=\"mermaid-caption\">Mermaid 圖表</p>
</div>

```

```
<!-- MERMAID_PLACEHOLDER_END -->",
  "createdAt": "1748314926000"
}
}
```

Create New Conversation

POST [/api/conversations/](#)

Request Body

Request Parameters

Field	Type	Required	Description
webChat	string (uuid)	Yes	

Request Schema Example

```
{
  "webChat": string (uuid)
}
```

Request Example Value

```
{
  "webChat": "550e8400-e29b-41d4-a716-446655440000"
}
```

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X POST "https://api.maiagent.ai/api/conversations/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "webChat": "550e8400-e29b-41d4-a716-446655440000"
}'

# Please make sure to replace YOUR_API_KEY and verify the request
# data before executing.
```

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "webChat": "550e8400-e29b-41d4-a716-446655440000"
};

axios.post("https://api.maiagent.ai/api/conversations/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });

```

Python

```
import requests

url = "https://api.maiagent.ai/api/conversations/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "webChat": "550e8400-e29b-41d4-a716-446655440000"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/conversations/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "webChat": "550e8400-e29b-41d4-a716-446655440000"
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 201

Response Schema Example

```
{
  "id": string (uuid)
```

```
  "contact": { // Lightweight Contact Serializer for Conversation List
    view

    Only includes fields necessary for List view, removes inboxes and
    mcp_credentials
    to avoid N+1 query issues
    {
      "id": string (uuid)
      "name": string
      "avatar"??: string (uri) // non-required
      "sourceId"??: string // non-required
    }
    }
    "inbox": {
    {
      "id": string (uuid)
      "name": string
      "channelType": string (enum: line, telegram, teams, web, messenger,
      instagram) // * `line` - LINE
      * `telegram` - Telegram
      * `teams` - Teams
      * `web` - Web
      * `messenger` - Messenger
      * `instagram` - Instagram
      "unreadConversationsCount"??: integer // non-required
      "signAuth"??: // non-required
    {
      "id": string (uuid)
      "signSource": string (uuid)
      "signParams": {
      {
        "keycloak":
        {
          "clientId": string
          "url": string
          "realm": string
        }
        "line":
        {
          "liffId": string
        }
        "ad":
        {
          "saml": string
        }
      }
    }
  }
}
```

```

}
}

"title": string
"lastMessage?": { // non-required
{
  "id": string (uuid)
  "type?": string // non-required
  "content?": string // non-required
  "contentPayload?": object // non-required
  "feedback":
  {
    "id": string (uuid)
    "type":
    {
    }
    "suggestion?": string // non-required
    "updatedAt": string (timestamp)
  }
  "createdAt": string (timestamp)
  "queryMetadata?": object // non-required
}
}
"lastMessageCreatedAt": string (timestamp)
"unreadMessagesCount": integer
"autoReplyEnabled": boolean
"isAutoReplyNow": boolean
"lastReadAt": string (timestamp)
"createdAt": string (timestamp)
"isGroupChat": boolean
"enableGroupMention?": boolean // non-required
"queryMetadata?": object // non-required
"toolMask?": object // Records enable/disable status for each tool
{tool_id: bool} (non-required)
  "llm": string (uuid)
  "ipAddress": string
  "ipCountryCode": string
  "ipRecordedAt": string (timestamp)
}

```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "contact": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "avatar": "https://example.com/file.jpg",
}
```

```
        "sourceId": "Response string"
    },
    "inbox": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string",
        "channelType": "line",
        "unreadConversationsCount": 456,
        "signAuth": {
            "id": "550e8400-e29b-41d4-a716-446655440000",
            "signSource": "550e8400-e29b-41d4-a716-446655440000",
            "signParams": {
                "keycloak": {
                    "clientId": "Response string",
                    "url": "Response string",
                    "realm": "Response string"
                },
                "line": {
                    "liffId": "Response string"
                },
                "ad": {
                    "saml": "Response string"
                }
            }
        },
        "title": "Response string",
        "lastMessage": {
            "id": "550e8400-e29b-41d4-a716-446655440000",
            "type": "Response string",
            "content": "Response string",
            "contentPayload": null,
            "feedback": {
                "id": "550e8400-e29b-41d4-a716-446655440000",
                "type": {},
                "suggestion": "Response string",
                "updatedAt": "Response string"
            },
            "createdAt": "Response string",
            "queryMetadata": null
        },
        "lastMessageCreatedAt": "Response string",
        "unreadMessagesCount": 456,
        "autoReplyEnabled": false,
        "isAutoReplyNow": false,
        "lastReadAt": "Response string",
        "createdAt": "Response string",
        "isGroupChat": false,
        "enableGroupMention": false,
```

```
        "queryMetadata": null,  
        "toolMask": null,  
        "llm": "550e8400-e29b-41d4-a716-446655440000",  
        "ipAddress": "Response string",  
        "ipCountryCode": "Response string",  
        "ipRecordedAt": "Response string"  
    }  
}
```

Get Message List

GET

/api/messages/

Parameters

Parameter Name	Required	Type	Description
conversation	V	string	Conversation ID
cursor	X	string	The pagination cursor value.
pageSize	X	integer	Number of results to return per page.

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/messages/?conversation=example&cursor=example&pageSize=1"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request data before executing.
```

JavaScript

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/messages/?conversation=example&cursor=example&pageSize=1", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/messages/?conversation=example&cursor=example&pageSize=1"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/messages/?conversation=example&cursor=example&pageSize=1", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // non-required
  "previous"??: string (uri) // non-required
  "results": [
    {
```

```
"id": string (uuid)
"conversation": string (uuid)
"sender":
{
  "id": string (uuid)
  "name": string
  "avatar": string
  "email?: string (email) // non-required
  "phoneNumber?: string // non-required
}
"type?: string // non-required
"content?: string // non-required
"contentPayload?: object // non-required
"feedback":
{
  "id": string (uuid)
  "type":
  {
  }
  "suggestion?: string // non-required
  "updatedAt": string (timestamp)
}
"createdAt": string (timestamp)
"attachments?: [ // non-required
{
  "id": string (uuid)
  "type?: // non-required
  {
  }
  "filename": string
  "file": string (uri)
  "conversation?: string (uuid) // non-required
}
]
"citations": [
{
  "id": string (uuid)
  "filename": string // Filename
  "file": string (uri) // File to upload
  "fileType": string
  "knowledgeBase?: // non-required
  {
    "id": string (uuid)
    "name": string
  }
  "size": integer
  "status":
  {

```

```
        }
      "parser": {
      {
        "id": string (uuid)
        "name": string
        "provider": {
        {
        }
        "order"??: integer // non-required
      }
    }
    "labels"??: [ // non-required
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "rawUserDefineMetadata"??: object // non-required
  "vectorStorageSize": integer // Size of vectors for this file in
Elasticsearch (bytes)
  "chunksCount": integer // Number of chunks/nodes generated from
this file
  "createdAt": string (timestamp)
}
]
"citationNodes": [
{
  "chatbotTextNode": {
  {
    "id": string (uuid)
    "charactersCount": integer
    "hitsCount": integer
    "text": string
    "updatedAt": string (timestamp)
    "filename": string
    "chatbotFile": string (uuid)
    "knowledgeBaseFile": string (uuid) // Same as
get_chatbot_file, provides backward compatibility
    "pageNumber": integer
  }
}
]
"score"??: number (double) // non-required
"displayScore": integer
}
]
"canvas"??: { // non-required
{
  "id": string (uuid)
```

```
        "name": string
        "canvasType":
        {
        }
        "title": string
        "content": string
        "createdAt": string (timestamp)
    }
}
{
    "queryMetadata?": object // non-required
    "recordId": string // Returns the ChatbotRecord ID corresponding to
the Message
}
]
```

Response Example Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "conversation": "550e8400-e29b-41d4-a716-446655440000",
      "sender": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string",
        "avatar": "Response string",
        "email": "response@example.com",
        "phoneNumber": "Response string"
      },
      "type": "Response string",
      "content": "Response string",
      "contentPayload": null,
      "feedback": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "type": {},
        "suggestion": "Response string",
        "updatedAt": "Response string"
      },
      "createdAt": "Response string",
      "attachments": [
        {
          "id": "550e8400-e29b-41d4-a716-446655440000",
          "type": {}
        }
      ]
    }
  ]
}
```

```
        "filename": "Response string",
        "file": "https://example.com/file.jpg",
        "conversation": "550e8400-e29b-41d4-a716-446655440000"
    }
],
"citations": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "filename": "Response string",
    "file": "https://example.com/file.jpg",
    "fileType": "Response string",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string"
    },
    "size": 456,
    "status": {},
    "parser": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string",
        "provider": {},
        "order": 456
    },
    "labels": [
    {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string"
    }
],
"rawUserDefineMetadata": null,
"vectorStorageSize": 456,
"chunksCount": 456,
"createdAt": "Response string"
}
],
"citationNodes": [
{
    "chatbotTextNode": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "charactersCount": 456,
        "hitsCount": 456,
        "text": "Response string",
        "updatedAt": "Response string",
        "filename": "Response string",
        "chatbotFile": "550e8400-e29b-41d4-a716-446655440000",
        "knowledgeBaseFile": "550e8400-e29b-41d4-a716-446655440000",
        "pageNumber": 456
    }
},
```

```
        "score": 456,
        "displayScore": 456
    }
],
"canvas": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "canvasType": {},
    "title": "Response string",
    "content": "Response string",
    "createdAt": "Response string"
},
"queryMetadata": null,
"recordId": "Response string"
}
]
}
```

Get Specific Message

GET

/api/messages/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this message.

conversation

V

string

Conversation ID

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/messages/550e8400-e29b-
41d4-a716-446655440000/?conversation=example"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request
# data before executing.
```

JavaScript

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/messages/550e8400-e29b-41d4-
a716-446655440000/?conversation=example", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/messages/550e8400-e29b-41d4-a716-
446655440000/?conversation=example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/messages/550e8400-e29b-41d4-a716-
446655440000/?conversation=example", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "conversation": string (uuid)
  "sender": {
    "
```

```
"id": string (uuid)
"name": string
"avatar": string
"email"??: string (email) // non-required
"phoneNumber"??: string // non-required
}
"type"??: string // non-required
"content"??: string // non-required
"contentPayload"??: object // non-required
"feedback":
{
  "id": string (uuid)
  "type":
  {
  }
  "suggestion"??: string // non-required
  "updatedAt": string (timestamp)
}
"createdAt": string (timestamp)
"attachments"??: [ // non-required
{
  "id": string (uuid)
  "type"??: // non-required
  {
  }
  "filename": string
  "file": string (uri)
  "conversation"??: string (uuid) // non-required
}
]
"citations": [
{
  "id": string (uuid)
  "filename": string // Filename
  "file": string (uri) // File to upload
  "fileType": string
  "knowledgeBase"??: // non-required
  {
    "id": string (uuid)
    "name": string
  }
  "size": integer
  "status":
  {
  }
  "parser": {
  }
  "id": string (uuid)
}
```

```
"name": string
"provider":
{
}
"order?": integer // non-required
}
}
"labels?": [ // non-required
{
  "id": string (uuid)
  "name": string
}
]
"rawUserDefineMetadata?": object // non-required
"vectorStorageSize": integer // Size of vectors for this file in
Elasticsearch (bytes)
"chunksCount": integer // Number of chunks/nodes generated from this
file
  "createdAt": string (timestamp)
}
]
"citationNodes": [
{
  "chatbotTextNode": {
    {
      "id": string (uuid)
      "charactersCount": integer
      "hitsCount": integer
      "text": string
      "updatedAt": string (timestamp)
      "filename": string
      "chatbotFile": string (uuid)
      "knowledgeBaseFile": string (uuid) // Same as get_chatbot_file,
provides backward compatibility
      "pageNumber": integer
    }
  }
}
"score?": number (double) // non-required
"displayScore": integer
}
]
"canvas?": { // non-required
{
  "id": string (uuid)
  "name": string
  "canvasType":
  {
  }
}
```

```
        "title": string
        "content": string
        "createdAt": string (timestamp)
    }
}
"queryMetadata"? object // non-required
"recordId": string // Returns the ChatbotRecord ID corresponding to the
Message
}
```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "conversation": "550e8400-e29b-41d4-a716-446655440000",
  "sender": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "avatar": "Response string",
    "email": "response@example.com",
    "phoneNumber": "Response string"
  },
  "type": "Response string",
  "content": "Response string",
  "contentPayload": null,
  "feedback": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "type": {},
    "suggestion": "Response string",
    "updatedAt": "Response string"
  },
  "createdAt": "Response string",
  "attachments": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "type": {},
      "filename": "Response string",
      "file": "https://example.com/file.jpg",
      "conversation": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "citations": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "filename": "Response string",
      "file": "https://example.com/file.jpg",
      "fileType": "Response string",
    }
  ]
}
```

```
"knowledgeBase": [
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
},
"size": 456,
"status": {},
"parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "provider": {},
    "order": 456
},
"labels": [
    {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string"
    }
],
"rawUserDefineMetadata": null,
"vectorStorageSize": 456,
"chunksCount": 456,
"createdAt": "Response string"
},
],
"citationNodes": [
{
    "chatbotTextNode": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "charactersCount": 456,
        "hitsCount": 456,
        "text": "Response string",
        "updatedAt": "Response string",
        "filename": "Response string",
        "chatbotFile": "550e8400-e29b-41d4-a716-446655440000",
        "knowledgeBaseFile": "550e8400-e29b-41d4-a716-446655440000",
        "pageNumber": 456
    },
    "score": 456,
    "displayScore": 456
}
],
"canvas": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "canvasType": {},
    "title": "Response string",
    "content": "Response string",
    "createdAt": "Response string"
```

```
},
"queryMetadata": null,
"recordId": "Response string"
}
```

Get Conversation List

GET

/api/conversations/

Parameters

Parameter Name	Required	Type	Description
cursor	X	string	The pagination cursor value.
inbox	V	string	Conversation platform ID
keyword	X	string	Keyword search
pageSize	X	integer	Number of results to return per page.

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/conversations/?cursor=example&inbox=example&keyword=example&pageSize=1"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request data before executing.
```

JavaScript

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/conversations/?cursor=example&inbox=example&keyword=example&pageSize=1", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/conversations/?cursor=example&inbox=example&keyword=example&pageSize=1"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/conversations/?cursor=example&inbox=example&keyword=example&pageSize=1", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // non-required
  "previous"??: string (uri) // non-required
  "results": [
```

```
{
  "id": string (uuid)
  "contact": { // Lightweight Contact Serializer for Conversation List
view

Only includes fields necessary for List view, removes inboxes and
mcp_credentials
to avoid N+1 query issues
{
  "id": string (uuid)
  "name": string
  "avatar?: string (uri) // non-required
  "sourceId?: string // non-required
}
}
}

"inbox": {
{
  "id": string (uuid)
  "name": string
  "channelType": string (enum: line, telegram, teams, web,
messenger, instagram) // * `line` - LINE
* `telegram` - Telegram
* `teams` - Teams
* `web` - Web
* `messenger` - Messenger
* `instagram` - Instagram
  "unreadConversationsCount?: integer // non-required
  "signAuth?: // non-required
{
  "id": string (uuid)
  "signSource": string (uuid)
  "signParams": {
  {
    "keycloak":
    {
      "clientId": string
      "url": string
      "realm": string
    }
    "line":
    {
      "liffId": string
    }
    "ad":
    {
      "saml": string
    }
  }
}
```

```

        }
    }
}

"title": string
"lastMessage?": { // non-required
{
    "id": string (uuid)
    "type?": string // non-required
    "content?": string // non-required
    "contentPayload?": object // non-required
    "feedback":
    {
        "id": string (uuid)
        "type":
        {
        }
        "suggestion?": string // non-required
        "updatedAt": string (timestamp)
    }
    "createdAt": string (timestamp)
    "queryMetadata?": object // non-required
}
}
{
    "lastMessageCreatedAt": string (timestamp)
    "unreadMessagesCount": integer
    "autoReplyEnabled": boolean
    "isAutoReplyNow": boolean
    "lastReadAt": string (timestamp)
    "createdAt": string (timestamp)
    "isGroupChat": boolean
    "enableGroupMention?": boolean // non-required
    "queryMetadata?": object // non-required
    "toolMask?": object // Records enable/disable status for each tool
    {tool_id: bool} (non-required)
    "llm": string (uuid)
    "ipAddress": string
    "ipCountryCode": string
    "ipRecordedAt": string (timestamp)
}
]
}

```

Response Example Value

```
{
    "count": 123,
```

```
"next": "http://api.example.org/accounts/?page=4",
"previous": "http://api.example.org/accounts/?page=2",
"results": [
  {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "contact": {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string",
      "avatar": "https://example.com/file.jpg",
      "sourceId": "Response string"
    },
    "inbox": {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string",
      "channelType": "line",
      "unreadConversationsCount": 456,
      "signAuth": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "signSource": "550e8400-e29b-41d4-a716-446655440000",
        "signParams": {
          "keycloak": {
            "clientId": "Response string",
            "url": "Response string",
            "realm": "Response string"
          },
          "line": {
            "liffId": "Response string"
          },
          "ad": {
            "saml": "Response string"
          }
        }
      }
    },
    "title": "Response string",
    "lastMessage": {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "type": "Response string",
      "content": "Response string",
      "contentPayload": null,
      "feedback": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "type": {},
        "suggestion": "Response string",
        "updatedAt": "Response string"
      },
      "createdAt": "Response string",
      "queryMetadata": null
    }
  }
]
```

```
        },
        "lastMessageCreatedAt": "Response string",
        "unreadMessagesCount": 456,
        "autoReplyEnabled": false,
        "isAutoReplyNow": false,
        "lastReadAt": "Response string",
        "createdAt": "Response string",
        "isGroupChat": false,
        "enableGroupMention": false,
        "queryMetadata": null,
        "toolMask": null,
        "llm": "550e8400-e29b-41d4-a716-446655440000",
        "ipAddress": "Response string",
        "ipCountryCode": "Response string",
        "ipRecordedAt": "Response string"
    }
]
}
```

Get Specific Conversation

GET

/api/conversations/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this conversation.

Code Examples

Shell/Bash

```
# API call example (Shell)
curl -X GET "https://api.maiagent.ai/api/conversations/550e8400-
e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please make sure to replace YOUR_API_KEY and verify the request
data before executing.
```

JavaScript

```
const axios = require('axios');

// Configure request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/conversations/550e8400-e29b-
41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request error occurred:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/conversations/550e8400-e29b-41d4-
a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request error occurred:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/conversations/550e8400-e29b-41d4-
a716-446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request error occurred: ' . $e->getMessage();
}
?>
```

Response

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "contact": { // Lightweight Contact Serializer for Conversation List
    view
```

```
Only includes fields necessary for List view, removes inboxes and
mcp_credentials
to avoid N+1 query issues
{
  "id": string (uuid)
  "name": string
  "avatar?: string (uri) // non-required
  "sourceId?: string // non-required
}
}
{
  "inbox": {
    {
      "id": string (uuid)
      "name": string
      "channelType": string (enum: line, telegram, teams, web, messenger,
instagram) // * `line` - LINE
* `telegram` - Telegram
* `teams` - Teams
* `web` - Web
* `messenger` - Messenger
* `instagram` - Instagram
      "unreadConversationsCount?: integer // non-required
      "signAuth?: // non-required
    {
      "id": string (uuid)
      "signSource": string (uuid)
      "signParams": {
        {
          "keycloak":
            {
              "clientId": string
              "url": string
              "realm": string
            }
          "line":
            {
              "liffId": string
            }
          "ad":
            {
              "saml": string
            }
        }
      }
    }
  }
}
}
"title": string
```

```

"lastMessage"?:
{
  "id": string (uuid)
  "type"??: string // non-required
  "content"??: string // non-required
  "contentPayload"??: object // non-required
  "feedback":
  {
    "id": string (uuid)
    "type":
    {
    }
    "suggestion"??: string // non-required
    "updatedAt": string (timestamp)
  }
  "createdAt": string (timestamp)
  "queryMetadata"??: object // non-required
}
}

"lastMessageCreatedAt": string (timestamp)
"unreadMessagesCount": integer
"autoReplyEnabled": boolean
"isAutoReplyNow": boolean
"lastReadAt": string (timestamp)
"createdAt": string (timestamp)
"isGroupChat": boolean
"enableGroupMention"??: boolean // non-required
"queryMetadata"??: object // non-required
"toolMask"??: object // Records enable/disable status for each tool
{tool_id: bool} (non-required)
  "llm": string (uuid)
  "ipAddress": string
  "ipCountryCode": string
  "ipRecordedAt": string (timestamp)
}

```

Response Example Value

```

{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "contact": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "avatar": "https://example.com/file.jpg",
    "sourceId": "Response string"
  },
  "inbox": {

```

```
"id": "550e8400-e29b-41d4-a716-446655440000",
"name": "Response string",
"channelType": "line",
"unreadConversationsCount": 456,
"signAuth": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "signSource": "550e8400-e29b-41d4-a716-446655440000",
    "signParams": {
        "keycloak": {
            "clientId": "Response string",
            "url": "Response string",
            "realm": "Response string"
        },
        "line": {
            "liffId": "Response string"
        },
        "ad": {
            "saml": "Response string"
        }
    }
},
"title": "Response string",
"lastMessage": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "type": "Response string",
    "content": "Response string",
    "contentPayload": null,
    "feedback": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "type": {},
        "suggestion": "Response string",
        "updatedAt": "Response string"
    },
    "createdAt": "Response string",
    "queryMetadata": null
},
"lastMessageCreatedAt": "Response string",
"unreadMessagesCount": 456,
"autoReplyEnabled": false,
"isAutoReplyNow": false,
"lastReadAt": "Response string",
"createdAt": "Response string",
"isGroupChat": false,
"enableGroupMention": false,
"queryMetadata": null,
"toolMask": null,
"llm": "550e8400-e29b-41d4-a716-446655440000",
```

```
"ipAddress": "Response string",
"ipCountryCode": "Response string",
"ipRecordedAt": "Response string"
}
```

Knowledge Base (New)

Create Knowledge Base

POST [/api/knowledge-bases/](#)

Request Body

Parameters

Field	Type	Required	Description
embeddingModel	string (uuid)	Yes	
rerankerModel	string (uuid)	Yes	
name	string	Yes	
description	string	No	
numberOfRetrievedChunks	integer	No	Number of retrieved reference chunks, default is 12, minimum is 1
sentenceWindowSize	integer	No	RAG augmented sentence window size, default is 2, minimum is 0
enableHyde	boolean	No	Enable HyDE, default is False
similarityCutoff	number (double)	No	Similarity cutoff, default is 0.0, range is 0.0-1.0
enableRerank	boolean	No	Enable reranking, default is True
chatbots	array[IdName]	No	

Field	Type	Required	Description
groups	array[IdName]	No	

Request Structure Example

```
{
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description"?: string // Optional
  "numberOfRetrievedChunks"?: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize"?: integer // RAG augmented sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde"?: boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff"?: number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank"?: boolean // Whether to enable reranking, default is True (Optional)
  "chatbots"?: [ // Optional
    {
      "id": string (uuid)
    }
  ]
  "groups"?: [ // Optional
    {
      "id": string (uuid)
    }
  ]
}
}
```

Request Example Value

```
{
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example string",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440001"
    }
  ]
}
```

```
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
],
"groups": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000"
}
]
}
```

Code Examples

```
# Call API Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}'.

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
};
```



```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": True,
    "similarityCutoff": 123,
    "enableRerank": True,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
            "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Example Name",
            "description": "Example String",
            "numberOfRetrievedChunks": 123,
            "sentenceWindowSize": 123,
            "enableHyde": true,
            "similarityCutoff": 123,
            "enableRerank": true,
            "chatbots": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "groups": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ]
        ]
    ]);
}

$data = json_decode($response→getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e→getMessage();
}
?>
```

Response Body

Status Code: 201

Response Schema Example

```
{
  "id": string (uuid)
  "user?": string (uuid) // Optional
  "organization?": string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?": string // Optional
  "numberOfRetrievedChunks?": integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?": integer // RAG sentence window size for expansion, default is 2, minimum is 0 (Optional)
  "enableHyde?": boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff?": number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank?": boolean // Enable reranking, default is True (Optional)
  "labels": [
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "chatbots?": [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "groups?": [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "filesCount": integer // Returns the number of ChatbotFiles under this knowledge base (excluding processed files)
  "createdAt": string (timestamp)
```

```
    "updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "filesCount": 456,
  "createdAt": "Response string",
  "updatedAt": "Response string"
}
```

- ### List all knowledge bases

GET </api/knowledge-bases/>

Parameters

Parameter Name	Required	Type	Description
embeddingModel	X	string	
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	
rerankerModel	X	string	

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/?
embeddingModel=550e8400-e29b-41d4-a716-
44665544000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-
41d4-a716-44665544000"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/?embeddingModel=550e8400-e29b-41d4-a716-44665544000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-41d4-a716-44665544000", config)
  .then(response => {
    console.log('Successfully got response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('Request failed:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/?embeddingModel=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-41d4-a716-446655440000"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/?embeddingModel=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&rerankerModel=550e8400-e29b-
41d4-a716-446655440000", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "user"??: string (uuid) // Optional
      "organization"??: string (uuid) // Optional
    }
  ]
}
```

```

    "embeddingModel": string (uuid)
    "rerankerModel": string (uuid)
    "name": string
    "description"??: string // Optional
    "numberOfRetrievedChunks"??: integer // The number of retrieved reference
chunks, default is 12, minimum is 1 (Optional)
    "sentenceWindowSize"??: integer // RAG augmented sentence window size,
default is 2, minimum is 0 (Optional)
    "enableHyde"??: boolean // Enable HyDE, default is False (Optional)
    "similarityCutoff"??: number (double) // Similarity cutoff, default is
0.0, range is 0.0-1.0 (Optional)
    "enableRerank"??: boolean // Whether to enable reranking, default is True
(Optional)
    "labels": [
        {
            "id": string (uuid)
            "name": string
        }
    ]
    "chatbots"??: [ // Optional
        {
            "id": string (uuid)
            "name": string
        }
    ]
    "groups"??: [ // List of groups that can access the knowledge base
(Optional)
        {
            "id": string (uuid)
            "name": string
        }
    ]
    "filesCount": integer // Returns the number of ChatbotFiles under this
knowledge base (excluding processed files)
    "createdAt": string (timestamp)
    "updatedAt": string (timestamp)
}
]
}

```

Response Example Value

```
{
    "count": 123,
    "next": "http://api.example.org/accounts/?page=4",
    "previous": "http://api.example.org/accounts/?page=2",
    "results": [
        ...
    ]
}
```

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "filesCount": 456,
  "createdAt": "Response string",
  "updatedAt": "Response string"
}
]
```

- ### Get Specific Knowledge Base Details

GET `/api/knowledge-bases/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Knowledge Base.

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

JavaScript

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Success:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Error: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "user?: string (uuid) // Optional
  "organization?: string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?: string // Optional
  "numberOfRetrievedChunks?: integer // Number of retrieved reference chunks,
default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?: integer // RAG augmented sentence window size, default
```

```

is 2, minimum is 0 (Optional)
"enableHyde"?: boolean // Enable HyDE, default is False (Optional)
"similarityCutoff"?: number (double) // Similarity cutoff, default is 0.0,
range is 0.0-1.0 (Optional)
"enableRerank"?: boolean // Enable rerank, default is True (Optional)
"labels": [
{
  "id": string (uuid)
  "name": string
}
]
"chatbots"?: [ // Optional
{
  "id": string (uuid)
  "name": string
}
]
"groups"?: [ // List of groups that can access the knowledge base (Optional)
{
  "id": string (uuid)
  "name": string
}
]
"filesCount": integer // Returns the number of ChatbotFiles under this
knowledge base (excluding processed files)
"createdAt": string (timestamp)
"updatedAt": string (timestamp)
}

```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Label 1"
}
]
}
```

```

        "name": "Response string"
    }
],
"chatbots": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
}
],
"groups": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
}
],
"filesCount": 456,
"createdAt": "Response string",
"updatedAt": "Response string"
}

```

- ### Update Knowledge Base

PUT `/api/knowledge-bases/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Knowledge Base.

Request Body

Request Parameters

Field	Type	Required	Description
<code>user</code>	string (uuid)	No	
<code>organization</code>	string (uuid)	No	
<code>embeddingModel</code>	string (uuid)	Yes	
<code>rerankerModel</code>	string (uuid)	Yes	
<code>name</code>	string	Yes	
<code>description</code>	string	No	

Field	Type	Required	Description
numberOfRetrievedChunks	integer	No	Number of retrieved reference chunks, default is 12, minimum is 1
sentenceWindowSize	integer	No	RAG augmented sentence window size, default is 2, minimum is 0
enableHyde	boolean	No	Enable HyDE, default is False
similarityCutoff	number (double)	No	Similarity cutoff, default is 0.0, range is 0.0-1.0
enableRerank	boolean	No	Whether to enable reranking, default is True
chatbots	array[IdName]	No	
groups	array[IdName]	No	List of groups that can access the knowledge base

Request Structure Example

```
{
  "user"?: string (uuid) // Optional
  "organization"?: string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description"?: string // Optional
  "numberOfRetrievedChunks"?: integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize"?: integer // RAG augmented sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde"?: boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff"?: number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank"?: boolean // Whether to enable rerank, default is True (Optional)
  "chatbots"?: [ // Optional
    {
      "id": string (uuid)
    }
  ]
  "groups"?: [ // List of groups that can access the knowledge base (Optional)
    {
      ...
    }
  ]
}
```

```
        "id": string (uuid)
    }
]
}
```

Request Example Value

```
{
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
}
```

Code Examples

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}'.

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Sample Name",
  "description": "Sample String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
  })
```

```
    console.error(error.response?.data || error.message);
});
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Sample Name",
    "description": "Sample String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```



```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "user": "550e8400-e29b-41d4-a716-446655440000",
            "organization": "550e8400-e29b-41d4-a716-446655440000",
            "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
            "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Sample Name",
            "description": "Sample String",
            "numberOfRetrievedChunks": 123,
            "sentenceWindowSize": 123,
            "enableHyde": true,
            "similarityCutoff": 123,
            "enableRerank": true,
            "chatbots": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "groups": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
```

```
}
```

```
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "user?": string (uuid) // Optional
  "organization?": string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?": string // Optional
  "numberOfRetrievedChunks?": integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?": integer // RAG sentence window size, default is 2, minimum is 0 (Optional)
  "enableHyde?": boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff?": number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank?": boolean // Enable rerank, default is True (Optional)
  "labels": [
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "chatbots?": [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "groups?": [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "filesCount": integer // Returns the number of ChatbotFiles under this
```

knowledge base (excluding processed files)

```
"createdAt": string (timestamp)
"updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "filesCount": 456,
  "createdAt": "Response string",
  "updatedAt": "Response string"
}
```

- ### Partially Update a Knowledge Base

PATCH `/api/knowledge-bases/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Knowledge Base.

Request Body

Request Parameters

Field	Type	Required	Description
<code>user</code>	string (uuid)	No	
<code>organization</code>	string (uuid)	No	
<code>embeddingModel</code>	string (uuid)	No	
<code>rerankerModel</code>	string (uuid)	No	
<code>name</code>	string	No	
<code>description</code>	string	No	
<code>numberOfRetrievedChunks</code>	integer	No	Number of retrieved reference chunks, default is 12, minimum is 1
<code>sentenceWindowSize</code>	integer	No	RAG augmented sentence window size, default is 2, minimum is 0
<code>enableHyde</code>	boolean	No	Enable HyDE, default is False
<code>similarityCutoff</code>	number (double)	No	Similarity cutoff threshold, default is 0.0, range is 0.0-1.0
<code>enableRerank</code>	boolean	No	Whether to enable reranking, default is True
<code>chatbots</code>	array[IdName]	No	
<code>groups</code>	array[IdName]	No	List of groups that can access the knowledge base

Request Schema Example

```
{
  "user"??: string (uuid) // Optional
  "organization"??: string (uuid) // Optional
  "embeddingModel"??: string (uuid) // Optional
  "rerankerModel"??: string (uuid) // Optional
  "name"??: string // Optional
  "description"??: string // Optional
  "numberOfRetrievedChunks"??: integer // Number of retrieved reference chunks,
  default is 12, minimum is 1 (Optional)
  "sentenceWindowSize"??: integer // RAG augmented sentence window size, default
  is 2, minimum is 0 (Optional)
  "enableHyde"??: boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff"??: number (double) // Similarity cutoff, default is 0.0,
  range is 0.0-1.0 (Optional)
  "enableRerank"??: boolean // Whether to enable reranking, default is True
  (Optional)
  "chatbots"??: [ // Optional
    {
      "id": string (uuid)
    }
  ]
  "groups"??: [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
    }
  ]
}
```

Request Example Value

```
{
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Example Name",
  "description": "Example string",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
}
```

```
],
"groups": [
{
  "id": "550e8400-e29b-41d4-a716-446655440000"
}
]
```

Code Examples

```
# Call API Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Example Name",
    "description": "Example String",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}'.

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Sample Name",
  "description": "Sample String",
  "numberOfRetrievedChunks": 123,
  "sentenceWindowSize": 123,
  "enableHyde": true,
  "similarityCutoff": 123,
  "enableRerank": true,
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ]
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
  })
```

```
    console.error(error.response?.data || error.message);
});
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "user": "550e8400-e29b-41d4-a716-446655440000",
    "organization": "550e8400-e29b-41d4-a716-446655440000",
    "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
    "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
    "name": "範例名稱",
    "description": "範例字串",
    "numberOfRetrievedChunks": 123,
    "sentenceWindowSize": 123,
    "enableHyde": true,
    "similarityCutoff": 123,
    "enableRerank": true,
    "chatbots": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "groups": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ]
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```



```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "user": "550e8400-e29b-41d4-a716-446655440000",
            "organization": "550e8400-e29b-41d4-a716-446655440000",
            "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
            "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
            "name": "Example Name",
            "description": "Example String",
            "numberOfRetrievedChunks": 123,
            "sentenceWindowSize": 123,
            "enableHyde": true,
            "similarityCutoff": 123,
            "enableRerank": true,
            "chatbots": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "groups": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
```

```
}
```

```
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "user?": string (uuid) // Optional
  "organization?": string (uuid) // Optional
  "embeddingModel": string (uuid)
  "rerankerModel": string (uuid)
  "name": string
  "description?": string // Optional
  "numberOfRetrievedChunks?": integer // Number of retrieved chunks, default is 12, minimum is 1 (Optional)
  "sentenceWindowSize?": integer // RAG sentence window size for expansion, default is 2, minimum is 0 (Optional)
  "enableHyde?": boolean // Enable HyDE, default is False (Optional)
  "similarityCutoff?": number (double) // Similarity cutoff, default is 0.0, range is 0.0-1.0 (Optional)
  "enableRerank?": boolean // Enable rerank, default is True (Optional)
  "labels": [
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "chatbots?": [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "groups?": [ // List of groups that can access the knowledge base (Optional)
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "filesCount": integer // Returns the number of ChatbotFiles under this
```

knowledge base (excluding processed files)

```
"createdAt": string (timestamp)
"updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "user": "550e8400-e29b-41d4-a716-446655440000",
  "organization": "550e8400-e29b-41d4-a716-446655440000",
  "embeddingModel": "550e8400-e29b-41d4-a716-446655440000",
  "rerankerModel": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response string",
  "description": "Response string",
  "numberOfRetrievedChunks": 456,
  "sentenceWindowSize": 456,
  "enableHyde": false,
  "similarityCutoff": 456,
  "enableRerank": false,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "chatbots": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "groups": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "filesCount": 456,
  "createdAt": "Response string",
  "updatedAt": "Response string"
}
```

- ### Delete Knowledge Base

DELETE /api/knowledge-bases/{id}/

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Knowledge Base.

Code Examples

Shell/Bash

```
# Example API Call (Shell)
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response

Status Code	Description
204	No response body

*

Create a New Label

POST /api/knowledge-bases/{knowledgeBasePk}/labels/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
name	string	Yes	

Request Structure Example

```
{  
  "name": string  
}
```

Request Example Value

```
{  
  "name": "Sample Name"  
}
```

Code Examples

```
# Call API Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/labels/"
  -H "Authorization: Api-Key YOUR_API_KEY"
  -H "Content-Type: application/json"
  -d '{
    "name": "Example Name"
  }'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "name": "Example Name"
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "name": "Example Name"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "name": "Sample Name"
        ]
    ]);

    $data = json_decode($response→getBody(), true);
    echo "Successfully received response:\n";
    print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e→getMessage();
}
?>

```

Response Body

Status Code: 201

Response Schema Example

```
{
  "id": string (uuid)
  "name?: string // Optional
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response String"  
}
```

- ### List labels in a knowledge base

GET </api/knowledge-bases/{knowledgeBasePk}/labels/>

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.

Code Examples

Shell/Bash

```
# API Call Example (Shell)  
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/labels/?page=1&pageSize=1"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/?page=1&pageSize=1", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/?page=1&pageSize=1"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/?page=1&pageSize=1", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully retrieved response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "name"??: string // Optional
    }
  ]
}
```

```
]  
}
```

Response Example Value

```
{  
  "count": 123,  
  "next": "http://api.example.org/accounts/?page=4",  
  "previous": "http://api.example.org/accounts/?page=2",  
  "results": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response String"  
    }  
  ]  
}
```

- ### Get Specific Label Details

GET `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

JavaScript

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully retrieved response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "name"??: string // Optional
}
```

Example Response Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
```

```
        "name": "Response String"  
    }
```

- ### Update Label

PUT `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>name</code>	string	Yes	

Request Structure Example

```
{  
  "name": string  
}
```

Request Example Value

```
{  
  "name": "Example Name"  
}
```

Code Examples

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "name": "Example Name"
}'

# Please replace YOUR_API_KEY and check the request data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "name": "Example Name"
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "name": "Example Name"
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "name": "Sample Name"
    ]
]);

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "name"??: string // Optional
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response String"  
}
```

- ### Partially Update Label

PATCH `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>name</code>	string	No	

Request Structure Example

```
{  
  "name"??: string // Optional  
}
```

Request Example Value

```
{  
  "name": "Example Name"  
}
```

Code Examples

```
# Call API Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "name": "Example Name"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "name": "Example Name"
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/",
  data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "name": "Example Name"
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "name": "Example Name"
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "name": string // Optional
}
```

Example Response Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "name": "Response String"  
}
```

- ### Delete Label

DELETE `/api/knowledge-bases/{knowledgeBasePk}/labels/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this knowledge base label.
<code>knowledgeBasePk</code>	V	string	

Code Examples

Shell/Bash

```
# API Call Example (Shell)  
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/",
  data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/labels/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response

Status Code	Description
204	No response body

*

Upload File to Knowledge Base

POST /api/knowledge-bases/{knowledgeBasePk}/files/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
fileType	X	string	Filter by file type, e.g., pdf, docx, txt, xlsx, etc.
knowledgeBase	X	string	Knowledge Base ID
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	Search keyword, can search by file name or file ID.
status	X	string	Filter by file status. Possible values: initial (Initializing), processing (Processing), done (Done), deleting (Deleting), deleted (Deleted), failed (Failed). initial : Initial ; processing : Processing ; done : Done ; deleting : Deleting ; deleted : Deleted ; `faile...

Request Body

Request Parameters

Field	Type	Required	Description
files	array[ChatbotFileCreate]	Yes	Supports batch upload of multiple files, with a limit of 50 files per request

Request Structure Example

```
{
  "files": [ // Supports batch upload of multiple files, with a limit of 50
    files per request
    {
      "filename": string // File name
      "file": string (uri) // The file to be uploaded
      "knowledgeBase?": // Optional
      {
        "id": string (uuid)
      }
      "parser?": string (uuid) // Optional
      "labels?": [ // Optional
      {
        "id": string (uuid)
      }
    }
  ]
}
```

```
        }
    ]
    "rawUserDefineMetadata"??: object // Optional
}
]
}
```

Request Example Value

```
{
  "files": [
    {
      "filename": "my-file.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "my-knowledge-base"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "labels": [
        {
          "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
          "name": "my-label"
        }
      ],
      "rawUserDefineMetadata": "string"
    },
    {
      "filename": "you-can-upload-multiple-files.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "string"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
    }
  ]
}
```

Code Examples

```
# Call API Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "files": [
    {
      "filename": "my-file.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "my-knowledge-base"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "labels": [
        {
          "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
          "name": "my-label"
        }
      ],
      "rawUserDefineMetadata": "string"
    },
    {
      "filename": "you-can-upload-multiple-files.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "string"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
    }
  ]
}'.

# Please replace YOUR_API_KEY and verify the request data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "files": [
    {
      "filename": "my-file.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "my-knowledge-base"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
      "labels": [
        {
          "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
          "name": "my-label"
        }
      ],
      "rawUserDefineMetadata": "string"
    },
    {
      "filename": "you-can-upload-multiple-files.pdf",
      "file": "string",
      "knowledgeBase": {
        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
        "name": "string"
      },
      "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
    }
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-
```

```
e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", data,
config)
.then(response => {
  console.log('Successfully received response:');
  console.log(response.data);
})
.catch(error => {
  console.error('An error occurred with the request:');
  console.error(error.response ?.data || error.message);
});
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "files": [
        {
            "filename": "my-file.pdf",
            "file": "string",
            "knowledgeBase": {
                "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                "name": "my-knowledge-base"
            },
            "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
            "labels": [
                {
                    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                    "name": "my-label"
                }
            ],
            "rawUserDefineMetadata": "string"
        },
        {
            "filename": "you-can-upload-multiple-files.pdf",
            "file": "string",
            "knowledgeBase": {
                "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                "name": "string"
            },
            "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
        }
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
```

```
print("Successfully received response:")
print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/?
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "files": [
            {
                "filename": "my-file.pdf",
                "file": "string",
                "knowledgeBase": {
                    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                    "name": "my-knowledge-base"
                },
                "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                "labels": [
                    {
                        "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                        "name": "my-label"
                    }
                ],
                "rawUserDefineMetadata": "string"
            },
            {
                "filename": "you-can-upload-multiple-files.pdf",
                "file": "string",
                "knowledgeBase": {
                    "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
                    "name": "string"
                },
                "parser": "3fa85f64-5717-4562-b3fc-2c963f66afa6"
            }
        ]
    }
}]
```

```
]);  
  
    $data = json_decode($response->getBody(), true);  
    echo "Successfully received response:\n";  
    print_r($data);  
} catch (Exception $e) {  
    echo 'Request failed: ' . $e->getMessage();  
}  
?>
```

Response Body

Status Code: 201

Response Schema Example

```
{  
    "count": integer  
    "next"??: string (uri) // Optional  
    "previous"??: string (uri) // Optional  
    "results": [  
        {  
            "id": string (uuid)  
            "filename": string // File name  
            "file": string (uri) // File to upload  
            "fileType": string  
            "knowledgeBase"??: // Optional  
            {  
                "id": string (uuid)  
                "name": string  
            }  
            "size": integer  
            "status":  
            {  
            }  
            "parser": {  
                {  
                    "id": string (uuid)  
                    "name": string  
                    "provider":  
                    {  
                    }  
                    "order"??: integer // Optional  
                }  
            }  
        }  
    ]  
}
```

```
        }
      "labels"? : [ // Optional
        {
          "id": string (uuid)
          "name": string
        }
      ]
    "rawUserDefineMetadata"? : object // Optional
    "createdAt": string (timestamp)
  }
]
}
```

Response Example Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "filename": "Response String",
      "file": "https://example.com/file.jpg",
      "fileType": "Response String",
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String"
      },
      "size": 456,
      "status": {},
      "parser": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String",
        "provider": {},
        "order": 456
      },
      "labels": [
        {
          "id": "550e8400-e29b-41d4-a716-446655440000",
          "name": "Response String"
        }
      ],
      "rawUserDefineMetadata": null,
      "createdAt": "Response String"
    }
  ]
}
```

```
    ]  
}
```

- ### Batch Delete Files

POST /api/knowledge-bases/{knowledgeBasePk}/files/batch-delete/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
ids	array[string]	Yes	

Request Structure Example

```
{  
  "ids": [  
    string (uuid)  
  ]  
}
```

Request Example Value

```
{  
  "ids": [  
    "550e8400-e29b-41d4-a716-446655440000"  
  ]  
}
```

Code Examples

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/batch-delete/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/batch-delete/" , data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/batch-delete/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "ids": [
        "550e8400-e29b-41d4-a716-446655440000"
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/batch-delete/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "ids": [
                "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response→getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Error during request: ' . $e→getMessage();
}
?>

```

Response Body

Status Code	Description
204	No response body

*

List Files in a Knowledge Base

GET /api/knowledge-bases/{knowledgeBasePk}/files/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
fileType	X	string	Filter by file type, e.g., pdf, docx, txt, xlsx, etc.
knowledgeBase	X	string	Knowledge Base ID
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	Search keyword, can search by file name or file ID
status	X	string	Filter by file status. Available values: <code>initial</code> : Initial ; <code>processing</code> : Processing ; <code>done</code> : Done ; <code>deleting</code> : Deleting ; <code>deleted</code> : Deleted ; `faile...

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted",
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/?
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "filename": string // File name
      "file": string (uri) // File to upload
    }
  ]
}
```

```

    "fileType": string
    "knowledgeBase"??: // Optional
    {
      "id": string (uuid)
      "name": string
    }
    "size": integer
    "status":
    {
    }
    "parser": {
    {
      "id": string (uuid)
      "name": string
      "provider":
      {
      }
      "order"??: integer // Optional
    }
    }
    "labels"??: [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "rawUserDefineMetadata"??: object // Optional
  "createdAt": string (timestamp)
}
]
}

```

Example Response Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "filename": "Response String",
      "file": "https://example.com/file.jpg",
      "fileType": "Response String",
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response String"
      }
    }
  ]
}
```

```

},
"size": 456,
"status": {},
"parser": {
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "name": "Response String",
  "provider": {},
  "order": 456
},
"labels": [
  {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
  }
],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}
]
}

```

- ### Get Specific File Details

GET `/api/knowledge-bases/{knowledgeBasePk}/files/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Chatbot File.
<code>knowledgeBasePk</code>	V	string	

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

JavaScript

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-
a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully retrieved response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "filename": string // File name
  "file": string (uri) // File to upload
  "fileType": string
  "knowledgeBase"??: // Optional
  {
    "id": string (uuid)
    "name": string
  }
}
```

```

  "size": integer
  "status":
  {
  }
  "parser": {
  {
    "id": string (uuid)
    "name": string
    "provider":
    {
    }
    "order"??: integer // Optional
  }
}
"labels"??: [ // Optional
{
  "id": string (uuid)
  "name": string
}
]
"rawUserDefineMetadata"??: object // Optional
"createdAt": string (timestamp)
}

```

Response Example Value

```
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "filename": "Response string",
  "file": "https://example.com/file.jpg",
  "fileType": "Response string",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
  },
  "size": 456,
  "status": {},
  "parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string",
    "provider": {},
    "order": 456
  },
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ]
}
```

```

    },
    "rawUserDefineMetadata": null,
    "createdAt": "Response string"
}

```

- ### Update File Information

PUT /api/knowledge-bases/{knowledgeBasePk}/files/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Chatbot File.
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
filename	string	Yes	File name
file	string (uri)	Yes	The file to upload
knowledgeBase	object (contains 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	
parser	string (uuid)	No	
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	

Request Structure Example

```
{
  "filename": string // File name
  "file": string (uri) // The file to upload
  "knowledgeBase?": // Optional
  {
    ...
  }
}
```

```
        "id": string (uuid)
    }
    "parser"??: string (uuid) // Optional
    "labels"??: [ // Optional
        {
            "id": string (uuid)
        }
    ]
    "rawUserDefineMetadata"??: object // Optional
}
```

Request Example Value

```
{
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}
```

Code Examples

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully got response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "filename": "document.pdf",
            "file": "https://example.com/file.jpg",
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ],
            "parser": "550e8400-e29b-41d4-a716-446655440000",
            "labels": [
                [
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                ]
            ],
            "rawUserDefineMetadata": null
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{  
  "id": string (uuid)  
  "filename": string // File name  
  "file": string (uri) // File to upload  
  "fileType": string  
  "knowledgeBase"??: // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
  "size": integer  
  "status":  
  {  
  }  
  "parser": {  
    "id": string (uuid)  
    "name": string  
    "provider":  
    {  
    }  
    "order"??: integer // Optional  
  }  
  "labels"??: [ // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
]  
  "rawUserDefineMetadata"??: object // Optional  
  "createdAt": string (timestamp)  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "filename": "Response String",  
  "file": "https://example.com/file.jpg",  
  "fileType": "Response String",  
  "knowledgeBase": {
```

```

    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
},
"size": 456,
"status": {},
"parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String",
    "provider": {},
    "order": 456
},
"labels": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
}
],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}

```

- ### Partially Update File

PATCH [/api/knowledge-bases/{knowledgeBasePk}/files/{id}/](#)

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Chatbot File.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>filename</code>	string	No	File name
<code>file</code>	string (uri)	No	The file to upload
<code>knowledgeBase</code>	object (contains 2 properties: id, name)	No	
<code>knowledgeBase.id</code>	string (uuid)	Yes	

Field	Type	Required	Description
parser	string (uuid)	No	
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	

Request Structure Example

```
{
  "filename"??: string // File name (Optional)
  "file"??: string (uri) // File to upload (Optional)
  "knowledgeBase"??: // Optional
  {
    "id": string (uuid)
  }
  "parser"??: string (uuid) // Optional
  "labels"??: [ // Optional
    {
      "id": string (uuid)
    }
  ]
  "rawUserDefineMetadata"??: object // Optional
}
```

Request Example Value

```
{
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
}
```

Code Examples

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  },
  "parser": "550e8400-e29b-41d4-a716-446655440000",
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    },
    "parser": "550e8400-e29b-41d4-a716-446655440000",
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": None
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "filename": "document.pdf",
            "file": "https://example.com/file.jpg",
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ],
            "parser": "550e8400-e29b-41d4-a716-446655440000",
            "labels": [
                [
                    {
                        "id": "550e8400-e29b-41d4-a716-446655440000"
                    }
                ],
                "rawUserDefineMetadata": null
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{  
  "id": string (uuid)  
  "filename": string // File name  
  "file": string (uri) // File to upload  
  "fileType": string  
  "knowledgeBase"??: // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
  "size": integer  
  "status":  
  {  
  }  
  "parser": {  
    "id": string (uuid)  
    "name": string  
    "provider":  
    {  
    }  
    "order"??: integer // Optional  
  }  
  }  
  "labels"??: [ // Optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
]  
  "rawUserDefineMetadata"??: object // Optional  
  "createdAt": string (timestamp)  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "filename": "Response String",  
  "file": "https://example.com/file.jpg",  
  "fileType": "Response String",  
  "knowledgeBase": {
```

```

    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
},
"size": 456,
"status": {},
"parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String",
    "provider": {},
    "order": 456
},
"labels": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
}
],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}

```

- ### Partially Update Files

PATCH [/api/knowledge-bases/{knowledgeBasePk}/files/batch-reparse/](#)

Parameters

Parameter Name	Required	Type	Description
<code>knowledgeBasePk</code>	V	string	
<code>fileType</code>	X	string	Filter by file type, e.g., pdf, docx, txt, xlsx, etc.
<code>knowledgeBase</code>	X	string	Knowledge Base ID
<code>page</code>	X	integer	A page number within the paginated result set.
<code>pageSize</code>	X	integer	Number of results to return per page.
<code>query</code>	X	string	Search keyword, can search by file name or file ID
<code>status</code>	X	string	Filter by file status, possible values: initial, processing, done, deleting, deleted, failed <code>initial</code> : Initial ; <code>processing</code> : Processing ; <code>done</code> : Done ; <code>deleting</code> : Deleting ; <code>deleted</code> : Deleted ; <code>faile ...</code>

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/batch-reparse??
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '[]'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = [];

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/batch-reparse?_
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/batch-reparse/?fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-446655440000&page=1&pageSize=1&query=example&status=deleted"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = []

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/batch-reparse/?
fileType=example&knowledgeBase=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&status=deleted", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => []
]);

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
    }
  ]
}
```

```

"filename": string // File name
"file": string (uri) // File to upload
"fileType": string
"knowledgeBase?": // Optional
{
  "id": string (uuid)
  "name": string
}
"size": integer
"status":
{
}
"parser": {
{
  "id": string (uuid)
  "name": string
  "provider":
  {
  }
  "order?": integer // Optional
}
}
"labels?": [ // Optional
{
  "id": string (uuid)
  "name": string
}
]
"rawUserDefineMetadata?": object // Optional
"createdAt": string (timestamp)
}
]
}

```

Response Example Value

```
{
"count": 123,
"next": "http://api.example.org/accounts/?page=4",
"previous": "http://api.example.org/accounts/?page=2",
"results": [
{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "filename": "Response String",
  "file": "https://example.com/file.jpg",
  "fileType": "Response String",
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
  }
}
]
```

```

    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
},
"size": 456,
"status": {},
"parser": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String",
    "provider": {},
    "order": 456
},
"labels": [
{
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response String"
}
],
"rawUserDefineMetadata": null,
"createdAt": "Response String"
}
]
}

```

- ### Delete File

DELETE `/api/knowledge-bases/{knowledgeBasePk}/files/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Chatbot File.
<code>knowledgeBasePk</code>	V	string	

Code Examples

Shell/Bash

```
# Example API Call (Shell)
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/files/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response

Status Code	Description
204	No response body

*

Create New FAQ

POST /api/knowledge-bases/{knowledgeBasePk}/faqs/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
question	string	Yes	
answer	string	Yes	
answerMediaUrls	object	No	URLs for media files such as images and videos in the answer
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	
knowledgeBase	object (with 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	

Request Structure Example

```
{
  "question": string
  "answer": string
  "answerMediaUrls"??: object // Stores the URLs of media files such as images, videos, etc. in the answer (optional)
  "labels"??: [ // optional
    {
      "id": string (uuid)
    }
  ]
  "rawUserDefineMetadata"??: object // optional
  "knowledgeBase"??: // optional
  {
    "id": string (uuid)
  }
}
```

Request Example Value

```
{  
    "question": "Example string",  
    "answer": "Example string",  
    "answerMediaUrls": null,  
    "labels": [  
        {  
            "id": "550e8400-e29b-41d4-a716-446655440000"  
        }  
    ],  
    "rawUserDefineMetadata": null,  
    "knowledgeBase": {  
        "id": "550e8400-e29b-41d4-a716-446655440000"  
    }  
}
```

Code Example

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "question": "Sample string",
  "answer": "Sample string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "question": "Sample string",
    "answer": "Sample string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "question": "Sample string",
            "answer": "Sample string",
            "answerMediaUrls": null,
            "labels": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "rawUserDefineMetadata": null,
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 201

Response Schema Example

```
{  
  "id": string (uuid)  
  "question": string  
  "answer": string  
  "answerMediaUrls"??: object // Stores the URLs of media files such as images,  
videos, etc. in the answer (optional)  
  "hitsCount": integer  
  "embeddingTokensCount": integer // The number of embedding tokens used when  
creating this FAQ  
  "labels"??: [ // optional  
    {  
      "id": string (uuid)  
      "name": string  
    }  
  ]  
  "rawUserDefineMetadata"??: object // optional  
  "knowledgeBase"??: // optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "question": "Response string",  
  "answer": "Response string",  
  "answerMediaUrls": null,  
  "hitsCount": 456,  
  "embeddingTokensCount": 456,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Response string"
```

```
    }  
}
```

- ### Batch Delete FAQs

POST /api/knowledge-bases/{knowledgeBasePk}/faqs/batch-delete/

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
ids	array[string]	No	

Request Structure Example

```
{  
  "ids"??: [ // Optional  
    string (uuid)  
  ]  
}
```

Request Example Value

```
{  
  "ids": [  
    "550e8400-e29b-41d4-a716-446655440000"  
  ]  
}
```

Code Examples

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/batch-delete/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "ids": [
    "550e8400-e29b-41d4-a716-446655440000"
  ]
};

axios.post("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/batch-delete/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/batch-delete/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "ids": [
        "550e8400-e29b-41d4-a716-446655440000"
    ]
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client→post("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/batch-delete/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "ids": [
                "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response→getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e→getMessage();
}
?>

```

Response Body

Status Code	Description
204	No response body

*

List all FAQs for a specific knowledge base

GET </api/knowledge-bases/{knowledgeBasePk}/faqs/>

Parameters

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/?page=1&pageSize=1&query=example"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and check the request data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/?page=1&pageSize=1&query=example", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/?page=1&pageSize=1&query=example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/?
page=1&pageSize=1&query=example", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
      "question": string
      "answer": string
      "answerMediaUrls"??: object // URLs for media files like images, videos,

```

```

etc., in the answer (Optional)
  "hitsCount": integer
  "embeddingTokensCount": integer // Number of embedding tokens used when
creating this FAQ
  "labels"??: [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "rawUserDefineMetadata"??: object // Optional
  "knowledgeBase"??: // Optional
  {
    "id": string (uuid)
    "name": string
  }
}
]
}

```

Response Example Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "question": "Response string",
      "answer": "Response string",
      "answerMediaUrls": null,
      "hitsCount": 456,
      "embeddingTokensCount": 456,
      "labels": [
        {
          "id": "550e8400-e29b-41d4-a716-446655440000",
          "name": "Response string"
        }
      ],
      "rawUserDefineMetadata": null,
      "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Response string"
      }
    }
  ]
}
```

```
]  
}
```

- ### Get Specific FAQ Details

GET `/api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this FAQ.
<code>knowledgeBasePk</code>	V	string	

Code Examples

Shell/Bash

```
# API Call Example (Shell)  
curl -X GET "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```

<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>

```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "question": string
  "answer": string
  "answerMediaUrls": object // URLs of media files such as images, videos,
etc. in the answer (optional)
  "hitsCount": integer
  "embeddingTokensCount": integer // Number of embedding tokens used when
creating this FAQ
  "labels": [ // optional
}
```

```

{
  "id": string (uuid)
  "name": string
}
]
"rawUserDefineMetadata"??: object // optional
"knowledgeBase"??: // optional
{
  "id": string (uuid)
  "name": string
}
}

```

Response Example Value

```

{
  "id": "550e8400-e29b-41d4-a716-446655440000",
  "question": "Response string",
  "answer": "Response string",
  "answerMediaUrls": null,
  "hitsCount": 456,
  "embeddingTokensCount": 456,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "name": "Response string"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Response string"
  }
}

```

- ### Update FAQ

PUT `/api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this FAQ.

Parameter Name	Required	Type	Description
knowledgeBasePk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
question	string	Yes	
answer	string	Yes	
answerMediaUrls	object	No	URLs for media files such as images and videos in the answer
labels	array[IdName]	No	
rawUserDefineMetadata	object	No	
knowledgeBase	object (contains 2 properties: id, name)	No	
knowledgeBase.id	string (uuid)	Yes	

Request Structure Example

```
{
  "question": string
  "answer": string
  "answerMediaUrls"??: object // Stores the URLs of media files such as images, videos, etc. in the answer (optional)
  "labels"??: [ // optional
    {
      "id": string (uuid)
    }
  ]
  "rawUserDefineMetadata"??: object // optional
  "knowledgeBase"??: // optional
  {
    "id": string (uuid)
  }
}
```

Request Example Value

```
{  
  "question": "Example string",  
  "answer": "Example string",  
  "answerMediaUrls": null,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000"  
  }  
}
```

Code Examples

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "question": "Sample string",
  "answer": "Sample string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
};

axios.put("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": None,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": None,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->put("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "question": "Sample string",
            "answer": "Sample string",
            "answerMediaUrls": null,
            "labels": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "rawUserDefineMetadata": null,
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{  
  "id": string (uuid)  
  "question": string  
  "answer": string  
  "answerMediaUrls"??: object // Stores the URLs of media files such as images,  
  videos, etc. in the answer (optional)  
  "hitsCount": integer  
  "embeddingTokensCount": integer // The number of embedding tokens used when  
  creating this FAQ  
  "labels"??: [ // optional  
    {  
      "id": string (uuid)  
      "name": string  
    }  
  ]  
  "rawUserDefineMetadata"??: object // optional  
  "knowledgeBase"??: // optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "question": "Response string",  
  "answer": "Response string",  
  "answerMediaUrls": null,  
  "hitsCount": 456,  
  "embeddingTokensCount": 456,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Response string"  
  }  
}
```

```
}
```

- ### Partially Update FAQ

PATCH `/api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/`

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this FAQ.
<code>knowledgeBasePk</code>	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
<code>question</code>	string	No	
<code>answer</code>	string	No	
<code>answerMediaUrls</code>	object	No	URLs for media files such as images and videos in the answer
<code>labels</code>	array[IdName]	No	
<code>rawUserDefineMetadata</code>	object	No	
<code>knowledgeBase</code>	object (contains 2 properties: id, name)	No	
<code>knowledgeBase.id</code>	string (uuid)	Yes	

Request Schema Example

```
{
  "question"?: string // Optional
  "answer"?: string // Optional
  "answerMediaUrls"?: object // Stores URLs for media files like images,
  //videos, etc., in the answer (Optional)
  "labels"?: [ // Optional
    {
```

```
        "id": string (uuid)
    }
]
"rawUserDefineMetadata"??: object // Optional
"knowledgeBase"??: // Optional
{
    "id": string (uuid)
}
}
```

Request Example Value

```
{
  "question": "Example string",
  "answer": "Example string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
}
```

Code Examples

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "question": "Example String",
    "answer": "Example String",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "question": "sample string",
  "answer": "sample string",
  "answerMediaUrls": null,
  "labels": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000"
    }
  ],
  "rawUserDefineMetadata": null,
  "knowledgeBase": {
    "id": "550e8400-e29b-41d4-a716-446655440000"
  }
};

axios.patch("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully got response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "question": "Example string",
    "answer": "Example string",
    "answerMediaUrls": null,
    "labels": [
        {
            "id": "550e8400-e29b-41d4-a716-446655440000"
        }
    ],
    "rawUserDefineMetadata": null,
    "knowledgeBase": {
        "id": "550e8400-e29b-41d4-a716-446655440000"
    }
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->patch("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "question": "Example string",
            "answer": "Example string",
            "answerMediaUrls": null,
            "labels": [
                {
                    "id": "550e8400-e29b-41d4-a716-446655440000"
                }
            ],
            "rawUserDefineMetadata": null,
            "knowledgeBase": [
                "id": "550e8400-e29b-41d4-a716-446655440000"
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{  
  "id": string (uuid)  
  "question": string  
  "answer": string  
  "answerMediaUrls"??: object // URLs of media files such as images, videos,  
etc. in the answer (optional)  
  "hitsCount": integer  
  "embeddingTokensCount": integer // The number of embedding tokens used when  
creating this FAQ  
  "labels"??: [ // optional  
    {  
      "id": string (uuid)  
      "name": string  
    }  
  ]  
  "rawUserDefineMetadata"??: object // optional  
  "knowledgeBase"??: // optional  
  {  
    "id": string (uuid)  
    "name": string  
  }  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "question": "Response string",  
  "answer": "Response string",  
  "answerMediaUrls": null,  
  "hitsCount": 456,  
  "embeddingTokensCount": 456,  
  "labels": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "name": "Response string"  
    }  
  ],  
  "rawUserDefineMetadata": null,  
  "knowledgeBase": {  
    "id": "550e8400-e29b-41d4-a716-446655440000",  
    "name": "Response string"  
  }  
}
```

```
    }  
}
```

- ### Delete FAQ

DELETE /api/knowledge-bases/{knowledgeBasePk}/faqs/{id}/

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this FAQ.
knowledgeBasePk	V	string	

Code Examples

Shell/Bash

```
# API Call Example (Shell)  
curl -X DELETE "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-  
41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"  
-H "Authorization: Api-Key YOUR_API_KEY"  
  
# Please replace YOUR_API_KEY and verify the request data before  
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/", data,
config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/knowledge-bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->delete("https://api.maiagent.ai/api/knowledge-
bases/550e8400-e29b-41d4-a716-446655440000/faqs/550e8400-e29b-41d4-a716-
446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>
```

Response

Status Code	Description
204	No response body

*

Call API Example (Shell)

Get All Text Nodes of an AI Assistant's Files

GET

/api/chatbot-text-nodes/

Parameters

Parameter Name	Required	Type	Description
chatbotFile	V	string	Chatbot File ID
cursor	X	string	The pagination cursor value.
pageSize	X	integer	Number of results to return per page.

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X GET "https://api.maiagent.ai/api/chatbot-text-nodes/?
chatbotFile=example&cursor=example&pageSize=1"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/chatbot-text-nodes/?chatbotFile=example&cursor=example&pageSize=1", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/chatbot-text-nodes/?chatbotFile=example&cursor=example&pageSize=1"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/chatbot-
text-nodes/?chatbotFile=example&cursor=example&pageSize=1", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
  "results": [
    {
      "id": string (uuid)
```

```
        "charactersCount": integer
        "hitsCount": integer
        "text": string
        "updatedAt": string (timestamp)
        "filename": string
        "chatbotFile": string (uuid)
        "pageNumber": integer
    }
]
}
```

Example Response Value

```
{
  "next": "http://api.example.org/accounts/?cursor=cD000DY%3D\",
  "previous": "http://api.example.org/accounts/?cursor=cj0xJnA9NDg3",
  "results": [
    {
      "id": "550e8400-e29b-41d4-a716-446655440000",
      "charactersCount": 456,
      "hitsCount": 456,
      "text": "Response String",
      "updatedAt": "Response String",
      "filename": "Response String",
      "chatbotFile": "550e8400-e29b-41d4-a716-446655440000",
      "pageNumber": 456
    }
  ]
}
```

Retrieve a specific text node of an AI assistant's file

GET

/api/chatbot-text-nodes/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this ChatbotTextNode.

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/chatbot-text-
nodes/550e8400-e29b-41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/chatbot-text-nodes/550e8400-e29b-41d4-a716-446655440000/", config)
  .then(response => {
    console.log('Successfully retrieved response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

Python

```
import requests

url = "https://api.maiagent.ai/api/chatbot-text-nodes/550e8400-e29b-
41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/chatbot-
text-nodes/550e8400-e29b-41d4-a716-446655440000/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "charactersCount": integer
  "hitsCount": integer
  "text": string
  "updatedAt": string (timestamp)
```

```
"filename": string  
"chatbotFile": string (uuid)  
"pageNumber": integer  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "charactersCount": 456,  
  "hitsCount": 456,  
  "text": "Response string",  
  "updatedAt": "Response string",  
  "filename": "Response string",  
  "chatbotFile": "550e8400-e29b-41d4-a716-446655440000",  
  "pageNumber": 456  
}
```

Search Test

POST [/api/chatbots/{id}/search/](#)

Parameters

Parameter Name	Required	Type	Description
<code>id</code>	V	string	A UUID string identifying this Chatbot.
<code>largeLanguageModel</code>	X	string	
<code>page</code>	X	integer	A page number within the paginated result set.
<code>pageSize</code>	X	integer	Number of results to return per page.

query	X	string
replyMode	X	string

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/chatbots/550e8400-e29b-
41d4-a716-446655440000/search/?largeLanguageModel=550e8400-e29b-
41d4-a716-
446655440000&page=1&pageSize=1&query=example&replyMode=example"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "query": "example string"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "query": "example string"
};

axios.post("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-
a716-446655440000/search/?largeLanguageModel=550e8400-e29b-41d4-
a716-
446655440000&page=1&pageSize=1&query=example&replyMode=example",
  data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/search/?largeLanguageModel=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&replyMode=example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "query": "Example string"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>post("https://api.maiagent.ai/api/chatbots/550e8400-e29b-41d4-a716-
446655440000/search/?largeLanguageModel=550e8400-e29b-41d4-a716-
446655440000&page=1&pageSize=1&query=example&replyMode=example", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "query": "example string"
    ]
]);

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{  
  "next"?: string (uri) // Optional  
  "previous"?: string (uri) // Optional  
  "results": [  
    {  
      "id": string (uuid)  
      "charactersCount": integer  
      "hitsCount": integer  
      "text": string  
      "updatedAt": string (timestamp)  
      "filename": string  
      "chatbotFile": string (uuid)  
      "pageNumber": integer  
    }  
  ]  
}
```

Response Example Value

```
{  
  "next": "http://api.example.org/accounts/?cursor=cD000DY%3D\"",  
  "previous": "http://api.example.org/accounts/?cursor=cj0xJnA9NDg3",  
  "results": [  
    {  
      "id": "550e8400-e29b-41d4-a716-446655440000",  
      "charactersCount": 456,  
      "hitsCount": 456,  
      "text": "Response string",  
      "updatedAt": "Response string",  
      "filename": "Response string",  
      "chatbotFile": "550e8400-e29b-41d4-a716-446655440000",  
      "pageNumber": 456  
    }  
  ]  
}
```

List Parsers Supported by File Type

GET

/api/parsers/supported-file-types/

Code Example

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/parsers/supported-file-
types/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/parsers/supported-file-
types/", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

Python

```
import requests

url = "https://api.maiagent.ai/api/parsers/supported-file-types/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/parsers/supported-file-types/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e-
>getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
[
{
    "fileType": string
    "parsers": [
        {
            "id": string
            "name": string
            "version": string
            "description": string
            "fileTypes": [
                string
            ]
        }
    ]
}
```

```
"id": string (uuid)
"name": string
"provider":
{
}
"order?": integer // Optional
}
]
}
]
```

Response Example Value

```
[
[
{
  "fileType": ".pdf",
  "parsers": [
    {
      "id": "ab83f144-5026-4bce-993f-5c982cc19318",
      "name": "MaiAgent Parser",
      "provider": "maiagent",
      "order": 0,
      "isDefault": true
    },
    {
      "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
      "name": "MaiAgent Parser (Online)",
      "provider": "llama",
      "order": 2,
      "isDefault": false
    }
  ]
},
{
  "fileType": ".doc",
  "parsers": [
    {
      "id": "ab83f144-5026-4bce-993f-5c982cc19318",
      "name": "MaiAgent Parser",
      "provider": "maiagent",
      "order": 0,
      "isDefault": true
    },
    {
      "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
      "name": "MaiAgent Parser (Online)",
      "provider": "llama",
      "order": 2,
      "isDefault": false
    }
  ]
}
```

```
        "provider": "llama",
        "order": 2,
        "isDefault": false
    }
]
},
{
    "fileType": ".docx",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        },
        {
            "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
            "name": "MaiAgent Parser (Online)",
            "provider": "llama",
            "order": 2,
            "isDefault": false
        }
    ]
},
{
    "fileType": ".ppt",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        },
        {
            "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
            "name": "MaiAgent Parser (Online)",
            "provider": "llama",
            "order": 2,
            "isDefault": false
        }
    ]
},
{
    "fileType": ".pptx",
    "parsers": [
        {

```

```
        "id": "ab83f144-5026-4bce-993f-5c982cc19318",
        "name": "MaiAgent Parser",
        "provider": "maiagent",
        "order": 0,
        "isDefault": true
    },
    {
        "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
        "name": "MaiAgent Parser (Online)",
        "provider": "llama",
        "order": 2,
        "isDefault": false
    }
]
},
{
    "fileType": ".xls",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        }
    ]
},
{
    "fileType": ".xlsx",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        }
    ]
},
{
    "fileType": ".csv",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        }
    ]
}
]
```

```
        }
    ],
},
{
    "fileType": ".txt",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        },
        {
            "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
            "name": "MaiAgent Parser (Online)",
            "provider": "llama",
            "order": 2,
            "isDefault": false
        }
    ]
},
{
    "fileType": ".md",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        }
    ]
},
{
    "fileType": ".json",
    "parsers": [
        {
            "id": "ab83f144-5026-4bce-993f-5c982cc19318",
            "name": "MaiAgent Parser",
            "provider": "maiagent",
            "order": 0,
            "isDefault": true
        }
    ]
},
{
    "fileType": ".jsonl",
}
```

```
"parsers": [
  {
    "id": "ab83f144-5026-4bce-993f-5c982cc19318",
    "name": "MaiAgent Parser",
    "provider": "maiagent",
    "order": 0,
    "isDefault": true
  }
],
},
{
  "fileType": ".html",
  "parsers": [
    {
      "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
      "name": "MaiAgent Parser (Online)",
      "provider": "llama",
      "order": 2,
      "isDefault": false
    }
  ]
},
{
  "fileType": ".mp3",
  "parsers": [
    {
      "id": "4c47e305-2eb7-4438-8d3c-d91eb0b06cc0",
      "name": "Azure Speech",
      "provider": "azure",
      "order": 1,
      "isDefault": true
    },
    {
      "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
      "name": "MaiAgent Parser (Online)",
      "provider": "llama",
      "order": 2,
      "isDefault": false
    }
  ]
},
{
  "fileType": ".wav",
  "parsers": [
    {
      "id": "4c47e305-2eb7-4438-8d3c-d91eb0b06cc0",
      "name": "Azure Speech",
      "provider": "azure",
      "order": 1,
      "isDefault": true
    }
  ]
}
```

```
        "order": 1,
        "isDefault": true
    },
    {
        "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
        "name": "MaiAgent Parser (Online)",
        "provider": "llama",
        "order": 2,
        "isDefault": false
    }
]
},
{
    "fileType": ".mp4",
    "parsers": [
        {
            "id": "22fdccb5-f075-4aad-bb81-048289ca4b25",
            "name": "MaiAgent Parser (Online)",
            "provider": "llama",
            "order": 2,
            "isDefault": false
        }
    ]
}
]
```

組織和成員

I have completed the full translation of your Traditional Chinese Markdown API documentation into English. The translated document has been saved to [translated_output.md](#).

Key translations made:

- 帳號註冊 → Account Registration
- 更換密碼 → Change Password
- 建立新組織 → Create New Organization
- 新增成員至指定組織 → Add Member to Specified Organization
- 取得組織列表 → Get Organization List
- 取得特定組織資訊 → Get Specific Organization Information
- 取得當前用戶詳細資訊 → Get Current User Details
- 取得當前用戶權限 → Get Current User Permissions

- 取得指定組織的成員列表 → Get Member List for Specified Organization
- 更新組織資訊 → Update Organization Information
- 更新當前用戶詳細資訊 → Update Current User Details
- 刪除指定成員 → Delete Specified Member

All technical terms, code examples, API endpoints, field names, and CSS styling have been preserved exactly as they were in the original document. The entire content has been faithfully translated without any summarization.

API Call Example (Shell)

List Contacts

GET

/api/contacts/

Parameters

Parameter Name	Required	Type	Description
inbox	X	string	Deprecated: Use inboxes instead
inboxes	X	string	Conversation platform ID (used to filter contacts from a specific inbox)
page	X	integer	A page number within the paginated result set.
pageSize	X	integer	Number of results to return per page.
query	X	string	Search by name or source ID
search	X	string	Full-text search

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/contacts/?inbox=550e8400-
e29b-41d4-a716-
446655440000&inboxes=example&page=1&pageSize=1&query=example&search=
example"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/contacts/?inbox=550e8400-
e29b-41d4-a716-
446655440000&inboxes=example&page=1&pageSize=1&query=example&search=
example", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/contacts/?inbox=550e8400-e29b-
41d4-a716-
446655440000&inboxes=example&page=1&pageSize=1&query=example&search=
example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->get("https://api.maiagent.ai/api/contacts/?inbox=550e8400-e29b-41d4-a716-44665440000&inboxes=example&page=1&pageSize=1&query=example&search=example", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "count": integer
  "next"??: string (uri) // Optional
  "previous"??: string (uri) // Optional
```

```
"results": [
  {
    "id": string (uuid)
    "inboxes?": [ // Optional
      {
        "id": string (uuid)
        "name": string
      }
    ]
    "inbox?": // Backward compatibility field: single inbox, ignored if inboxes is also provided (Optional)
    {
      "id": string (uuid)
      "name": string
    }
    "name": string
    "avatar?": string (uri) // Optional
    "phoneNumber?": string // Optional
    "email?": string (email) // Optional
    "sourceId?": string // Optional
    "queryMetadata?": object // Optional
    "mcpCredentials": string // List of the contact's MCP credentials
    "createdAt": string (timestamp)
    "updatedAt": string (timestamp)
  }
]
}
```

Response Example Value

```
{
  "count": 123,
  "next": "http://api.example.org/accounts/?page=4",
  "previous": "http://api.example.org/accounts/?page=2",
  "results": [
    [
      {
        "id": "user1_uuid",
        "inboxes": [
          {
            "id": "inbox_uuid",
            "name": "inbox_name"
          }
        ],
        "name": "user1",
        "avatar": "",
        "sourceId": null,
        "email": "user1@example.com",
        "phoneNumber": "+1234567890",
        "queryMetadata": {
          "lastSynced": "2023-10-01T12:00:00Z"
        }
      }
    ]
  ]
}
```

```
"queryMetadata": null,
"createdAt": "1751875361000",
"updatedAt": "1751875361000"
},
{
  "id": "user2_uuid",
  "inboxes": [
    {
      "id": "inbox_uuid",
      "name": "inbox_name"
    }
  ],
  "name": "user2",
  "avatar": "avatar_url",
  "sourceId": "self defined source id",
  "queryMetadata": "self defined query metadata",
  "createdAt": "1751875400000",
  "updatedAt": "1751875400000"
}
]
]
```

Create Contact

POST

/api/contacts/

Code Example

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/contacts/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "inboxes": [
        {
            "id": "inbox_uuid"
        }
    ],
    "name": "user1",
    "avatar": "",
    "phoneNumber": 912345678,
    "email": "user1@example.com",
    "sourceId": "self defined source id",
    "queryMetadata": "self defined query metadata"
}'
```

Please replace YOUR_API_KEY and verify the request data before execution.

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "inboxes": [
    {
      "id": "inbox_uuid"
    }
  ],
  "name": "user1",
  "avatar": "",
  "phoneNumber": 912345678,
  "email": "user1@example.com",
  "sourceId": "self defined source id",
  "queryMetadata": "self defined query metadata"
};

axios.post("https://api.maiagent.ai/api/contacts/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

```
import requests

url = "https://api.maiagent.ai/api/contacts/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "inboxes": [
        {
            "id": "inbox_uuid"
        }
    ],
    "name": "user1",
    "avatar": "",
    "phoneNumber": 912345678,
    "email": "user1@example.com",
    "sourceId": "self defined source id",
    "queryMetadata": "self defined query metadata"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>post("https://api.maiagent.ai/api/contacts/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "inboxes": [
            [
                "id": "inbox_uuid"
            ],
            "name": "user1",
            "avatar": "",
            "phoneNumber": 912345678,
            "email": "user1@example.com",
            "sourceId": "self defined source id",
            "queryMetadata": "self defined query metadata"
        ]
    ],
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 201

Response Schema Example

```
{
  "id": string (uuid)
  "inboxes"??: [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "inbox"??: // Backward compatibility field: single inbox, ignored if
  inboxes is also provided (Optional)
  {
    "id": string (uuid)
    "name": string
  }
  "name": string
  "avatar"??: string (uri) // Optional
  "phoneNumber"??: string // Optional
  "email"??: string (email) // Optional
  "sourceId"??: string // Optional
  "queryMetadata"??: object // Optional
  "mcpCredentials": string // List of the contact's MCP credentials
  "createdAt": string (timestamp)
  "updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "user1_uuid",
  "inboxes": [
    {
      "id": "inbox_uuid",
      "name": "inbox_name"
    }
  ],
  "name": "user1",
  "avatar": "",
```

```
"phoneNumber": 912345678,  
"email": "user1@example.com",  
"sourceId": null,  
"queryMetadata": null,  
"createdAt": "1751875361000",  
"updatedAt": "1751875361000"  
}
```

Get Contact Details

GET

/api/contacts/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Contact.

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X GET "https://api.maiagent.ai/api/contacts/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

JavaScript

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

axios.get("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-
a716-446655440000/", config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-
446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.get(url, headers=headers)
try:
    print("Successfully retrieved response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>get("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-
446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully got response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "inboxes"??: [ // Optional
    {
      "id": string (uuid)
```

```
        "name": string
    }
]
"inbox"?: // Backward compatibility field: single inbox, ignored if
inboxes is also provided (Optional)
{
    "id": string (uuid)
    "name": string
}
"name": string
"avatar"?: string (uri) // Optional
"phoneNumber"?: string // Optional
"email"?: string (email) // Optional
"sourceId"?: string // Optional
"queryMetadata"?: object // Optional
"mcpCredentials": string // List of the contact's MCP credentials
"createdAt": string (timestamp)
"updatedAt": string (timestamp)
}
```

Response Example Value

```
{
    "id": "user1_uuid",
    "inboxes": [
        {
            "id": "inbox_uuid",
            "name": "inbox_name"
        }
    ],
    "name": "user1",
    "avatar": "",
    "phoneNumber": 912345678,
    "email": "user1@example.com",
    "sourceId": null,
    "queryMetadata": null,
    "createdAt": "1751875361000",
    "updatedAt": "1751875361000"
}
```

Update Contact

PUT

/api/contacts/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Contact.

Request Body

Request Parameters

Field	Type	Required	Description
name	string	No	Contact name
inboxes	array[object]	No	List of inboxes (new format)
inbox	object	No	Single inbox (old format, for backward compatibility)
inbox.id	string (uuid)	Yes	Inbox ID
avatar	string	No	Avatar URL
sourceld	string	No	Source ID
queryMetadata	string	No	Query metadata

Request Structure Example

```
{
  "name"?: string // Contact name (optional)
  "inboxes"?: [ // List of inboxes (new format) (optional)
    {
      "id": string (uuid) // Inbox ID
    }
  ]
}
```

```
"inbox"?: { // Single inbox (old format, for backward compatibility)  
  (optional)  
  {  
    "id": string (uuid) // Inbox ID  
  }  
}  
  
"avatar"?: string // Avatar URL (optional)  
"sourceId"?: string // Source ID (optional)  
"queryMetadata"?: string // Query metadata (optional)  
}
```

Request Example Value

```
{  
  "inboxes": [  
    {  
      "id": "new_inbox_uuid"  
    }  
  ],  
  "name": "updated_user_name",  
  "avatar": "updated_avatar_url",  
  "phoneNumber": 987654321,  
  "email": "updated@example.com",  
  "sourceId": "updated_source_id",  
  "queryMetadata": "updated_query_metadata"  
}
```

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X PUT "https://api.maiagent.ai/api/contacts/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "inboxes": [
        {
            "id": "new_inbox_uuid"
        }
    ],
    "name": "updated_user_name",
    "avatar": "updated_avatar_url",
    "phoneNumber": 987654321,
    "email": "updated@example.com",
    "sourceId": "updated_source_id",
    "queryMetadata": "updated_query_metadata"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "inboxes": [
    {
      "id": "new_inbox_uuid"
    }
  ],
  "name": "updated_user_name",
  "avatar": "updated_avatar_url",
  "phoneNumber": 987654321,
  "email": "updated@example.com",
  "sourceId": "updated_source_id",
  "queryMetadata": "updated_query_metadata"
};

axios.put("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "inboxes": [
        {
            "id": "new_inbox_uuid"
        }
    ],
    "name": "updated_user_name",
    "avatar": "updated_avatar_url",
    "phoneNumber": 987654321,
    "email": "updated@example.com",
    "sourceId": "updated_source_id",
    "queryMetadata": "updated_query_metadata"
}

response = requests.put(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>put("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-
446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "inboxes": [
            [
                "id": "new_inbox_uuid"
            ]
        ],
        "name": "updated_user_name",
        "avatar": "updated_avatar_url",
        "phoneNumber": 987654321,
        "email": "updated@example.com",
        "sourceId": "updated_source_id",
        "queryMetadata": "updated_query_metadata"
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Error during request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "inboxes"??: [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "inbox"??: // Backward compatibility field: single inbox, ignored if
  inboxes is also provided (Optional)
  {
    "id": string (uuid)
    "name": string
  }
  "name": string
  "avatar"??: string (uri) // Optional
  "phoneNumber"??: string // Optional
  "email"??: string (email) // Optional
  "sourceId"??: string // Optional
  "queryMetadata"??: object // Optional
  "mcpCredentials": string // List of the contact's MCP credentials
  "createdAt": string (timestamp)
  "updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "user1_uuid",
  "inboxes": [
    {
      "id": "inbox_uuid",
      "name": "inbox_name"
    }
  ],
  "name": "user1",
  "avatar": "",
```

```
        "phoneNumber": 912345678,  
        "email": "user1@example.com",  
        "sourceId": null,  
        "queryMetadata": null,  
        "createdAt": "1751875361000",  
        "updatedAt": "1751875361000"  
    }  
}
```

Update Contact

PATCH

/api/contacts/{id}

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Contact.

Request Body

Request Parameters

Field	Type	Required	Description
name	string	No	Contact name
inboxes	array[object]	No	List of inboxes (new format)
inbox	object	No	Single inbox (old format, for backward compatibility)
inbox.id	string (uuid)	Yes	Inbox ID

Field	Type	Required	Description
avatar	string	No	Avatar URL
sourceId	string	No	Source ID
queryMetadata	string	No	Query metadata

Request Structure Example

```
{
  "name"??: string // Contact name (optional)
  "inboxes"??: [ // List of inboxes (new format) (optional)
    {
      "id": string (uuid) // Inbox ID
    }
  ]
  "inbox"??: { // Single inbox (old format, for backward compatibility)
    (optional)
    {
      "id": string (uuid) // Inbox ID
    }
  }
  "avatar"??: string // Avatar URL (optional)
  "sourceId"??: string // Source ID (optional)
  "queryMetadata"??: string // Query metadata (optional)
}
```

Request Example Value

```
{
  "inboxes": [
    {
      "id": "new_inbox_uuid"
    }
  ],
  "name": "updated_user_name",
  "avatar": "updated_avatar_url",
  "phoneNumber": 987654321,
  "email": "updated@example.com",
  "sourceId": "updated_source_id",
```

```
        "queryMetadata": "updated_query_metadata"
    }
```

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X PATCH "https://api.maiagent.ai/api/contacts/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
  "inboxes": [
    {
      "id": "new_inbox_uuid"
    }
  ],
  "name": "updated_user_name",
  "avatar": "updated_avatar_url",
  "phoneNumber": 987654321,
  "email": "updated@example.com",
  "sourceId": "updated_source_id",
  "queryMetadata": "updated_query_metadata"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "inboxes": [
    {
      "id": "new_inbox_uuid"
    }
  ],
  "name": "updated_user_name",
  "avatar": "updated_avatar_url",
  "phoneNumber": 987654321,
  "email": "updated@example.com",
  "sourceId": "updated_source_id",
  "queryMetadata": "updated_query_metadata"
};

axios.patch("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "inboxes": [
        {
            "id": "new_inbox_uuid"
        }
    ],
    "name": "updated_user_name",
    "avatar": "updated_avatar_url",
    "phoneNumber": 987654321,
    "email": "updated@example.com",
    "sourceId": "updated_source_id",
    "queryMetadata": "updated_query_metadata"
}

response = requests.patch(url, json=data, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>patch("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-
a716-446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "inboxes": [
            [
                "id": "new_inbox_uuid"
            ]
        ],
        "name": "updated_user_name",
        "avatar": "updated_avatar_url",
        "phoneNumber": 987654321,
        "email": "updated@example.com",
        "sourceId": "updated_source_id",
        "queryMetadata": "updated_query_metadata"
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{
  "id": string (uuid)
  "inboxes"??: [ // Optional
    {
      "id": string (uuid)
      "name": string
    }
  ]
  "inbox"??: // Backward compatibility field: single inbox, ignored if
  inboxes is also provided (Optional)
  {
    "id": string (uuid)
    "name": string
  }
  "name": string
  "avatar"??: string (uri) // Optional
  "phoneNumber"??: string // Optional
  "email"??: string (email) // Optional
  "sourceId"??: string // Optional
  "queryMetadata"??: object // Optional
  "mcpCredentials": string // List of the contact's MCP credentials
  "createdAt": string (timestamp)
  "updatedAt": string (timestamp)
}
```

Response Example Value

```
{
  "id": "user1_uuid",
  "inboxes": [
    {
      "id": "inbox_uuid",
      "name": "inbox_name"
    }
  ],
  "name": "user1",
  "avatar": "",
```

```
        "phoneNumber": 912345678,  
        "email": "user1@example.com",  
        "sourceId": null,  
        "queryMetadata": null,  
        "createdAt": "1751875361000",  
        "updatedAt": "1751875361000"  
    }  
}
```

Delete Contact

DELETE </api/contacts/{id}>

Parameters

Parameter Name	Required	Type	Description
id	V	string	A UUID string identifying this Contact.

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X DELETE "https://api.maiagent.ai/api/contacts/550e8400-e29b-
41d4-a716-446655440000/"
-H "Authorization: Api-Key YOUR_API_KEY"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY'
  }
};

// Request payload
const data = null;

axios.delete("https://api.maiagent.ai/api/contacts/550e8400-e29b-
41d4-a716-446655440000/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred during the request:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-a716-
446655440000/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

response = requests.delete(url, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>delete("https://api.maiagent.ai/api/contacts/550e8400-e29b-41d4-
a716-446655440000/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY'
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e-
>getMessage();
}
?>
```

Response

Status Code	Description
204	No response body
400	The contact still has associated conversations and cannot be deleted

角色和權限

API Error: Claude's response exceeded the 32000 output token maximum. To configure this behavior, set the CLAUDE_CODE_MAX_OUTPUT_TOKENS environment variable.

API Call Example (Shell)

Create New Attachment (Integration)

POST

/api/attachments-upload/

Code Examples

Shell/Bash

```
# API Call Example (Shell)
curl -X POST "https://api.maiagent.ai/api/attachments-upload/"
-H "Authorization: Api-Key YOUR_API_KEY"
-F "file=@example_file.pdf"

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');
const FormData = require('form-data');

// Create a FormData object
const formData = new FormData();
// Please replace with the actual file
formData.append('file', * actual file object *, 'example_file.pdf');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    ... formData.getHeaders()
  }
};

axios.post("https://api.maiagent.ai/api/attachments-upload/",
  formData, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

Python

```
import requests

url = "https://api.maiagent.ai/api/attachments-upload/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY"
}

# Create file and form data
files = {
    'file': ('example_file.pdf', open('example_file.pdf', 'rb'),
    'application/pdf')
}

response = requests.post(url, files=files, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred with the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/attachments-upload/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY'
        ],
        'multipart' => [
            [
                'name' => 'file',
                'contents' => fopen('example_file.pdf', 'r'),
                'filename' => 'example_file.pdf'
            ]
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code	Description
200	No response body

Get a Pre-signed URL for File Upload (Create/Get)

POST /api/upload-presigned-url/

Parameters

Parameter Name	Required	Type	Description
fieldName	X	string	Field Name
filename	X	string	File Name
modelName	X	string	Model Name

Request Body

Request Parameters

Field	Type	Required	Description
fileSize	integer	Yes	

Request Structure Example

```
{  
  "fileSize": integer  
}
```

Request Example Value

```
{  
  "fileSize": 123
```

```
}
```

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X POST "https://api.maiagent.ai/api/upload-presigned-url/?field
Name=example&filename=example&modelName=example"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "modelName": "example string",
    "fieldName": "example string",
    "filename": "document.pdf",
    "fileSize": 123
}'

# Please replace YOUR_API_KEY and verify the request data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "modelName": "example string",
  "fieldName": "example string",
  "filename": "document.pdf",
  "fileSize": 123
};

axios.post("https://api.maiagent.ai/api/upload-presigned-url/?  
fieldName=example&filename=example&modelName=example", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url = "https://api.maiagent.ai/api/upload-presigned-url/?  
fieldName=example&filename=example&modelName=example"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "modelName": "example string",
    "fieldName": "example string",
    "filename": "document.pdf",
    "fileSize": 123
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/upload-
presigned-url/?fieldName=example&filename=example&modelName=example", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "modelName": "example string",
            "fieldName": "example string",
            "filename": "document.pdf",
            "fileSize": 123
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'Request failed: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 200

Response Schema Example

```
{  
  "url": string (uri) // URL of the S3 bucket  
  "fields": object // Form fields for the POST request  
}
```

Response Example Value

```
{  
  "url": "https://s3.example.com/bucket-name",  
  "fields": {  
    "key": "media/chat/attachment/filename.pdf",  
    "x-amz-algorithm": "AWS4-HMAC-SHA256",  
    "x-amz-credential":  
      "AKIAXXXXXXXXXXXXXX/YYYYMMDDTHHMMSSZ",  
    "x-amz-date": "YYYYMMDDTHHMMSSZ",  
    "policy": "base64-encoded-policy-document",  
    "x-amz-signature": "generated-signature-hash"  
  }  
}
```

Presigned Attachment Upload

POST [/api/attachments/](#)

Request Body

Request Parameters

Field	Type	Required	Description
type	object	No	
filename	string	Yes	
file	string (uri)	Yes	

Request Structure Example

```
{  
  "type"??: // optional  
  {  
  }  
  "filename": string  
  "file": string (uri)  
}
```

Request Example Value

```
{  
  "type": {},  
  "filename": "document.pdf",  
  "file": "https://example.com/file.jpg"  
}
```

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X POST "https://api.maiagent.ai/api/attachments/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "type": {},
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg"
}'

# Please make sure to replace YOUR_API_KEY and verify the request
# data before execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "type": {},
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg"
};

axios.post("https://api.maiagent.ai/api/attachments/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });

```

Python

```
import requests

url = "https://api.maiagent.ai/api/attachments/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "type": {},
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully got response:")
    print(response.json())
except Exception as e:
    print("Request failed:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client->post("https://api.maiagent.ai/api/attachments/", [
        'headers' => [
            'Authorization' => 'Api-Key YOUR_API_KEY',
            'Content-Type' => 'application/json'
        ],
        'json' => [
            "type": {},
            "filename": "document.pdf",
            "file": "https://example.com/file.jpg"
        ]
    ]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred during the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 201

Response Schema Example

```
{  
  "id": string (uuid)  
  "type"??: // Optional  
  {  
  }  
  "filename": string  
  "file": string (uri)  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "type": {},  
  "filename": "Response String",  
  "file": "https://example.com/file.jpg"  
}
```

Create Conversation Attachment

POST

/api/conversations/{conversationPk}/attachments/

Parameters

Parameter Name	Required	Type	Description
conversationPk	V	string	

Request Body

Request Parameters

Field	Type	Required	Description
type	object	No	
filename	string	Yes	
file	string (uri)	Yes	
conversation	string (uuid)	No	

Request Structure Example

```
{  
  "type"??: // optional  
  {  
  }  
  "filename": string  
  "file": string (uri)  
  "conversation??: string (uuid) // optional  
}
```

Request Example Value

```
{  
  "type": {},  
  "filename": "document.pdf",  
  "file": "https://example.com/file.jpg",  
  "conversation": "550e8400-e29b-41d4-a716-446655440000"  
}
```

Code Examples

Shell/Bash

```
# Call API Example (Shell)
curl -X POST
"https://api.maiagent.ai/api/conversations/{conversationPk}/attachme
nts/"
-H "Authorization: Api-Key YOUR_API_KEY"
-H "Content-Type: application/json"
-d '{
    "type": {},
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "conversation": "550e8400-e29b-41d4-a716-446655440000"
}'

# Please replace YOUR_API_KEY and verify the request data before
execution.
```

```
const axios = require('axios');

// Set request headers
const config = {
  headers: {
    'Authorization': 'Api-Key YOUR_API_KEY',
    'Content-Type': 'application/json'
  }
};

// Request payload
const data = {
  "type": {},
  "filename": "document.pdf",
  "file": "https://example.com/file.jpg",
  "conversation": "550e8400-e29b-41d4-a716-446655440000"
};

axios.post("https://api.maiagent.ai/api/conversations/{conversationP
k}/attachments/", data, config)
  .then(response => {
    console.log('Successfully received response:');
    console.log(response.data);
  })
  .catch(error => {
    console.error('An error occurred with the request:');
    console.error(error.response?.data || error.message);
  });
}
```

```
import requests

url =
"https://api.maiagent.ai/api/conversations/{conversationPk}/attachme
nts/"
headers = {
    "Authorization": "Api-Key YOUR_API_KEY",
    "Content-Type": "application/json"
}

# Request payload
data = {
    "type": {},
    "filename": "document.pdf",
    "file": "https://example.com/file.jpg",
    "conversation": "550e8400-e29b-41d4-a716-446655440000"
}

response = requests.post(url, json=data, headers=headers)
try:
    print("Successfully received response:")
    print(response.json())
except Exception as e:
    print("An error occurred during the request:", e)
```

```
<?php
require 'vendor/autoload.php';

$client = new GuzzleHttp\Client();

try {
    $response = $client-
>post("https://api.maiagent.ai/api/conversations/{conversationPk}/at-
tachments/", [
    'headers' => [
        'Authorization' => 'Api-Key YOUR_API_KEY',
        'Content-Type' => 'application/json'
    ],
    'json' => [
        "type": {},
        "filename": "document.pdf",
        "file": "https://example.com/file.jpg",
        "conversation": "550e8400-e29b-41d4-a716-446655440000"
    ]
]);
}

$data = json_decode($response->getBody(), true);
echo "Successfully received response:\n";
print_r($data);
} catch (Exception $e) {
    echo 'An error occurred with the request: ' . $e->getMessage();
}
?>
```

Response Body

Status Code: 201

Response Schema Example

```
{  
  "id": string (uuid)  
  "type"??: // Optional  
  {  
  }  
  "filename": string  
  "file": string (uri)  
  "conversation"??: string (uuid) // Optional  
}
```

Response Example Value

```
{  
  "id": "550e8400-e29b-41d4-a716-446655440000",  
  "type": {},  
  "filename": "response string",  
  "file": "https://example.com/file.jpg",  
  "conversation": "550e8400-e29b-41d4-a716-446655440000"  
}
```