# C:\Users\s2770873\AppData\Local\Microsoft\Windows\INetCache\Content.Word\logo.png

# Introduction

The Juceer (July, Celia & Erwin) song API is organized around representational state transfer (REST). Our API uses standard HTTP response codes and authentication. Juceer simplifies song data selection by various variables such as song information, genre, artist and ranking. Further on, the basic documentation of the Juceer API is given.

*Note: The Juceer API documentation is a “living” document. Version 1.0*

## Filter songs by artist name

GET song/artist/:artist

Returns all songs of identified artist name.

Resource URL: <https://api.juceer.com/0.1/song/artist/:artist> (This URL is an example and is not determined yet)

### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Required | Description | Default value | Example |
| artist name | Required | The name of the desired artist | - | Michael Jackson |
| year | Optional | The year of release of a song | - | 1969 |
| genre | Optional | The name of the desired genre | - | pop |

Example Request 1: GET <https://api.juceer.com/0.1/song/artist/Michael-Jackson> (Returns songs from Michael Jackson)

Example Request 2: GET <https://api.juceer.com/0.1/song/artist/Michael-Jackson?year=1969&genre=pop> (Returns songs from Michael Jackson in the pop genre release in 1969)

### Responses:

Success: Code: 200

Content: {all songs from specific artist}

Error:

“If missing parameters or artist name not found”

Code: 400 Bad Request

Content: { error : "The parameters introduce are not valid" }

Or

Code: 404 Not Found

Content: { error : "The Artist name cannot be found" }

## Filter songs by genre

GET song/genre/:genre

Returns all songs of identified genre.

Resource URL: <https://api.juceer.com/0.1/song/genre/:genre>

### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Required | Description | Default value | Example |
| genre | Required | The name of the desired genre | - | pop |

Example Request: GET <https://api.juceer.com/0.1/song/genre/pop>

### Responses:

Success: Code: 200

Content: {all songs from specific genre}

Error:

“If genre incorrect or no artist found from specific genre”

Code: 400 Bad Request

Content: { error : "The genre name doesn’t exist" }

Or

Code: 404 Not Found

Content: { error : "No artist from this genre can be found" }

## Filter song by dataset ID

GET song/id/:id

Returns the specified song by the id parameter.

Resource URL: https://api.juceer.com/0.1/song/id/:id

### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Required | Description | Default value | Example |
| ID | Required | The ID of a song | - | 12345678910 |

Example Request: GET <https://api.juceer.com/0.1/song/id/12345687910>

### Responses:

Success:

Code: 200

Content: {show song}

Error:

“If ID incorrect”

Code: 404 Not Found

Content: { error : "Incorrect ID" }

## Filter song by year

GET song/year/:year

Returns the songs specified by the year.

Resource URL: https://api.juceer.com/0.1/song/year/:year

### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Required | Description | Default value | Example |
| year | Required | The year of release of a song | - | 1969 |

Example Request: GET <https://api.juceer.com/0.1/song/year/1969>

### Responses:

Success: Code: 200

Content: {show songs given year}

Error:

“If year incorrect (no songs belonging to that year)”

Code: 404 Not Found

Content: { error : "No songs belonging to this year" }

## Order songs by popularity

GET song/popularity/:popularity

Returns the song specified by the popularity. In case the second parameter is identified, Juceer will return the list of songs by popularity from the first parameter (popularity) to the second (popularityEnd).

Resource URL: https://api.juceer.com/0.1/song/popularity/:popularity

### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Required | Description | Default value | Example |
| popularity | Required | The rank in popularity of a song | - | 1 |
| popularityEnd | Optional | The end of popularity ranking list | - | 50 |
| year | Optional | The year of desired popularity ranking | - | 1969 |

Example Request 1: GET <https://api.juceer.com/0.1/song/popularity/1?50> (Returns the top 50 songs)

Example Request 2: GET <https://api.juceer.com/0.1/song/popularity/1?popularityEnd=50&year=1969> (Returns the top 50 songs from 1969)

### Responses:

Success: Code: 200

Content: {show popularity ranking}

Error:

“If parameters not introduced correctly or year incorrect”

Code: 400 Bad Request

Content: { error : "Parameters missing" }

Or

Code: 404 Not Found

Content: { error : "The parameter introduced is not valid" }

## Order songs by descriptive statistics

GET song/statistics/:statistics

Returns the descriptive statistics of the identified artist. An optional parameter is added to specify a desired year.

Resource URL: https://api.juceer.com/0.1/song/statistics/:statistics

### Parameters

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Required | Description | Default value | Example |
| statistics | Required | The desired descriptive statistics | Mean | Standard deviation |
| artist | Required | The name of the desired artist | - | Michael Jackson |
| year | Optional | The year of desired descriptive statistics | - | 1969 |

Example Request 1: GET <https://api.juceer.com/0.1/song/statistics/artist=michael-jackson> (Returns the mean statistics of Michael Jackson songs)

Example Request 2: GET <https://api.juceer.com/0.1/song/statistics/statistics=standard-deviation&artist=michael-jackson&year=1969> (Returns the standard deviation statistics of Michael Jackson songs in 1969)

### Responses:

Success: Code: 200

Content: {show statistics asked by parameters}

Error:

“If parameters not introduced correctly”

Code: 400 Bad Request

Content: { error : "Parameters incorrect"

### Errors:

We use conventional HTTP response codes to indicate the success or failure of an API request. In general codes starting with 2XX indicate success in the request and those with 4XX indicate error. If the server fails it indicates 5XX errors. We indicated for each API probable errors. We can also consider for all of the above for example a server error and a forbidden access. If a user encounters an error he/she should return to the previous access.

Code: 500 Internal Server Error

Content: { error : "Internal Server Error. Refresh"

Code: 403 Forbidden

Content: { error : "Page Forbidden. Return home page"