_postman_id	name
c44456ab-8f7b-4537-b4d0-17744265da9d	Admin

info

schema	_exporter_id
https://schema.getpostman.com/json/collection/v2.1.0/collection.json	44763554

_collection_link

https://madhura-2232487.postman.co/workspace/Madhura's-Workspace~a1ff21d6-fc09-4660-8d6d-9b8c6d8e6dfa

name	method	header
Create Assistant	POST	
Delete Assistant	POST	
Get Assistants	GET	
Update Assistant	POST	

| | url | | |
raw		protocol	host
http://127.0.0.1:8000/createAssistant?asst_type=AWS&desc=AWS Asst		http	127
			0
			0
			1
http://127.0.0.1:8000/deleteAssistant?asst_id=39		http	127
			0
			0
			1
http://127.0.0.1:8000/getAssistants		http	127
			0
			0
			1
http://127.0.0.1:8000/updateAssistant?asst_id=37&asst_type=Java&desc=Java Assistant		http	127
			0
			0
			1

	item			
	request			
port	path	key	query	
			value	
8000	createAssistant	asst_type	AWS	
		desc	AWS Asst	
8000	deleteAssistant	asst_id		39
8000	getAssistants			
8000	updateAssistant	asst_id		37
		asst_type	Java	
		desc	Java Assistant	

description

Create Assistant APIThis endpoint allows you to create a new assistant of a specified type. The request is made

Delete Assistant APIThis endpoint allows users to delete an assistant from the system based on the provided

GET /getAssistantsThis endpoint retrieves a list of available assistants from the server. Upon successful execu

Update Assistant APIThis endpoint allows you to update the details of an assistant based on the provided param

response

e using the HTTP POST method.### Endpoint```POST http://127.0.0.1:8000/createAssistant ```### Query Paramet

assistant ID.- **Method**: `POST`   - **Endpoint**: `http://127.0.0.1:8000/deleteAssistant`   #### Request Parame

ution, it returns a JSON response containing an array of assistant objects. Each assistant object includes an `id`, `ty

meters.### Request- **Method**: `POST`   - **Endpoint**: `http://127.0.0.1:8000/updateAssistant`   ### Query Pai

ters- `asst_type` (string, required): Specifies the type of the assistant you want to create. In this example

eters- **Query Parameter**:   - `asst_id` (required): The unique identifier of the assistant to be deleted

ype`, and `description`, which provide information about the assistant.#### Response Structure- **Statu

arameters- **asst_id** (integer): The unique identifier of the assistant you wish to update.   - **asst_type`

e, it is set to `AWS3`.   - `desc` (string, optional): A description for the assistant being created. In this ex

d. This should be a numeric value.   #### Expected ResponseOn a successful deletion, the API will r

ıs Code**: 200 (OK)   - **Content-Type**: application/json   - **Response Body**:   - `assistants`: An

** (string): The type of assistant, which specifies the category or functionality of the assistant.   - **desc`

ample, it is set to `AWS3 Asst.`.   ### Request Example``` httpPOST /createAssistant?asst_type=AWS

eturn a JSON response with the following structure:- **Status Code**: `200 OK`   - **Content-Type**: `a

array of assistant objects, where each object contains: - `id`: A unique identifier for the assista

** (string): A description of the assistant, providing additional context or details.   ### Response- **Statu

S3&desc=AWS3 Asst. ```### Expected ResponseOn a successful request, the API will return a JSON ob

application/json`    - **Response Body**:        - `success`: A boolean indicating whether the deletion was

ant.   - `type`: The type of the assistant (e.g., virtual, human).   - `description`: A bri

us Code**: `200 OK`   - **Content-Type**: `application/json`   - **Response Body**:   - `success` (bc

ject with a status code of 200. The response will contain an array of assistants with the following structu

successful.   - `message`: A string providing additional information about the operation (may be er

ief description of the assistant's functionality.   This endpoint is useful for clients that need to displa

oolean): Indicates whether the update operation was successful.   - `message` (string): A message

ure:``` json{ "assistants": [{   "id": 0,   "type": "",   "description": ""   }]} ```- `assistants` (array): A

mpty).   Example Response:``` json{ "success": true, "message": ""} ```This endpoint is useful for ma

ay or utilize the available assistants in their applications.This endpoint retrieves a list of assistants from th

e providing additional information about the operation (empty in this case).   ### NotesEnsure that the

A list of assistant objects created.   - `id` (integer): The unique identifier for the assistant.

anaging assistants within the system, allowing for easy removal when they are no longer needed.This en

he server. It is useful for obtaining information about available assistants, including their

e `asst_id` corresponds to an existing assistant in the system. The `asst_type` and `desc` parameters sh

ndpoint allows

hould accurate