**SUPERFITY Database API** <sup>110</sup> API documentation for the SUPERTIFY database auth Authentication operations  $\wedge$ POST /auth/login Login and get a token POST /auth/register Register a new user accounts Manage accounts POST /accounts/ Create a new account GET /accounts/ Get all accounts DELETE /accounts/{account\_id} Delete an account /accounts/{account\_id} Update an account GET /accounts/{account\_id} Get an account by ID **USERS** Manage users ^ POST /users/ Create a new user GET /users/ Get all users GET /users/follower-counts Get all users with their follower counts GET /users/nickname/{nickname} Get a user by nickname DELETE /users/{user\_id} Delete a user PUT /users/{user\_id} Update a user GET /users/{user\_id} Get a user by ID followers Manage follower relationships POST /followers/ Create a new follower relationship GET /followers/ Get all follower relationships **DELETE** /followers/{user\_id\_1}/followers/{user\_id\_2} Delete a follower relationship GET /followers/{user\_id\_1}/followers/{user\_id\_2} Get a specific follower relationship playlists Manage playlists POST /playlists/ Create a new playlist GET /playlists/ Get all playlists **DELETE** /playlists/{playlist\_id} Delete a playlist PUT /playlists/{playlist\_id} Update a playlist GET /playlists/{playlist\_id} Get a playlist by ID playlist\_users Manage playlist-user relationships POST /playlist\_users/ Create a new playlist-user relationship GET /playlist\_users/ Get all playlist-user relationships DELETE /playlist\_users/{user}/{fallows} Delete a playlist-user relationship GET /playlist\_users/{user}/{fallows} Get a specific playlist-user relationship playlist\_songs Manage playlist-song relationships POST /playlist\_songs/ Create a new playlist-song relationship GET /playlist\_songs/ Get all playlist-song relationships DELETE /playlist\_songs/{playlist\_id}/songs/{song\_id} Delete a playlist-song relationship GET /playlist\_songs/{playlist\_id}/songs/{song\_id} Get a specific playlist-song relationship likes Manage likes for songs POST /likes/ Create a new like GET /likes/ Get all likes DELETE /likes/{user\_id}/songs/{song\_id} Delete a like GET /likes/{user\_id}/songs/{song\_id} Get a specific like songs Manage songs POST /songs/ Create a new song GET /songs/ Get all songs GET /songs/name/{song\_name} Get a song by name **DELETE** /songs/{song\_id} Delete a song PUT /songs/{song\_id} Update a song GET /songs/{song\_id} Get a song by ID genres Manage genres POST /genres/ Create a new genre GET /genres/ Get all genres **GET** /genres/most-listened-last-month Get the most listened genre in the last month DELETE /genres/{song\_id} Delete a genre GET /genres/{song\_id} Get a genre by song ID albums Manage albums POST /albums/ Create a new album GET /albums/ Get all albums GET /albums/streaming-stats Get total listening time for each album **DELETE** /albums/{album\_id} Delete an album PUT /albums/{album\_id} Update an album GET /albums/{album\_id} Get an album by ID album\_infos Manage relationships between albums and songs POST /album\_infos/ Create a new album-song relationship /album\_infos/ Get all album-song relationships **DELETE** /album\_infos/{album\_id}/songs/{song\_id} Delete an album-song relationship GET /album\_infos/{album\_id}/songs/{song\_id} Get a specific album-song relationship groups Manage groups POST /groups/ Create a new group GET /groups/ Get all groups DELETE /groups/{group\_id} Delete a group PUT /groups/{group\_id} Update a group GET /groups/{group\_id} Get a group by ID album\_groups Manage relationships between albums and groups POST /album\_groups/ Create a new album-group relationship GET /album\_groups/ Get all album-group relationships **DELETE** /album\_groups/{album\_id}/groups/{group\_id} Delete an album-group relationship GET /album\_groups/{album\_id}/groups/{group\_id} Get a specific album-group relationship artists Manage artists POST /artists/ Create a new artist GET /artists/ Get all artists DELETE /artists/{group\_id}/artists/{full\_name} Delete an artist GET /artists/{group\_id}/artists/{full\_name} Get an artist by group ID and name histories Manage user listening history POST /histories/ Create a new history record GET /histories/ Get all history records DELETE /histories/{user\_id}/history/{start\_time} Delete a history record GET /histories/{user\_id}/history/{start\_time} Get a specific history record Models > [...] > [...] Login 🗸 { > [...] > [...] Account ✓ { account\_id\* > [...] mail\* > [...] full\_name > [...] is\_subscriber > [...] registration\_date > [...] country > [...] sex > [...] language\* > [...] birth\_date > [...] User ✔ {
user\_id\* > [...] nickname\* > [...] favorite\_genre > [...] user\_image > [...] Follower V { user\_id\_1\* > [...] user\_id\_2\* > [...] Playlist • {
playlist\_id > [...] playlist\_name\* > [...] playlist\_description > [...] playlist\_image > [...] > [...] PlaylistUser V { > [...] fallows\* > [...] PlaylistSong < { playlist\_id\* > [...] song\_id\* > [...] Like ∨ { user\_id\* > [...] song\_id\* > [...] Song ✓ {
song\_id\* > [...] song\_name\* > [...] song\_time\* > [...] song\_image > [...] > [...] Genre ∨ { song\_id\* > [...] genre\* > [...] Album ∨ { album\_id\* > [...] album\_name\* > [...] about > [...] album\_image > [...] AlbumInfo ∨ { album\_id\* > [...] song\_id\* > [...] Group ✔ {
group\_id\* > [...] group\_name\* > [...] number\_of\_members > [...] creation\_date > [...] group\_image > [...] AlbumGroup ∨ { album\_id\* > [...]

group\_id\*

Artist ✔ {
group\_id\*

full\_name\*

History ✔ {
user\_id\*

start\_time\*

duration

song\*

> [...]

> [...]

> [...]

> [...]

> [...]

> [...]

> [...]