
google-music-proto

Release 2.10.0

thebigmunch

May 01, 2020

CONTENTS

1	Getting Started	1
1.1	mobileclient.models — Mobile API base classes	1
1.2	mobileclient.calls — Mobile API call classes	2
1.3	mobileclient.schemas — Mobile API response schemas	18
1.4	mobileclient.types — Mobile API object types	36
1.5	musicmanager.models — Music Manager API base classes	37
1.6	musicmanager.calls — Music Manager API call classes	37
	Python Module Index	41
	Index	43

GETTING STARTED

Install google-music-proto with `pip`.

```
$ pip install google-music-proto
```

1.1 `mobileclient.models` — Mobile API base classes

```
class MobileClientCall
```

```
    base_url = 'https://mclients.googleapis.com/sj/v2.5'
```

```
    follow_redirects = True
```

```
class MobileClientBatchCall(events_or_mutations)
```

```
    method = 'POST'
```

```
    batch_key = 'mutations'
```

```
    events_or_mutations
```

```
class MobileClientFeedCall(max_results=None, start_token=None, updated_min=-1)
```

```
    method = 'POST'
```

```
    max_results
```

```
    start_token
```

```
    updated_min
```

```
class MobileClientFetchCall
```

```
    method = 'GET'
```

```
class MobileClientStreamCall
```

```
    base_url = 'https://mclients.googleapis.com/music'
```

```
    method = 'GET'
```

```
class MobileClientSchema(extra=None, only=None, exclude=(), prefix="", strict=None, many=False,  
                        context=None, load_only=(), dump_only=(), partial=False)
```

```
class Meta

    strict = True

    check_for_unknown_fields(data)

    opts = <marshmallow.schema.SchemaOpts object>
```

1.2 mobileclient.calls — Mobile API call classes

Example of making an HTTP request using the `requests` library:

```
# Assuming an authenticated session named 'session'.
>>> from google_music_proto.mobileclient.calls import TrackFeed
>>> call = TrackFeed()
>>> response = session.request(
    call.method, call.url, params=call.params, headers=call.headers,
    data=call.body, allow_redirects=call.follow_redirects
)
```

1.2.1 Attributes

Call class instances have the following attributes:

body

Binary-encoded body of the HTTP request.

follow_redirects

Boolean to enable/disable request redirects.

headers

Headers to send with the HTTP request.

method

Method for the HTTP request.

params

Dict of parameters to send in the query string of the HTTP request.

url

URL for the HTTP request.

1.2.2 Classes

class ActivityRecordRealtime (*events_or_mutations*)

Record track play and rate events.

Use *play* to build track play event dicts. Use *rate* to build track rate events dicts.

Parameters *events* (*list* or *dict*) – A list of event dicts or a single event dict.

endpoint

activity/recordrealtime

method

POST

static play (*track_id*, *track_duration*, *, *play_time=None*, *stream_auth_id=None*)

Build a track play event.

Parameters

- **track_id** (*str*) – A track ID.
- **track_duration** (*int* or *str*) – The duration of the track.
- **play_time** (*int*, *Optional*) – The amount of time user played the track in seconds. Default: *track_duration*
- **stream_auth_id** (*str*, *Optional*) – The stream auth ID from a stream call's headers.

Returns An event dict.

Return type *dict*

static rate (*track_id*, *rating*)

Build a track rate event.

Parameters

- **track_id** (*str*) – A track ID.
- **rating** (*int*) – 0 (not rated), 1 (thumbs down), or 5 (thumbs up).

Returns An event dict.

Return type *dict*

class BrowseStationCategories

Get a listing of station categories from the browse stations tab.

endpoint

browse/stationcategories

method

GET

schema

BrowseStationCategoriesSchema

class BrowseStations (*station_category_id*)

Get a listing of stations by category from browse tab.

Parameters *station_category_id* (*str*) – A station category ID as found in *BrowseStationCategories* response.

```
    endpoint
        browse/stations

    method
        GET

    schema
        BrowseStationsSchema

class BrowseTopChart
    Get a listing of the default top charts.

    endpoint
        browse/topchart

    method
        GET

    schema
        BrowseTopChartSchema

class BrowseTopChartForGenre (genre_id)
    Get a listing of top charts for a top chart genre.

    Parameters genre_id (str) – A top chart genre ID as found in BrowseTopChartGenres
        response.

    endpoint
        browse/topchartforgenes

    method
        GET

    schema
        BrowseTopChartSchema

class BrowseTopChartGenres
    Get a listing of genres from the browse top charts tab.

    endpoint
        browse/topchartgenres

    method
        GET

    schema
        BrowseTopChartGenresSchema

class Config
    Get a listing of mobile client configuration settings.

    endpoint
        config

    method
        GET

    schema
        ConfigListSchema

class DeviceManagementInfo
    Get a listing of devices registered to a Google Music account.
```


endpoint
devicemanagementinfo

method
GET

schema
UserClientIDListSchema

class DeviceManagementInfoDelete (*device_id*)

Delete a registered device.

Parameters *device_id* (*str*) – A device ID as found in *DeviceManagementInfo* response.

endpoint
devicemanagementinfo

method
DELETE

class EphemeralTop (*max_results=1000, start_token=None, updated_min=None*)

Get a listing of ‘Thumbs Up’ store tracks.

Note: ‘Thumbs Up’ library tracks are handled client-side. Use the *TrackFeed* call to find library tracks with a 'rating' of 5.

Note: The track list is paged. Getting all tracks will require looping through all pages.

Parameters

- **max_results** (*int*, *Optional*) – The maximum number of results on returned page. Default: 1000
- **start_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds.

endpoint
ephemeral/top

method
POST

schema
EphemeralTopSchema

class ExploreGenres (*parent_genre_id=None*)

Get a listing of track genres.

Parameters *parent_genre_id* (*str*, *Optional*) – A genre ID. If given, a listing of this genre’s sub-genres is returned.

endpoint
explore/genres

method
GET

schema*GenreListSchema***class FetchAlbum** (*album_id*, *include_description=True*, *include_tracks=True*)

Get information about an album.

Parameters

- **album_id** (*str*) – The album ID to look up.
- **include_description** (*bool*, *Optional*) – Include description of the album in the response. Default: True
- **include_tracks** (*bool*, *Optional*) – Include tracks from the album in the response. Default: True

endpoint*fetchalbum***method**

GET

schema*AlbumSchema***class FetchArtist** (*artist_id*, *include_albums=True*, *num_related_artists=5*, *num_top_tracks=5*)

Get information about an artist.

Parameters

- **artist_id** (*str*) – The artist ID to look up.
- **include_albums** (*bool*, *Optional*) – Include albums from the artist in the response. Default: True
- **num_related_artists** (*int*, *Optional*) – The maximum number of related artists to include in the response. Default: 5
- **num_top_tracks** (*int*, *Optional*) – The maximum number of top tracks to include in the response. Default: 5

endpoint*fetchartist***method**

GET

schema*ArtistSchema***class FetchTrack** (*track_id*)

Get information about a track.

Parameters **track_id** (*str*) – A track ID to look up.**endpoint***fetchtrack***method**

GET

schema*StoreTrackSchema*

class IsPlaylistShared (*playlist_id*)

Check if a playlist is shared.

Parameters *playlist_id* (*str*) – A playlist ID.

endpoint

isplaylistshared

method

GET

schema

IsPlaylistSharedSchema

class ListenNowGetDismissedItems

Get a listing of items dismissed from Listen Now tab.

endpoint

listennow/get_dismissed_items

method

GET

schema

ListenNowDismissedItemsSchema

class ListenNowGetListenNowItems

Get a listing of Listen Now items.

Note: This does not include situations; use *ListenNowSituations* to get situations.

endpoint

listennow/getlistennowitems

method

GET

schema

ListenNowItemListSchema

class ListenNowSituations (*tz_offset=None*)

Get a listing of Listen Now situations.

Parameters *tz_offset* (*int*, *Optional*) – A time zone offset from UTC in seconds. Default is automatic detection.

endpoint

listennow/situations

method

POST

schema

ListenNowSituationListSchema

class PlaylistBatch (*events_or_mutations*)

Create, delete, and edit playlists.

Use *create* to build playlist creation mutation dicts. Use *delete* to build playlist delete mutation dicts. Use *edit* to build playlist edit mutation dicts.

Parameters *mutations* (*list* or *dict*) – A list of mutation dicts or a single mutation dict.

endpoint

playlistbatch

method

POST

static create (*name*, *description*, *type_*, *, *owner_name=None*, *share_state=None*,
share_token=None)

Build a playlist create event.

Parameters

- **name** (*str*) – Name to give the playlist.
- **description** (*str*) – Description to give the playlist.
- **type_** (*str*) – 'SHARED' if subscribing to a public playlist, 'USER_GENERATED' if creating a playlist.
- **share_state** (*str*, *Optional*) – 'PUBLIC' to share the created playlist, 'PRIVATE' otherwise.
- **owner_name** (*str*, *Optional*) – Owner name when subscribing to a playlist.
- **share_token** (*str*, *Optional*) – The share token of a shared playlist to subscribe to.

Returns A mutation dict.

Return type dict

static delete (*playlist_id*)

Build a playlist delete event.

Parameters **playlist_id** (*str*) – A playlist ID.

Returns A mutation dict.

Return type dict

static edit (*playlist_id*, *name*, *description*, *share_state*)

Build a playlist edit event.

Parameters

- **playlist_id** (*str*) – A playlist ID.
- **name** (*str*) – Name to give the playlist.
- **description** (*str*) – Description to give the playlist.
- **share_state** (*str*) – 'PUBLIC' to share the playlist, 'PRIVATE' otherwise.

Returns A mutation dict.

Return type dict

class PlaylistEntriesBatch (*events_or_mutations*)

Create, delete, and edit playlist entries.

Use *create* to build playlist entry creation mutation dicts. Use *delete* to build playlist entry delete mutation dicts. Use *update* to build playlist entry update mutation dicts.

Parameters **mutations** (*list or dict*) – A list of mutation dicts or a single mutation dict.

endpoint

plentriesbatch

method

POST

static create (*track_id*, *playlist_id*, *, *playlist_entry_id=None*, *preceding_entry_id=None*, *following_entry_id=None*)

Build a playlist entry create event.

Parameters

- **track_id** (*str*) – A track ID.
- **playlist_id** (*str*) – A playlist ID.
- **playlist_entry_id** (*str*, *Optional*) – A playlist entry ID to assign to the created entry. Default: Automatically generated.
- **preceding_entry_id** (*str*, *Optional*) – The playlist entry ID that should precede the added track. None if entry is to be in first position. Default: None
- **following_entry_id** (*str*, *Optional*) – The playlist entry ID that should follow the added track. None if entry is to be in last position. Default: None

Returns A mutation dict.

Return type dict

static delete (*playlist_entry_id*)

Build a playlist entry delete event.

Parameters **playlist_entry_id** (*str*) – A playlist entry ID.

Returns A mutation dict.

Return type dict

static update (*playlist_entry*, *, *preceding_entry_id=None*, *following_entry_id=None*)

Build a playlist entry update event.

Parameters

- **playlist_id** (*str*) – A playlist ID.
- **preceding_entry_id** (*str*, *Optional*) – The playlist entry ID that should precede the added track. None if entry is to be in first position. Default: None
- **following_entry_id** (*str*, *Optional*) – The playlist entry ID that should follow the added track. None if entry is to be in last position. Default: None

Returns A mutation dict.

Return type dict

class PlaylistEntriesShared (*share_tokens*, *max_results=250*, *start_token=None*, *updated_min=0*)

Get a listing of shared playlist entries.

Note: The shared playlist entries list is paged. Getting all shared playlist entries will require looping through all pages.

Parameters

- **max_results** (*int*, *Optional*) – The maximum number of results on returned page. Default: 250

- **start_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

endpoint

plentries/shared

method

POST

schema

SharedPlaylistEntryListSchema

class PlaylistEntryFeed (*max_results=250, start_token=None, updated_min=-1*)

Get a listing of user playlist entries.

Note: The playlist entry list is paged. Getting all playlist entries will require looping through all pages.

Parameters

- **max_results** (*int*, *Optional*) – The maximum number of results on returned page. Default: 250
- **start_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

endpoint

plentryfeed

method

POST

schema

PlaylistEntryListSchema

class PlaylistFeed (*max_results=250, start_token=None, updated_min=-1*)

Get a listing of library playlists.

Note: The playlist list is paged. Getting all playlists will require looping through all pages.

Parameters

- **max_results** (*int*, *Optional*) – The maximum number of results on returned page. Default: 250
- **start_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

```

    endpoint
        playlistfeed

    method
        POST

    schema
        PlaylistListSchema

class Playlists
    Get a listing of library playlists.

    endpoint
        playlists

    method
        GET

    schema
        PlaylistListSchema

class PlaylistsCreate (name, description, public=False)
    Create a playlist.

    Parameters
        • name (str) – Name to give the playlist.
        • description (str) – Description to give the playlist.
        • public (bool) – If True and account has a subscription, make playlist public. Default:
          False

    endpoint
        playlists

    method
        POST

    schema
        PlaylistSchema

class PlaylistsDelete (playlist_id)
    Delete a playlist.

    Parameters playlist_id (str) – A playlist ID.

    endpoint
        playlists

    method
        DELETE

class PlaylistsUpdate (playlist_id, name, description, public=False)
    Edit a playlist.

    endpoint
        playlists

    method
        PUT

    schema
        PlaylistSchema

```

```
class PodcastBrowse (podcast_genre_id='JZCpodcasttopchartall')
    Get a listing of podcasts from Podcasts browse tab.

    Parameters podcast_genre_id (str, Optional) – A podcast genre ID as found in
        PodcastBrowseHierarchy. Default: 'JZCpodcasttopchartall'

    endpoint
        podcast/browse

    method
        GET

    schema
        PodcastBrowseSchema

class PodcastBrowseHierarchy
    Get a listing of genres from Podcasts browse tab dropdown.

    endpoint
        podcast/browsehierarchy

    method
        GET

    schema
        PodcastBrowseHierarchySchema

class PodcastEpisode (device_id=None, max_results=250, start_token=None, updated_min=-1)
    Retrieve list of episodes from user-subscribed podcast series.
```

Note: The podcast episode list is paged. Getting all podcast episodes will require looping through all pages.

Parameters

- **device_id** (*str*) – A mobile device ID.
- **max_results** (*int*, *Optional*) – The maximum number of results on returned page. Default: 250
- **start_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

```
endpoint
    podcastepisode

method
    GET

schema
    PodcastEpisodeListSchema
```

```
class PodcastEpisodeStreamURL (podcast_episode_id, quality='hi', device_id=None)
    Get a URL to stream a podcast episode.
```

Parameters

- **podcast_episode_id** (*str*) – A podcast episode ID.
- **device_id** (*str*) – A mobile device ID.

- **quality**(*str*, *Optional*) –

Stream quality is one of:

- 'hi' (320Kbps)
- 'med' (160Kbps)
- 'low' (128Kbps)

Default: 'hi'

class PodcastFetchEpisode(*podcast_episode_id*)

Get information about a podcast episode.

Parameters **podcast_episode_id**(*str*) – A podcast episode ID to look up.

endpoint

podcast/fetchepisode

method

GET

schema

PodcastEpisodeSchema

class PodcastFetchSeries(*podcast_series_id*, *max_episodes=50*)

Get information about a podcast series.

Parameters **podcast_series_id**(*str*) – A podcast series ID to look up.

endpoint

podcast/fetchseries

method

GET

schema

PodcastSeriesSchema

class PodcastSeries(*device_id=None*, *max_results=250*, *start_token=None*, *updated_min=-1*)

Retrieve list of user-subscribed podcast series.

Note: The podcast series list is paged. Getting all podcast series will require looping through all pages.

Parameters

- **device_id**(*str*) – A mobile device ID.
- **max_results**(*int*, *Optional*) – The maximum number of results on returned page. Default: 250
- **start_token**(*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min**(*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

endpoint

podcastseries

method

GET

schema

PodcastSeriesListSchema

class PodcastSeriesBatchMutate (*events_or_mutations*)

endpoint

podcastseries/batchmutate

method

POST

class Query (*query*, *, *max_results=100*, ***kwargs*)

Search Google Music.

Parameters

- **query** (*str*) – Search text.
- **max_results** (*int*, *Optional*) – Maximum number of results per type to retrieve. Google only accepts values up to 100. Default: 100
- **kwargs** (*bool*, *Optional*) –

Any of:

- albums
- artists
- genres
- playlists,
- podcasts
- situations
- songs
- stations
- videos

set to True will include that result type in the response. Setting none of them will include all result types in the response.

endpoint

query

method

GET

schema

SearchResponseSchema

class QuerySuggestion (*query*)

Get a search suggestion.

Parameters **query** (*str*) – Search text.

class RadioStation (*max_results=250*, *start_token=None*, *updated_min=-1*)

Generate a listing of stations.

Note: The station list is paged. Getting all stations will require looping through all pages.

Parameters

- **max_results** (*int*, *Optional*) – The maximum number of results on returned page. Default: 250
- **start_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

endpoint

radio/station

method

POST

schema*RadioListSchema***class RadioStationFeed** (*station_infos=None, num_entries=25, num_stations=4*)

Generate stations and get tracks from station(s).

Parameters

- **station_infos** (*list*) –

A list of station dicts containing keys:

- 'station_id' or 'seed'
- 'num_entries'
- 'library_content_only'
- 'recently_played'

station_id is a station ID.

'seed' is a dict containing a seed ID and seed type ('seedType'). See [StationSeedType](#) for seed type values.

A seed ID can be:

- artistId
- albumId
- genreId
- trackId (store track)
- trackLockerId (library track)

num_entries is the maximum number of tracks to return from the station.

library_content_only when True limits the station to library tracks. Default: False

recently_played is a list of dicts in the form of {'id': '', 'type'} where id is a track ID and type is 0 for a library track and 1 for a store track.

- **num_entries** (*int*) – The total number of tracks to return. Default: 25
- **num_stations** (*int*) – The number of stations to return when no station_infos is provided. Default: 4

endpoint
radio/stationfeed

method
POST

schema
[RadioFeedSchema](#)

class RadioStationTrackStreamURL (*track_id*, *wentry_id*, *session_token*, *quality*='hi', *device_id*=None)
Get a URL to stream a station track with a free account.

Note: Subscribed accounts should use [TrackStreamURL](#).

Unlike the other stream calls, this returns JSON with a 'url' key, not the location in headers.

Parameters

- **track_id** (*str*) – A station track ID.
- **wentry_id** (*str*) – The wentryid from a station track dict.
- **session_token** (*str*) – The sessionToken from a [RadioStationFeed](#) response.
- **quality** (*str*, *Optional*) –

Stream quality is one of:

- 'hi' (320Kbps)
- 'med' (160Kbps)
- 'low' (128Kbps)

Default: 'hi'

- **device_id** (*str*) – A mobile device ID.

class TrackBatch (*events_or_mutations*)
Add, delete, and edit library tracks.

Note: This previously supported editing most metadata. It now only supports changing rating. However, changing the rating should be done with [ActivityRecordRealtime](#) and [ActivityRecordRealtime.rate](#) instead.

Use [add](#) to build track add mutation dicts. Use [delete](#) to build track delete mutation dicts. Use [edit](#) to build track edit mutation dicts.

Parameters mutations (*list* or *dict*) – A list of mutation dicts or a single mutation dict.

endpoint
trackbatch

method
POST

schema
[TrackBatchSchema](#)

static add (*store_track*)

Build a track add event.

Parameters **store_track** (*dict*) – A store track dict.

Returns A mutation dict.

Return type *dict*

static delete (*track_id*)

Build a track add event.

Parameters **track_id** (*str*) – A track ID.

Returns A mutation dict.

Return type *dict*

static edit (*track*)

Build a track edit event.

Parameters **track** (*dict*) – A library track dict.

Returns A mutation dict.

Return type *dict*

class TrackFeed (*max_results=250, start_token=None, updated_min=-1*)

Get a listing of library tracks.

Note: The track list is paged. Getting all tracks will require looping through all pages.

Parameters

- **max_results** (*int, Optional*) – The maximum number of results on returned page. Default: 250
- **start_token** (*str, Optional*) – The token of the page to return. Default: Not sent to get first page.
- **updated_min** (*int, Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

endpoint

trackfeed

method

POST

schema

TrackListSchema

class TrackStreamURL (*track_id, quality='hi', device_id=None*)

Get a URL to stream a track.

Parameters

- **device_id** (*str*) – A mobile device ID.
- **track_id** (*str*) – A library or store track ID. A Google Music subscription is required to stream store tracks.
- **quality** (*str, Optional*) –

Stream quality is one of:

- 'hi' (320Kbps)
- 'med' (160Kbps)
- 'low' (128Kbps)

Default: 'hi'

class Tracks (*max_results=1000, start_token=None, updated_min=-1*)

Get a listing of library tracks.

Note: The track list is paged. Getting all tracks will require looping through all pages.

Parameters

- **max_results** (*int, Optional*) – The maximum number of results on returned page. Max allowed is 49995. Default: 1000
- **start_token** (*str, Optional*) – The token of the page to return. Default: None to get first page.
- **updated_min** (*int, Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

endpoint

tracks

method

GET

schema

TrackListSchema

1.3 mobileclient.schemas — Mobile API response schemas

All mobile client schemas are a subclass of `marshmallow.Schema` with the `strict` option set to `True`. Unknown fields trigger a `marshmallow.ValidationError` exception.

class AlbumSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'albumArtist': 'albumArtist',
    'albumArtRef': 'albumArtRef',
    'albumId': 'albumId',
    'artist': 'artist',
    'artistId': ['artistId'],
    'contentType': 'contentType',
    'description': 'description',
    'description_attribution': ATIBUTION,
    'explicitType': 'explicitType',
    'kind': 'sj#album',
    'name': 'name',
    'tracks': [STORE_TRACK],
```

(continues on next page)

(continued from previous page)

```

    'year': 2000
}

```

class ArtRefSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'url': 'url'
}

```

class ArtistSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'albums': [ALBUM],
    'artistArtRef': 'artistArtRef',
    'artistArtRefs': [IMAGE_REF],
    'artistBio': 'artistBio',
    'artistId': 'artistId',
    'artist_bio_attribution': ATTRIBUTION,
    'kind': 'sj#artist',
    'name': 'name',
    'related_artists': [],
    'topTracks': [STORE_TRACK],
    'total_albums': 1
}

```

class AttributionSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'kind': 'sj#attribution',
    'license_title': 'license title',
    'license_url': 'license url',
    'source_title': 'source title',
    'source_url': 'source url'
}

```

class BrowseStationCategoriesSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'kind': 'sj#getStationCategoriesResponse',
    'root': STATION_CATEGORY
}

```

class BrowseStationsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'kind': 'sj#getStationsResponse',
    'stations': [RADIO_STATION]
}
```

class BrowseTopChartGenresSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'genres': [GENRE_REF]
}
```

class BrowseTopChartSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'chart': TOP_CHART,
    'header': TOP_CHART_HEADER
}
```

class ColorStyleSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'blue': 0,
    'green': 0,
    'red': 0
}
```

class ConfigEntrySchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'key': 'key',
    'kind': 'sj#configEntry',
    'value': 'value'
}
```

class ConfigListEntriesSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'entries': [CONFIG_ENTRY]
}
```

class ConfigListSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:


```
{
  'data': CONFIG_LIST_ENTRIES,
  'kind': 'sj#configList'
}
```

class DeviceManagementInfoSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'friendlyName': 'friendlyName',
  'id': 'id',
  'kind': 'sj#devicemanagementinfo',
  'lastAccessedTimeMs': 42,
  'smartPhone': True,
  'type': 'type'
}
```

class DistilledContextWrapperSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'distilledContextToken': 'distilledContextToken'
}
```

class EphemeralTopItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'items' = [STORE_TRACKS]
}
```

class EphemeralTopSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'data': EPHEMERAL_TOP_ITEMS,
  'kind': 'sj#trackList',
  'nextPageToken': 'nextPageToken'
}
```

class ExploreTabEntityGroupSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'continuationToken': 'continuationToken',
  'entities': [EXPLORE_TAB_ENTITY],
  'group_type': 'group_type',
  'kind': 'sj#exEntityGroup',

```

(continues on next page)

(continued from previous page)

```
'start_position': 0,  
'title': 'title'  
}
```

```
class ExploreTabEntitySchema (extra=None, only=None, exclude=(), prefix="", strict=None,  
                               many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{  
    'album': ALBUM,  
    'genre': GENRE,  
    'kind': 'sj#exEntity',  
    'playlist': PLAYLIST,  
    'track': STORE_TRACK  
}
```

```
class ExploreTabSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False,  
                        context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{  
    'groups': [EXPLORE_TAB_ENTITY_GROUP],  
    'kind': 'sj#exTab',  
    'tab_type': 'tab_type'  
}
```

```
class ExploreTabsSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False,  
                          context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{  
    'kind': 'sj#exGetTabsResponse',  
    'tabs': [EXPLORE_TAB]  
}
```

```
class GenreListSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False,  
                       context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{  
    'genres': [GENRE],  
    'kind': 'sj#exGetMusicGenresResponse'  
}
```

```
class GenreRefSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None,  
                      load_only=(), dump_only=(), partial=False)
```

Validates:

```
{  
    'id': 'id',  
    'kind': 'sj#genreRef',  
    'title': 'title'  
}
```

```
class GenreSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None,  
                   load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
  'children': ['child'],
  'id': 'id',
  'images': [ART_REF],
  'kind': 'sj#musicGenre',
  'name': 'name',
  'parentId': 'parentId'
}
```

class ImageColorStylesSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'accent': {
    'blue': 0,
    'green': 0,
    'red': 0
  },
  'primary': {
    'blue': 0,
    'green': 0,
    'red': 0
  },
  'scrim': {
    'blue': 0,
    'green': 0,
    'red': 0
  }
}
```

class ImageRefSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'aspectRatio': 'aspectRatio',
  'autogen': False,
  'colorStyles': {
    'accent': {
      'blue': 0,
      'green': 0,
      'red': 0
    },
    'primary': {
      'blue': 0,
      'green': 0,
      'red': 0
    },
    'scrim': {
      'blue': 0,
      'green': 0,
      'red': 0
    }
  },
}
```

(continues on next page)

(continued from previous page)

```
'kind': 'sj#imageRef',
'url': 'url'
}
```

```
class IsPlaylistSharedSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                             many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'is_shared': True
}
```

```
class ListenNowAlbumIDSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                              many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'artist': 'artist',
    'metajamCompactKey': 'metajamCompactKey',
    'title': 'title'
}
```

```
class ListenNowAlbumSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                             many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'artist_metajam_id': 'artist_metajam_id',
    'artist_name': 'artist_name',
    'artist_profile_image': ART_REF,
    'description': 'description',
    'description_attribution': CONTRIBUTION,
    'explicitType': 'explicitType',
    'id': LISTEN_NOW_ALBUM_ID,
    'title': 'title'
}
```

```
class ListenNowDismissedItemsSchema (extra=None, only=None, exclude=(), prefix="",
                                     strict=None, many=False, context=None, load_only=(),
                                     dump_only=(), partial=False)
```

Validates:

```
{
    'kind': 'sj#listenNowDismissedItemList',
    'minLastModifiedIgnored': True
}
```

```
class ListenNowItemListSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                               many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
  'kind': 'sj#listenNowItemList',
  'listennow_items': [LISTEN_NOW_ITEM]
}
```

class ListenNowItemSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'album': LISTEN_NOW_ALBUM,
  'compositeArtRefs': [IMAGE_REF],
  'images': [IMAGE_REF],
  'kind': 'sj#listennowitem',
  'radio_station': LISTEN_NOW_RADIO_STATION,
  'suggestion_reason': 'suggestion_reason',
  'suggestion_text': 'suggestion_text',
  'type': 'type'
}
```

class ListenNowRadioStationIDSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'seeds': [RADIO_SEED]
}
```

class ListenNowRadioStationSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'highlight_color': 'highlight_color',
  'id': LISTEN_NOW_RADIO_STATION_ID,
  'profile_image': ART_REF,
  'title': 'title'
}
```

class ListenNowSituationListSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'distilledContextWrapper': DISTILLED_CONTEXT_WRAPPER,
  'primaryHeader': 'primaryHeader',
  'situations': [SITUATION],
  'subHeader': 'subHeader'
}
```

class PlaylistEntryItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'items': [PLAYLIST_ENTRY]
}
```

```
class PlaylistEntryListSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                               many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'data': PLAYLIST_ENTRY_ITEMS,
    'kind': 'sj#playlistEntryList',
    'nextPageToken': 'nextPageToken'
}
```

```
class PlaylistEntrySchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                            many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'absolutePosition': 'absolutePosition',
    'clientId': 'clientId',
    'creationTimestamp': 'creationTimestamp',
    'deleted': False,
    'id': 'id',
    'kind': 'sj#playlistEntry',
    'lastModifiedTimestamp': 'lastModifiedTimestamp',
    'playlistId': 'playlistId',
    'track': STORE_TRACK,
    'trackId': 'trackId'
}
```

```
class PlaylistListItemsSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                                many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'items': [PLAYLIST]
}
```

```
class PlaylistListSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False,
                           context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'data': PLAYLIST_LIST_ITEMS,
    'kind': 'sj#playlistList',
    'nextPageToken': 'nextPageToken'
}
```

```
class PlaylistSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None,
                       load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
  'accessControlled': False,
  'albumArtRef': [ART_REF],
  'clientId': 'clientId',
  'contentType': 'contentType',
  'creationTimestamp': 'creationTimestamp',
  'deleted': False,
  'description': 'description',
  'explicitType': 'explicitType',
  'id': 'id',
  'kind': 'sj#playlist',
  'lastModifiedTimestamp': 'lastModifiedTimestamp',
  'name': 'name',
  'ownerName': 'ownerName',
  'ownerProfilePhotoUrl': 'ownerProfilePhotoUrl',
  'recentTimestamp': 'recentTimestamp',
  'shareState': 'shareState',
  'shareToken': 'shareToken',
  'type': 'USER_GENERATED'
}
```

class PodcastBrowseHierarchySchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'groups': [PODCAST_GENRE]
}
```

class PodcastBrowseSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'series': [PODCAST_SERIES]
}
```

class PodcastEpisodeListItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'items': [PODCAST_EPISODE]
}
```

class PodcastEpisodeListSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'data': PODCAST_EPISODE_LIST_ITEMS,
  'kind': 'sj#podcastEpisodeList',
  'nextPageToken': 'nextPageToken'
}
```

(continues on next page)

(continued from previous page)

```
}
```

```
class PodcastEpisodeSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                             many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'art': [IMAGE_REF],
    'author': 'author',
    'deleted': False,
    'description': 'description',
    'durationMillis': 'durationMillis',
    'episodeId': 'episodeId',
    'explicitType': 'explicitType',
    'fileSize': 'fileSize',
    'playbackPositionMillis': 'playbackPositionMillis',
    'publicationTimestampMillis': 'publicationTimestampMillis',
    'seriesId': 'seriesId',
    'seriesTitle': 'seriesTitle',
    'title': 'title'
}
```

```
class PodcastGenreSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False,
                           context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'displayName': 'displayName',
    'id': 'id',
    'subgroups': []
}
```

```
class PodcastSeriesListItemsSchema (extra=None, only=None, exclude=(), prefix="",
                                     strict=None, many=False, context=None, load_only=(),
                                     dump_only=(), partial=False)
```

Validates:

```
{
    'items': [PODCAST_SERIES]
}
```

```
class PodcastSeriesListSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                               many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'data': PODCAST_SERIES_LIST_ITEMS,
    'kind': 'sj#podcastSeriesList',
    'nextPageToken': 'nextPageToken'
}
```

```
class PodcastSeriesSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                            many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:


```
{
  'art': [IMAGE_REF],
  'author': 'author',
  'continuationToken': 'continuationToken',
  'copyright': 'copyright',
  'description': 'description',
  'episodes': [PODCAST_EPISODE],
  'explicitType': 'explicitType',
  'link': 'link',
  'seriesId': 'seriesId',
  'title': 'title',
  'totalNumEpisodes': 42,
  'userPreferences': PODCAST_SERIES_USER_PREFERENCES
}
```

class PodcastSeriesUserPreferencesSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'autoDownload': False,
  'notifyOnNewEpisode': False,
  'subscribed': True
}
```

class RadioFeedItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'currentTimestampMillis': 'currentTimestampMillis',
  'radioConstraints': RADIO_CONSTRAINTS
  'stations': [RADIO_STATION],
}
```

class RadioFeedSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'data': RADIO_FEED_ITEMS,
  'kind': 'sj#radioFeed'
}
```

class RadioListItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'items': [RADIO_STATION]
}
```

class RadioListSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'data': RADIO_LIST_ITEMS,
  'kind': 'sj#radioList',
  'nextPageToken': 'nextPageToken'
}
```

class RadioSeedMetadataSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'artist': ARTIST,
  'genre': GENRE,
  'kind': 'sj#radioSeedMetadata',
  'track': STORE_TRACK
}
```

class RadioSeedSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'albumId': 'albumId',
  'artistId': 'artistId',
  'curatedStationId': 'curatedStationId',
  'genreId': 'genreId',
  'kind': 'sj#radioSeed',
  'metadataSeed': RADIO_SEED_METADATA,
  'seedType': 'seedType',
  'trackId': 'trackId',
  'trackLockerId': 'trackLockerId'
}
```

class RadioStationSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'byline': 'byline',
  'clientId': 'clientId',
  'compositeArtRefs': [IMAGE_REF],
  'contentTypes': ['contentType'],
  'deleted': False,
  'description': 'description',
  'enforcementResult': False,
  'id': 'id',
  'imageUrl': 'imageUrl',
  'imageUrls': [IMAGE_REF],
  'inLibrary': True,
  'kind': 'sj#radioStation',
  'lastModifiedTimestamp': 'lastModifiedTimestamp',
  'name': 'name',
  'recentTimestamp': 'recentTimestamp',
  'seed': RADIO_SEED,
  'sessionToken': 'sessionToken',
  'skipEventHistory': [],

```

(continues on next page)

(continued from previous page)

```

    'stationSeeds': [RADIO_SEED],
    'tracks': [STORE_TRACK]
}

```

class SearchResponseSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'clusterDetail': [SEARCH_RESULT_CLUSTER],
    'kind': 'sj#searchresponse'
}

```

class SearchResultSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'album': ALBUM,
    'artist': ARTIST,
    'playlist': {k: v for k, v in PLAYLIST.items() if k not in [
↪ 'creationTimestamp', 'id', 'lastModifiedTimestamp']},
    'score': 42,
    'series': PODCAST_SERIES,
    'situation': SITUATION,
    'station': RADIO_STATION,
    'track': STORE_TRACK,
    'type': 'type',
    'youtube_video': SEARCH_RESULT_VIDEO
}

```

class SearchResultVideoSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

SEARCH_RESULT_VIDEO = {
    'id': 'id',
    'kind': 'sj#video',
    'thumbnails': [VIDEO_THUMBNAIL],
    'title': 'title'
}

```

class SearchSuggestedQuerySchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'suggestionCategory': 'suggestionCategory',
    'suggestion_string': 'suggestion_string',
    'type': 'type'
}

```

```
class SearchSuggestionSchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                               many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'suggested_queries': [SEARCH_SUGGESTED_QUERY]
}
```

```
class SharedPlaylistEntryItemSchema (extra=None, only=None, exclude=(), prefix="",
                                      strict=None, many=False, context=None, load_only=(),
                                      dump_only=(), partial=False)
```

Validates:

```
{
    'playlistEntry': SHARED_PLAYLIST_ENTRY,
    'responseCode': 'responseCode',
    'shareToken': 'shareToken'
}
```

```
class SharedPlaylistEntryListSchema (extra=None, only=None, exclude=(), prefix="",
                                      strict=None, many=False, context=None, load_only=(),
                                      dump_only=(), partial=False)
```

Validates:

```
{
    'entries': [SHARED_PLAYLIST_ENTRY_ITEM],
    'kind': 'sj#listSharedPlaylistEntriesResponse'
}
```

```
class SharedPlaylistEntrySchema (extra=None, only=None, exclude=(), prefix="", strict=None,
                                  many=False, context=None, load_only=(), dump_only=(),
                                  partial=False)
```

Validates:

```
{
    'absolutePosition': 'absolutePosition',
    'creationTimestamp': 'creationTimestamp',
    'deleted': False,
    'id': 'id',
    'kind': 'sj#playlistEntry',
    'lastModifiedTimestamp': 'lastModifiedTimestamp',
    'source': 'source',
    'track': STORE_TRACK,
    'trackId': 'trackId'
}
```

```
class SituationSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False,
                       context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'description': 'description',
    'id': 'id',
    'imageUrl': 'imageUrl',
    'situations': [],
    'stations': [RADIO_STATION],
    'title': 'title',
}
```

(continues on next page)

(continued from previous page)

```

    'wideImageUrl': 'wideImageUrl'
}

```

class StationCategorySchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'display_name': 'display_name',
    'id': 'id',
    'kind': 'sj#stationCategory',
    'subcategories': []
}

```

class StoreTrackSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'album': 'album',
    'albumArtRef': [IMAGE_REF],
    'albumArtist': 'albumArtist',
    'albumAvailableForPurchase': True,
    'albumId': 'albumId',
    'artist': 'artist',
    'artistArtRef': [IMAGE_REF],
    'artistId': ['artistId'],
    'composer': 'composer',
    'discNumber': 1,
    'durationMillis': 'durationMillis',
    'estimatedSize': 'estimatedSize',
    'explicitType': 'explicitType',
    'genre': 'genre',
    'kind': 'sj#track',
    'lastModifiedTimestamp': 'lastModifiedTimestamp',
    'lastRatingChangeTimestamp': 'lastRatingChangeTimestamp',
    'nid': 'nid',
    'playCount': 42,
    'primaryVideo': VIDEO,
    'rating': '5',
    'storeId': 'storeId',
    'title': 'title',
    'trackAvailableForPurchase': True,
    'trackAvailableForSubscription': True,
    'trackNumber': 1,
    'trackType': 'trackType',
    'year': 2000
}

```

class TopChartHeaderSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```

{
    'header_image': IMAGE_REF,

```

(continues on next page)

(continued from previous page)

```
{
    'kind': 'sj#topChartHeader'
}
```

class TopChartSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'albums': [ALBUM],
    'kind': 'sj#topChart',
    'tracks': [STORE_TRACK]
}
```

class TrackBatchItemSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'client_id': 'client_id',
    'id': 'id',
    'response_code': 'response_code'
}
```

class TrackBatchSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'mutate_response': [TRACK_BATCH_ITEMS]
}
```

class TrackListItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'items': [STORE_TRACK, UPLOADED_TRACK]
}
```

class TrackListSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
    'data': TRACK_LIST_ITEMS,
    'kind': 'sj#trackList',
    'nextPageToken': 'nextPageToken'
}
```

class TrackSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Combination of *StoreTrackSchema* and *UploadedTrackSchema*.

class UploadedTrackSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'album': 'album',
  'albumArtRef': [IMAGE_REF],
  'albumArtist': 'albumArtist',
  'albumId': 'albumId',
  'artist': 'artist',
  'artistArtRef': [IMAGE_REF],
  'artistId': ['artistId'],
  'beatsPerMinute': 142,
  'clientId': 'clientId',
  'comment': 'comment',
  'composer': 'composer',
  'creationTimestamp': 'creationTimestamp',
  'deleted': False,
  'discNumber': 1,
  'durationMillis': 'durationMillis',
  'estimatedSize': 'estimatedSize',
  'explicitType': 'explicitType',
  'genre': 'genre',
  'id': 'id',
  'kind': 'sj#track',
  'lastModifiedTimestamp': 'lastModifiedTimestamp',
  'lastRatingChangeTimestamp': 'lastRatingChangeTimestamp',
  'nid': 'nid',
  'playCount': 42,
  'primaryVideo': VIDEO,
  'rating': '5',
  'recentTimestamp': 'recentTimestamp',
  'storeId': 'storeId',
  'title': 'title',
  'totalDiscCount': 1,
  'totalTrackCount': 1,
  'trackNumber': 1,
  'trackType': 'trackType',
  'year': 2000
}
```

class UserClientIDListItemsSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'items': [DEVICE_MANAGEMENT_INFO]
}
```

class UserClientIDListSchema (*extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False*)

Validates:

```
{
  'data': USER_CLIENT_ID_LIST_ITEMS,
  'kind': 'sj#userClientIdList'
}
```

```
class VideoSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'id': 'id',
    'kind': 'sj#video',
    'thumbnails': [
        {
            'height': 100,
            'url': 'url',
            'width': 100
        }
    ]
}
```

```
class VideoThumbnailSchema (extra=None, only=None, exclude=(), prefix="", strict=None, many=False, context=None, load_only=(), dump_only=(), partial=False)
```

Validates:

```
{
    'height': 100,
    'url': 'url',
    'width': 100
}
```

1.4 mobileclient.types — Mobile API object types

```
class ListenNowItemType
```

Listen Now item type.

```
class QueryResultType
```

Search result types.

```
search_result['type']
```

```
class StationSeedType
```

Station seed types.

```
station['seed']['seedType']
```

```
class TrackRating
```

Track ratings.

```
song['rating']
```


1.5 musicmanager.models — Music Manager API base classes

class MusicManagerCall

```

    base_url = 'https://android.clients.google.com/upsj'

    request_type
        alias of upload_pb2.UploadResponse

    response_type
        alias of upload_pb2.UploadResponse

    property body
        Binary-encoded body of the HTTP request.

    parse_response(response_headers, response_body)

```

1.6 musicmanager.calls — Music Manager API call classes

Example of making an HTTP request using the `requests` library:

```

# Assuming an authenticated session called 'session'.
>>> from google_music_proto.musicmanager.calls import ExportIds
>>> call = ExportIds(<uploader_id>)
>>> response = session.request(
    call.method, call.url, params=call.params, headers=call.headers,
    data=call.body, allow_redirects=call.follow_redirects
)

```

1.6.1 Attributes

Call classes have the following attributes:

body

Binary-encoded body of the HTTP request.

follow_redirects

Boolean to enable/disable request redirects.

headers

Headers to send with the HTTP request.

method

Method for the HTTP request.

params

Dict of parameters to send in the query string of the HTTP request.

url

URL for the HTTP request.

1.6.2 Classes

class **ClientState** (*uploader_id*)

Get information about the state of a Google Music account.

Note: This provides things like the quota for uploaded songs.

Parameters **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.

class **Export** (*uploader_id*, *song_id*)

Download a song from a Google Music library.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **song_id** (*str*) – A song ID.

class **ExportIDs** (*uploader_id*, *continuation_token=None*, *export_type=1*, *updated_min=-1*)

Get a listing of uploaded and purchased library tracks.

Note: The track list is paged. Getting all tracks will require looping through all pages.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **continuation_token** (*str*, *Optional*) – The token of the page to return. Default: Not sent to get first page.
- **export_type** (*int*, *Optional*) – The type of tracks to return: 1 for all tracks, 2 for promotional and purchased. Default: 1
- **updated_min** (*int*, *Optional*) – List changes since the given Unix epoch time in microseconds. Default lists all changes.

class **GetJobs** (*uploader_id*)

Get a listing of pending upload jobs.

Parameters **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.

class **Metadata** (*uploader_id*, *tracks*)

Send upload track metadata to Google Music.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **tracks** (*list*) – A list of tracks in the form of `locker_pb2.Track`. Use `Metadata.get_track_info` to generate locker tracks from audio files.

static **get_track_info** (*song*)

Create a locker track from an audio file.

Parameters `song` (*os.PathLike* or *str* or *audio_metadata.Format*) – The path to an audio file or an instance of *audio_metadata.Format*.

Returns A locker track of the given audio file.

Return type *locker_pb2.Track*

class `Sample` (*uploader_id*, *track_samples*)
Send samples of audio files to Google Music.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **track_samples** (*list*) – A list of track samples in the form of *upload_pb2.TrackSample*. Use *Sample.generate_sample* to generate a track sample from an audio file.

static `generate_sample` (*song*, *track*, *sample_request*, *, *external_art=None*, *no_sample=False*)
Generate a track sample from an audio file.

Parameters

- **track** (*locker_pb2.Track*) – A locker track of the audio file as created by *Metadata.get_track_info*.
- **sample_request** (*upload_pb2.SignedChallengeInfo*) – The 'signed_challenge_info' portion for the audio file from the *Metadata* response.
- **external_art** (*bytes*, *Optional*) – The binary data of an external album art image. If not provided, embedded album art will be used, if present.
- **no_sample** (*bool*, *Optional*) – Don't generate an audio sample from song; send empty audio sample. Default: Create an audio sample using ffmpeg/avconv.

class `ScottyAgentPost` (*uploader_id*, *server_track_id*, *track*, *song*, *external_art=None*, *total_song_count=1*, *total_uploaded_count=0*)
Request an upload URL for a track from Google Music.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **server_track_id** (*str*) – The server ID of the audio file to upload as given in the response of *Metadata* or *Sample*.
- **track** (*locker_pb2.Track*) – A locker track of the audio file as created by *Metadata.get_track_info*.
- **song** (*os.PathLike* or *str* or *audio_metadata.Format*) – The path to an audio file or an instance of *audio_metadata.Format*.
- **external_art** (*bytes*, *Optional*) – The binary data of an external album art image. If not provided, embedded album art will be used, if present.
- **total_song_count** (*int*, *Optional*) – Total number of songs to be uploaded in this session. Default: 1
- **total_uploaded_count** (*int*, *Optional*) – Number of songs uploaded in this session. Default: 0

class **ScottyAgentPut** (*upload_url*, *audio_file*, *content_type*='audio/mpeg')

Upload a file to a Google Music library.

Parameters

- **upload_url** (*str*) – The upload URL given by *ScottyAgentPost* response.
- **audio_file** (*os.PathLike* or *str* or *bytes*) – An audio file as *os.PathLike*, a file/bytes-like object, or binary data.
- **content_type** (*str*) – The mime type to be sent in the ContentType header field. Default: 'audio/mpeg'

class **UpAuth** (*uploader_id*, *uploader_name*)

Authenticate device as a Music Manager uploader.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **uploader_name** (*str*) – The name given to the device in the Google Music device listing.

class **UploadState** (*uploader_id*, *state*)

Notify Google Music of the state of an upload.

Parameters

- **uploader_id** (*str*) – A unique ID given as a MAC address. Only one Music Manager client may use a given uploader ID.
- **state** (*str*) – Can be one of 'START', 'PAUSED', 'STOPPED'. Will be uppcased if lowercase is given.

PYTHON MODULE INDEX

g

- google_music_proto.mobileclient.calls,
3
- google_music_proto.mobileclient.models,
1
- google_music_proto.mobileclient.schemas,
18
- google_music_proto.mobileclient.types,
36
- google_music_proto.musicmanager.calls,
37
- google_music_proto.musicmanager.models,
37

INDEX

A

ActivityRecordRealtime (class in *google_music_proto.mobileclient.calls*), 3
 add() (*TrackBatch* static method), 16
 AlbumSchema (class in *google_music_proto.mobileclient.schemas*), 18
 ArtistSchema (class in *google_music_proto.mobileclient.schemas*), 19
 ArtRefSchema (class in *google_music_proto.mobileclient.schemas*), 19
 AttributionSchema (class in *google_music_proto.mobileclient.schemas*), 19

B

base_url (*MobileClientCall* attribute), 1
 base_url (*MobileClientStreamCall* attribute), 1
 base_url (*MusicManagerCall* attribute), 37
 batch_key (*MobileClientBatchCall* attribute), 1
 body, 2
 body (in module *google_music_proto.musicmanager.calls*), 37
 body() (*MusicManagerCall* property), 37
 BrowseStationCategories (class in *google_music_proto.mobileclient.calls*), 3
 BrowseStationCategoriesSchema (class in *google_music_proto.mobileclient.schemas*), 19
 BrowseStations (class in *google_music_proto.mobileclient.calls*), 3
 BrowseStationsSchema (class in *google_music_proto.mobileclient.schemas*), 19
 BrowseTopChart (class in *google_music_proto.mobileclient.calls*), 4
 BrowseTopChartForGenre (class in *google_music_proto.mobileclient.calls*), 4
 BrowseTopChartGenres (class in *google_music_proto.mobileclient.calls*), 4

BrowseTopChartGenresSchema (class in *google_music_proto.mobileclient.schemas*), 20
 BrowseTopChartSchema (class in *google_music_proto.mobileclient.schemas*), 20

C

check_for_unknown_fields() (*MobileClientSchema* method), 2
 ClientState (class in *google_music_proto.musicmanager.calls*), 38
 ColorStyleSchema (class in *google_music_proto.mobileclient.schemas*), 20
 Config (class in *google_music_proto.mobileclient.calls*), 4
 ConfigEntrySchema (class in *google_music_proto.mobileclient.schemas*), 20
 ConfigListEntriesSchema (class in *google_music_proto.mobileclient.schemas*), 20
 ConfigListSchema (class in *google_music_proto.mobileclient.schemas*), 20
 create() (*PlaylistBatch* static method), 8
 create() (*PlaylistEntriesBatch* static method), 9

D

delete() (*PlaylistBatch* static method), 8
 delete() (*PlaylistEntriesBatch* static method), 9
 delete() (*TrackBatch* static method), 17
 DeviceManagementInfo (class in *google_music_proto.mobileclient.calls*), 4
 DeviceManagementInfoDelete (class in *google_music_proto.mobileclient.calls*), 5
 DeviceManagementInfoSchema (class in *google_music_proto.mobileclient.schemas*), 21
 DistilledContextWrapperSchema (class in *google_music_proto.mobileclient.schemas*), 21

E

edit() (*PlaylistBatch* static method), 8
 edit() (*TrackBatch* static method), 17

endpoint (*ActivityRecordRealtime* attribute), 3
 endpoint (*BrowseStationCategories* attribute), 3
 endpoint (*BrowseStations* attribute), 3
 endpoint (*BrowseTopChart* attribute), 4
 endpoint (*BrowseTopChartForGenre* attribute), 4
 endpoint (*BrowseTopChartGenres* attribute), 4
 endpoint (*Config* attribute), 4
 endpoint (*DeviceManagementInfo* attribute), 4
 endpoint (*DeviceManagementInfoDelete* attribute), 5
 endpoint (*EphemeralTop* attribute), 5
 endpoint (*ExploreGenres* attribute), 5
 endpoint (*FetchAlbum* attribute), 6
 endpoint (*FetchArtist* attribute), 6
 endpoint (*FetchTrack* attribute), 6
 endpoint (*IsPlaylistShared* attribute), 7
 endpoint (*ListenNowGetDismissedItems* attribute), 7
 endpoint (*ListenNowGetListenNowItems* attribute), 7
 endpoint (*ListenNowSituations* attribute), 7
 endpoint (*PlaylistBatch* attribute), 7
 endpoint (*PlaylistEntriesBatch* attribute), 8
 endpoint (*PlaylistEntriesShared* attribute), 10
 endpoint (*PlaylistEntryFeed* attribute), 10
 endpoint (*PlaylistFeed* attribute), 10
 endpoint (*Playlists* attribute), 11
 endpoint (*PlaylistsCreate* attribute), 11
 endpoint (*PlaylistsDelete* attribute), 11
 endpoint (*PlaylistsUpdate* attribute), 11
 endpoint (*PodcastBrowse* attribute), 12
 endpoint (*PodcastBrowseHierarchy* attribute), 12
 endpoint (*PodcastEpisode* attribute), 12
 endpoint (*PodcastFetchEpisode* attribute), 13
 endpoint (*PodcastFetchSeries* attribute), 13
 endpoint (*PodcastSeries* attribute), 13
 endpoint (*PodcastSeriesBatchMutate* attribute), 14
 endpoint (*Query* attribute), 14
 endpoint (*RadioStation* attribute), 15
 endpoint (*RadioStationFeed* attribute), 15
 endpoint (*TrackBatch* attribute), 16
 endpoint (*TrackFeed* attribute), 17
 endpoint (*Tracks* attribute), 18
 EphemeralTop (class in *google_music_proto.mobileclient.calls*), 5
 EphemeralTopItemsSchema (class in *google_music_proto.mobileclient.schemas*), 21
 EphemeralTopSchema (class in *google_music_proto.mobileclient.schemas*), 21
 events_or_mutations (*MobileClientBatchCall* attribute), 1
 ExploreGenres (class in *google_music_proto.mobileclient.calls*), 5
 ExploreTabEntityGroupSchema (class in *google_music_proto.mobileclient.schemas*), 21
 ExploreTabEntitySchema (class in *google_music_proto.mobileclient.schemas*), 22
 ExploreTabSchema (class in *google_music_proto.mobileclient.schemas*), 22
 ExploreTabsSchema (class in *google_music_proto.mobileclient.schemas*), 22
 Export (class in *google_music_proto.musicmanager.calls*), 38
 ExportIDs (class in *google_music_proto.musicmanager.calls*), 38

F

 FetchAlbum (class in *google_music_proto.mobileclient.calls*), 6
 FetchArtist (class in *google_music_proto.mobileclient.calls*), 6
 FetchTrack (class in *google_music_proto.mobileclient.calls*), 6
 follow_redirects, 2
 follow_redirects (in *module google_music_proto.musicmanager.calls*), 37
 follow_redirects (*MobileClientCall* attribute), 1

G

 generate_sample() (*Sample* static method), 39
 GenreListSchema (class in *google_music_proto.mobileclient.schemas*), 22
 GenreRefSchema (class in *google_music_proto.mobileclient.schemas*), 22
 GenreSchema (class in *google_music_proto.mobileclient.schemas*), 22
 get_track_info() (*Metadata* static method), 38
 GetJobs (class in *google_music_proto.musicmanager.calls*), 38
 google_music_proto.mobileclient.calls (module), 3
 google_music_proto.mobileclient.models (module), 1
 google_music_proto.mobileclient.schemas (module), 18
 google_music_proto.mobileclient.types (module), 36
 google_music_proto.musicmanager.calls (module), 37
 google_music_proto.musicmanager.models (module), 37

H

 headers, 2

headers	(in module <i>google_music_proto.musicmanager.calls</i>), 37	method (<i>BrowseTopChartForGenre</i> attribute), 4 method (<i>BrowseTopChartGenres</i> attribute), 4 method (<i>Config</i> attribute), 4 method (<i>DeviceManagementInfo</i> attribute), 5 method (<i>DeviceManagementInfoDelete</i> attribute), 5
I		
ImageColorStylesSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 23	method (<i>EphemeralTop</i> attribute), 5 method (<i>ExploreGenres</i> attribute), 5
ImageRefSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 23	method (<i>FetchAlbum</i> attribute), 6 method (<i>FetchArtist</i> attribute), 6
IsPlaylistShared	(class in <i>google_music_proto.mobileclient.calls</i>), 6	method (<i>FetchTrack</i> attribute), 6 method (in module <i>google_music_proto.musicmanager.calls</i>), 37
IsPlaylistSharedSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 24	method (<i>IsPlaylistShared</i> attribute), 7 method (<i>ListenNowGetDismissedItems</i> attribute), 7 method (<i>ListenNowGetListenNowItems</i> attribute), 7 method (<i>ListenNowSituations</i> attribute), 7
L		
ListenNowAlbumIDSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 24	method (<i>MobileClientBatchCall</i> attribute), 1 method (<i>MobileClientFeedCall</i> attribute), 1
ListenNowAlbumSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 24	method (<i>MobileClientFetchCall</i> attribute), 1 method (<i>MobileClientStreamCall</i> attribute), 1
ListenNowDismissedItemsSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 24	method (<i>PlaylistBatch</i> attribute), 8 method (<i>PlaylistEntriesBatch</i> attribute), 8
ListenNowGetDismissedItems	(class in <i>google_music_proto.mobileclient.calls</i>), 7	method (<i>PlaylistEntriesShared</i> attribute), 10 method (<i>PlaylistEntryFeed</i> attribute), 10 method (<i>PlaylistFeed</i> attribute), 11
ListenNowGetListenNowItems	(class in <i>google_music_proto.mobileclient.calls</i>), 7	method (<i>Playlists</i> attribute), 11 method (<i>PlaylistsCreate</i> attribute), 11 method (<i>PlaylistsDelete</i> attribute), 11 method (<i>PlaylistsUpdate</i> attribute), 11
ListenNowItemListSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 24	method (<i>PodcastBrowse</i> attribute), 12 method (<i>PodcastBrowseHierarchy</i> attribute), 12
ListenNowItemSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 25	method (<i>PodcastEpisode</i> attribute), 12 method (<i>PodcastFetchEpisode</i> attribute), 13 method (<i>PodcastFetchSeries</i> attribute), 13 method (<i>PodcastSeries</i> attribute), 13
ListenNowItemType	(class in <i>google_music_proto.mobileclient.types</i>), 36	method (<i>PodcastSeriesBatchMutate</i> attribute), 14 method (<i>Query</i> attribute), 14
ListenNowRadioStationIDSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 25	method (<i>RadioStation</i> attribute), 15 method (<i>RadioStationFeed</i> attribute), 16
ListenNowRadioStationSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 25	method (<i>TrackBatch</i> attribute), 16 method (<i>TrackFeed</i> attribute), 17
ListenNowSituationListSchema	(class in <i>google_music_proto.mobileclient.schemas</i>), 25	method (<i>Tracks</i> attribute), 18
ListenNowSituations	(class in <i>google_music_proto.mobileclient.calls</i>), 7	<i>MobileClientBatchCall</i> (class in <i>google_music_proto.mobileclient.models</i>), 1
M		
max_results	(<i>MobileClientFeedCall</i> attribute), 1	<i>MobileClientCall</i> (class in <i>google_music_proto.mobileclient.models</i>), 1
Metadata	(class in <i>google_music_proto.musicmanager.calls</i>), 38	<i>MobileClientFeedCall</i> (class in <i>google_music_proto.mobileclient.models</i>), 1
method	2	<i>MobileClientFetchCall</i> (class in <i>google_music_proto.mobileclient.models</i>), 1
method (<i>ActivityRecordRealtime</i> attribute), 3		
method (<i>BrowseStationCategories</i> attribute), 3		
method (<i>BrowseStations</i> attribute), 4		
method (<i>BrowseTopChart</i> attribute), 4		

MobileClientSchema	(class in 11 google_music_proto.mobileclient.models),	1
MobileClientSchema.Meta	(class in 11 google_music_proto.mobileclient.models),	1
MobileClientStreamCall	(class in 11 google_music_proto.mobileclient.models),	1
MusicManagerCall	(class in 11 google_music_proto.musicmanager.models),	37
opts	(<i>MobileClientSchema</i> attribute),	2
P		
params	, 2	
params	(in module google_music_proto.musicmanager.calls),	37
parse_response()	(<i>MusicManagerCall</i> method),	37
play()	(<i>ActivityRecordRealtime</i> static method),	3
PlaylistBatch	(class in 7 google_music_proto.mobileclient.calls),	7
PlaylistEntriesBatch	(class in 8 google_music_proto.mobileclient.calls),	8
PlaylistEntriesShared	(class in 9 google_music_proto.mobileclient.calls),	9
PlaylistEntryFeed	(class in 10 google_music_proto.mobileclient.calls),	10
PlaylistEntryItemsSchema	(class in 25 google_music_proto.mobileclient.schemas),	25
PlaylistEntryListSchema	(class in 26 google_music_proto.mobileclient.schemas),	26
PlaylistEntrySchema	(class in 26 google_music_proto.mobileclient.schemas),	26
PlaylistFeed	(class in 10 google_music_proto.mobileclient.calls),	10
PlaylistListItemsSchema	(class in 26 google_music_proto.mobileclient.schemas),	26
PlaylistListSchema	(class in 26 google_music_proto.mobileclient.schemas),	26
Playlists	(class in 11 google_music_proto.mobileclient.calls),	11
PlaylistSchema	(class in 26 google_music_proto.mobileclient.schemas),	26
PlaylistsCreate	(class in 11 google_music_proto.mobileclient.calls),	11
PlaylistsDelete	(class in 11 google_music_proto.mobileclient.calls),	11
PlaylistsUpdate	(class in 11 google_music_proto.mobileclient.calls),	11
PodcastBrowse	(class in 11 google_music_proto.mobileclient.calls),	11
PodcastBrowseHierarchy	(class in 12 google_music_proto.mobileclient.calls),	12
PodcastBrowseHierarchySchema	(class in 27 google_music_proto.mobileclient.schemas),	27
PodcastBrowseSchema	(class in 27 google_music_proto.mobileclient.schemas),	27
PodcastEpisode	(class in 12 google_music_proto.mobileclient.calls),	12
PodcastEpisodeListItemsSchema	(class in 27 google_music_proto.mobileclient.schemas),	27
PodcastEpisodeListSchema	(class in 27 google_music_proto.mobileclient.schemas),	27
PodcastEpisodeSchema	(class in 28 google_music_proto.mobileclient.schemas),	28
PodcastEpisodeStreamURL	(class in 12 google_music_proto.mobileclient.calls),	12
PodcastFetchEpisode	(class in 13 google_music_proto.mobileclient.calls),	13
PodcastFetchSeries	(class in 13 google_music_proto.mobileclient.calls),	13
PodcastGenreSchema	(class in 28 google_music_proto.mobileclient.schemas),	28
PodcastSeries	(class in 13 google_music_proto.mobileclient.calls),	13
PodcastSeriesBatchMutate	(class in 14 google_music_proto.mobileclient.calls),	14
PodcastSeriesListItemsSchema	(class in 28 google_music_proto.mobileclient.schemas),	28
PodcastSeriesListSchema	(class in 28 google_music_proto.mobileclient.schemas),	28
PodcastSeriesSchema	(class in 28 google_music_proto.mobileclient.schemas),	28
PodcastSeriesUserPreferencesSchema	(class in 29 google_music_proto.mobileclient.schemas),	29

Q

Query (class in *google_music_proto.mobileclient.calls*),
14

QueryResultType (class in
google_music_proto.mobileclient.types),
36

QuerySuggestion (class in
google_music_proto.mobileclient.calls),
14

R

RadioFeedItemsSchema (class in
google_music_proto.mobileclient.schemas), 29

RadioFeedSchema (class in
google_music_proto.mobileclient.schemas), 29

RadioListItemsSchema (class in
google_music_proto.mobileclient.schemas), 29

RadioListSchema (class in
google_music_proto.mobileclient.schemas), 29

RadioSeedMetadataSchema (class in
google_music_proto.mobileclient.schemas), 30

RadioSeedSchema (class in
google_music_proto.mobileclient.schemas), 30

RadioStation (class in
google_music_proto.mobileclient.calls),
14

RadioStationFeed (class in
google_music_proto.mobileclient.calls),
15

RadioStationSchema (class in
google_music_proto.mobileclient.schemas), 30

RadioStationTrackStreamURL (class in
google_music_proto.mobileclient.calls),
16

rate() (*ActivityRecordRealtime* static method), 3

request_type (*MusicManagerCall* attribute), 37

response_type (*MusicManagerCall* attribute), 37

S

Sample (class in *google_music_proto.musicmanager.calls*),
39

schema (*BrowseStationCategories* attribute), 3

schema (*BrowseStations* attribute), 4

schema (*BrowseTopChart* attribute), 4

schema (*BrowseTopChartForGenre* attribute), 4

schema (*BrowseTopChartGenres* attribute), 4

schema (*Config* attribute), 4

schema (*DeviceManagementInfo* attribute), 5

schema (*EphemeralTop* attribute), 5

schema (*ExploreGenres* attribute), 5

schema (*FetchAlbum* attribute), 6

schema (*FetchArtist* attribute), 6

schema (*FetchTrack* attribute), 6

schema (*IsPlaylistShared* attribute), 7

schema (*ListenNowGetDismissedItems* attribute), 7

schema (*ListenNowGetListenNowItems* attribute), 7

schema (*ListenNowSituations* attribute), 7

schema (*PlaylistEntriesShared* attribute), 10

schema (*PlaylistEntryFeed* attribute), 10

schema (*PlaylistFeed* attribute), 11

schema (*Playlists* attribute), 11

schema (*PlaylistsCreate* attribute), 11

schema (*PlaylistsUpdate* attribute), 11

schema (*PodcastBrowse* attribute), 12

schema (*PodcastBrowseHierarchy* attribute), 12

schema (*PodcastEpisode* attribute), 12

schema (*PodcastFetchEpisode* attribute), 13

schema (*PodcastFetchSeries* attribute), 13

schema (*PodcastSeries* attribute), 13

schema (*Query* attribute), 14

schema (*RadioStation* attribute), 15

schema (*RadioStationFeed* attribute), 16

schema (*TrackBatch* attribute), 16

schema (*TrackFeed* attribute), 17

schema (*Tracks* attribute), 18

ScottyAgentPost (class in
google_music_proto.musicmanager.calls),
39

ScottyAgentPut (class in
google_music_proto.musicmanager.calls),
39

SearchResponseSchema (class in
google_music_proto.mobileclient.schemas), 31

SearchResultSchema (class in
google_music_proto.mobileclient.schemas), 31

SearchResultVideoSchema (class in
google_music_proto.mobileclient.schemas), 31

SearchSuggestedQuerySchema (class in
google_music_proto.mobileclient.schemas), 31

SearchSuggestionSchema (class in
google_music_proto.mobileclient.schemas), 31

SharedPlaylistEntryItemSchema (class in
google_music_proto.mobileclient.schemas), 32

SharedPlaylistEntryListSchema (class in
google_music_proto.mobileclient.schemas), 32

SharedPlaylistEntrySchema (class in
google_music_proto.mobileclient.schemas), 32

SituationSchema (class in
google_music_proto.mobileclient.schemas), 32

start_token (*MobileClientFeedCall* attribute), 1

StationCategorySchema (class in
google_music_proto.mobileclient.schemas), 33

StationSeedType (class in
google_music_proto.mobileclient.types),
36

StoreTrackSchema (class in
google_music_proto.mobileclient.schemas), 33

strict (*MobileClientSchema.Meta* attribute), 2

T

TopChartHeaderSchema (class in *google_music_proto.mobileclient.schemas*), 33

TopChartSchema (class in *google_music_proto.mobileclient.schemas*), 34

TrackBatch (class in *google_music_proto.mobileclient.calls*), 16

TrackBatchItemSchema (class in *google_music_proto.mobileclient.schemas*), 34

TrackBatchSchema (class in *google_music_proto.mobileclient.schemas*), 34

TrackFeed (class in *google_music_proto.mobileclient.calls*), 17

TrackListItemsSchema (class in *google_music_proto.mobileclient.schemas*), 34

TrackListSchema (class in *google_music_proto.mobileclient.schemas*), 34

TrackRating (class in *google_music_proto.mobileclient.types*), 36

Tracks (class in *google_music_proto.mobileclient.calls*), 18

TrackSchema (class in *google_music_proto.mobileclient.schemas*), 34

TrackStreamURL (class in *google_music_proto.mobileclient.calls*), 17

U

UpAuth (class in *google_music_proto.musicmanager.calls*), 40

update() (*PlaylistEntriesBatch* static method), 9

updated_min (*MobileClientFeedCall* attribute), 1

UploadedTrackSchema (class in *google_music_proto.mobileclient.schemas*), 34

UploadState (class in *google_music_proto.musicmanager.calls*), 40

url, 2

url (in module *google_music_proto.musicmanager.calls*), 37

UserClientIDListItemsSchema (class in *google_music_proto.mobileclient.schemas*), 35

UserClientIDListSchema (class in *google_music_proto.mobileclient.schemas*), 35

V

VideoSchema (class in *google_music_proto.mobileclient.schemas*), 35

VideoThumbnailSchema (class in *google_music_proto.mobileclient.schemas*), 36