

API docs

• • •

目錄

1. Get All Text Nodes of an AI Assistant's Files
2. Retrieve a specific text node of an AI assistant's file
3. Search Test
4. List Parsers Supported by File Type
5. List Contacts
6. Create Contact
7. Get Contact Details
8. Update Contact
9. Update Contact
10. Delete Contact
11. Create New Attachment (Integration)
12. Get a Pre-signed URL for File Upload (Create/Get)
13. Presigned Attachment Upload
14. Create Conversation Attachment

根據您提供的文件內容,我發現這份文件已經是繁體中文了。內容包括:

1. 建立新 AI 助理 (POST `/api/chatbots/`)
2. 列出 Embedding Model (GET `/api/embedding-models/`)
3. 列出 Reranker Model (GET `/api/reranker-models/`)
4. 取得 AI 助理列表 (GET `/api/chatbots/`)
5. 取得特定 AI 助理 (GET `/api/chatbots/{id}/`)
6. 更新 AI 助理 (PUT `/api/chatbots/{id}/`)
7. 部分更新 AI 助理 (PATCH `/api/chatbots/{id}/`)
8. 刪除 AI 助理 (DELETE `/api/chatbots/{id}/`)

所有的介面說明、參數描述、程式碼範例都已經是繁體中文。

請問您希望我協助什麼?

1. 是否需要將此文件儲存到 `zh-tw` 資料夾?
2. 或是需要檢查翻譯品質並做優化?
3. 或是有其他特定的調整需求?

請告訴我您的需求,我會立即協助您處理!

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
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
id	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
<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 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
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)	No	
rerankerModel	string (uuid)	No	
name	string	No	
description	string	No	
numberOfRetrievedChunks	integer	No	Number of retrieved 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	
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) // 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
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 for 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> .

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 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

Shell/Bash

```
# 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);
  });
}
```

```
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();
}
?>
```

Response Body

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

翻譯完成！我已經將文件中的英文內容翻譯為繁體中文。主要翻譯的部分包括：

主要翻譯內容：

1. 參數說明表格：

- "A UUID string identifying this web chat" → "識別此 Web Chat 的 UUID 字串"

- "A UUID string identifying this Chatbot ID" → "識別此 Chatbot 的 UUID 字串"
 - "A UUID string identifying this Chatbot." → "識別此 Chatbot 的 UUID 字串"
1. 保持一致性：所有文檔中的標題、欄位名稱、狀態碼說明等都已經是繁體中文，維持原有格式和樣式。
 2. 程式碼範例：保持原有的程式碼不變，僅翻譯註釋和說明文字。

翻譯後的文件已儲存為 `translated_api_doc.md`。文件包含 4 個 API 端點的完整文檔，包括參數說明、請求回應範例、以及多種程式語言的程式碼範例（Shell/Bash、JavaScript、Python、PHP）。

對話和訊息

太好了！檔案已經完整存在並且內容完全正確。這個翻譯文件包含了：

1. 發送訊息 (串流) - POST `/api/chatbots/{chatbotId}/completions/`
2. 發送訊息 (建立) - POST `/api/messages/`
3. 建立新的對話 - POST `/api/conversations/`
4. 取得訊息列表 - GET `/api/messages/`
5. 取得特定訊息 - GET `/api/messages/{id}/`
6. 取得對話列表 - GET `/api/conversations/`
7. 取得特定對話 - GET `/api/conversations/{id}/`

每個 API 端點都包含了完整的：

- 參數說明
- 請求內容結構
- 程式碼範例（Shell/Bash, JavaScript, Python, PHP）
- 回應內容結構
- 完整的 CSS 樣式

檔案已經成功保存在：

`/home/runner/work/skills-for-Agents/skills-for-Agents/source_repo/zh-tw/api-reference/對話和訊息.md`

翻譯工作已完成！這份文件現在可以在 GitBook 的繁體中文版本中正常顯示了。

分析結果

我已經收到了這個大型 Markdown 文件的前半部分。這是一份 API 文件，包含了知識庫（Knowledge Bases）相關的 API 端點說明。文件已經是繁體中文（zh-tw），內容包括：

1. 建立知識庫 (POST `/api/knowledge-bases/`)
2. 列出所有知識庫 (GET `/api/knowledge-bases/`)

3. 取得特定知識庫詳情 (GET /api/knowledge-bases/{id})
)
4. 更新知識庫 (PUT /api/knowledge-bases/{id})
)
5. 部分更新知識庫 (PATCH /api/knowledge-bases/{id})
)
6. 刪除知識庫 (DELETE /api/knowledge-bases/{id})
)
7. 創建新標籤 (POST /api/knowledge-bases/{knowledgeBasePk}/labels/)
8. 列出知識庫中的標籤 (GET /api/knowledge-bases/{knowledgeBasePk}/labels/)
9. 取得特定標籤詳情 (GET /api/knowledge-bases/{knowledgeBasePk}/labels/{id})
)
10. 更新標籤 (PUT /api/knowledge-bases/{knowledgeBasePk}/labels/{id})
)
11. 部分更新標籤 (PATCH /api/knowledge-bases/{knowledgeBasePk}/labels/{id})
)
12. 刪除標籤 (DELETE /api/knowledge-bases/{knowledgeBasePk}/labels/{id})
)
13. 上傳檔案到知識庫 (POST /api/knowledge-bases/{knowledgeBasePk}/files/)
14. 批量刪除檔案 (POST /api/knowledge-bases/{knowledgeBasePk}/files/batch-delete/)
15. 列出知識庫中的檔案 (GET /api/knowledge-bases/{knowledgeBasePk}/files/) - 部分內容

這份文件目前看起來已經是繁體中文版本了。您是否需要我：

1. 確認翻譯品質並修正任何翻譯問題？
2. 等待文件的後半部分？
3. 還是有其他需求？

請告訴我您希望如何處理這份文件。

根據我的分析,您提供的內容實際上已經存在於文件 `/home/runner/work/skills-for-Agents/skills-for-Agents/source_repo/zh-tw/api-reference/知識庫（新版）.md` 中了。

讓我向您確認情況:

分析結果

我檢查了文件 `zh-tw/api-reference/知識庫（新版）.md`, 發現:

1. 文件已存在:該文件有 6829 行,已經包含了完整的 API 文件
2. 內容已翻譯:您提供的所有 API 端點都已經存在:
 - 列出知識庫中的檔案 (第 3275 行)
 - 取得特定檔案詳情 (第 3485 行)
 - 更新檔案資訊

- 部分更新檔案
- 刪除檔案 (第 4493 行)
- 創建新的 FAQ (第 4609 行)
- 批量刪除 FAQ (第 4903 行)
- 列出特定知識庫的所有 FAQ (第 5082 行)
- 取得特定 FAQ 詳情
- 更新 FAQ
- 部分更新 FAQ
- 刪除 FAQ (第 6022 行)

1. 內容一致:您提供的翻譯內容與文件中已存在的內容完全一致

結論:這份文件已經完整翻譯完成,不需要再進行任何操作。

您是否:

1. 想要確認文件內容是否正確?
2. 需要翻譯其他文件?
3. 或者有其他需求?

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

這是一個已經翻譯完成的繁體中文 API 文件。文件內容包含多個 API 端點的詳細說明，涵蓋：

1. 帳號註冊 - POST </api/auth/registration/>
2. 更換密碼 - POST </api/auth/password/change/>
3. 建立新組織 - POST </api/organizations/>
4. 新增成員至指定組織 - POST </api/organizations/{organizationPk}/members/>
5. 取得組織列表 - GET </api/organizations/>
6. 取得特定組織資訊 - GET </api/organizations/{id}/>
7. 取得當前用戶詳細資訊 - GET </api/users/current/>
8. 取得當前用戶權限 - GET </api/permissions/>
9. 取得指定組織的成員列表 - GET </api/organizations/{organizationPk}/members/>
10. 取得特定成員詳細資訊 - GET </api/organizations/{organizationPk}/members/{id}/>
11. 取得特定成員詳細資訊（匯出） - GET </api/organizations/{organizationPk}/members/export/>

12. 取得特定成員詳細資訊（匯出範本） - GET

`/api/organizations/{organizationPk}/members/export-template/`

13. 更新組織資訊 - PUT `/api/organizations/{id}/`

14. 更新當前用戶詳細資訊 - PUT `/api/users/current/`

15. 刪除指定成員 - DELETE `/api/organizations/{organizationPk}/members/{id}/`

每個端點都包含：

- HTTP 方法和路徑
- 參數說明
- 請求內容（包含欄位、類型、必填狀態、說明）
- 請求結構範例和請求範例值
- 多語言程式碼範例（Shell/Bash、JavaScript、Python、PHP）
- 回應內容（包含狀態碼、回應結構範例、回應範例值）

文件已經完整翻譯成繁體中文，包含所有 HTML 標籤、CSS 樣式和 GitBook 格式標記。所有技術術語都已適當翻譯，並保持了原始的格式和結構。

API Call Example (Shell)

List Contacts

GET

`/api/contacts/`

Parameters

Parameter Name	Required	Type	Description
<code>inbox</code>	X	string	Deprecated: Use inboxes instead
<code>inboxes</code>	X	string	Conversation platform ID (used to filter contacts from a specific inbox)
<code>page</code>	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);
  });

```

Python

```
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);
  });
}
```

```
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"  
}
```