



API documentation

## Introduction

The Juceer (July, Celia & Erwin) song API is organized around representational state transfer (REST). Our API uses standard HTTP response codes. 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.

### Sort artists by artist name (show the whole list)

GET /juceer/0.1/artists/ HTTP/1.1

This feature returns the artist identified by a particular artist name.

Resource URL: <http://127.0.0.1:8000/juceer/0.1/artists/>

### Examples:

Request 1:

http GET <http://127.0.0.1:8000/juceer/0.1/artists/>

Request 2:

http GET <http://127.0.0.1:8000/juceer/0.1/artist/>

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_id": "ARTW7I31187B9A4CA9",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARTW7I31187B9A4CA9/",
      "artist_name": "!!!"
    },
    {
      "artist_id": "ARGKPBC1187B9ADF8E",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARGKPBC1187B9ADF8E/",
      "artist_name": "\"Jean LumiA\"\\re\""
    },
    ...

  ],
  "title": "List of artists"
```

}

Error:

HTTP/1.1 404 Not Found

### Find artists by artist id

GET /juceer/artists/{artist\_id}/ HTTP/1.1

This feature returns the artist identified by a particular artist id code.

Resource URL: [http://127.0.0.1:8000/juceer/0.1/artists/{artist\\_id}/](http://127.0.0.1:8000/juceer/0.1/artists/{artist_id}/)

### Examples:

Request 1:

http GET

<http://127.0.0.1:8000/juceer/0.1/artists/ARW1MUW1187FB4D5A1/>

Request 2:

http GET <http://127.0.0.1:8000/juceer/0.1/artists/S0MEN0NSENSE/>

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object": {
    "artist_hottnesss": 0.608849019,
    "artist_id": "ARW1MUW1187FB4D5A1",
    "artist_name": "3 Doors Down",
    "artist_terms": "post-grunge"
  },
  "songs_of_the_artist":
    "http://127.0.0.1:8000/juceer/0.1/artists/ARW1MUW1187FB4D5A1/songs/"
}
```

Error:

HTTP/1.1 404 Not Found

```
{
  "detail": "Not found."
}
```

### Show list of genres

GET /juceer/0.1/genres/ HTTP/1.1

This feature returns the list of genres.

Resource URL: <http://127.0.0.1:8000/juceer/0.1/genres/>

### Examples:

Request 1:

http GET http://127.0.0.1:8000/juceer/0.1/genres/

Request 2:

http GET http://127.0.0.1:8000/juceer/0.1/genre/

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_terms": "8-bit"
    },
    {
      "artist_terms": "afrobeat"
    },
    ....
    {
      "artist_terms": "zydeco"
    }
  ],
  "title": "List of genres"
}
```

Error:

HTTP/1.1 404 Not Found

### Show list artists of genre

GET /juceer/0.1/genres/{artist\_terms}/artists/ HTTP/1.1

This feature returns the list of artists of the given genre.

Resource URL:

http://127.0.0.1:8000/juceer/0.1/genres/{artist\_terms}/artists/

### Examples:

Request 1:

http GET http://127.0.0.1:8000/juceer/0.1/genres/argentine\_rock/artists/

Request 2:

http GET http://127.0.0.1:8000/juceer/0.1/genres/nonsense/artists/

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_id": "ARVMPJC1187FB3B954",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARVMPJC1187FB3B954/",
      "artist_name": "Pe'z"
    }
  ],
  "title": "List of artists of argentine rock genre"
}
```

Error:

HTTP/1.1 404 Not Found

### Show info about artist

GET /juceer/0.1/artists/{artist\_id}/ HTTP/1.1

This feature returns the info about artists.

Resource URL: http://127.0.0.1:8000/juceer/0.1/artists/{artist\_id}/

### Examples:

Request 1:

http GET http://127.0.0.1:8000/juceer/0.1/artists/ARVMPJC1187FB3B954/

Request 2:

http GET http://127.0.0.1:8000/juceer/0.1/genres/nonsense/artists/

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object": {
    "artist_hottnesss": 0.431839301,
    "artist_id": "ARVMPJC1187FB3B954",
    "artist_name": "Pe'z",
    "artist_terms": "argentine_rock"
  },
  "songs_of_the_artist":
    "http://127.0.0.1:8000/juceer/0.1/artists/ARVMPJC1187FB3B954/songs/"
}
```

Error:

HTTP/1.1 404 Not Found



### Show all songs of the artist

GET /juceer/0.1/artists/{artist\_id}/songs/ HTTP/1.1

This feature returns the list of the songs of the artist.

Resource URL: [http://127.0.0.1:8000/juceer/0.1/artists/{artist\\_id}/songs/](http://127.0.0.1:8000/juceer/0.1/artists/{artist_id}/songs/)

### Examples:

Request 1:

http GET

<http://127.0.0.1:8000/juceer/0.1/artists/ARVMPJC1187FB3B954/songs/>

Request 2:

http GET <http://127.0.0.1:8000/juceer/0.1/genres/nonsense/artists/>

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_hottnesss": 0.431839301,
      "artist_id": "ARVMPJC1187FB3B954",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARVMPJC1187FB3B954/",
      "artist_name": "Pe'z",
      "artist_terms": "argentine_rock",
      "release_id": 163436,
      "song_hottnesss": -1.0,
      "song_id": "SOWHAEF12A8C13EF0C",
      "song_info":
"http://127.0.0.1:8000/juceer/0.1/songs/SOWHAEF12A8C13EF0C/",
      "song_year": 0
    }
  ],
  "title": "List of songs of Pe'z"
}
```

Error:

HTTP/1.1 404 Not Found

### Show info about song

GET /juceer/0.1/songs/{song\_id}/ HTTP/1.1

This feature returns the detail of the song.

Resource URL: http://127.0.0.1:8000/juceer/0.1/songs/{song\_id}/

### Examples:

Request 1:

http GET

http://127.0.0.1:8000/juceer/0.1/songs/SOABLAF12AB018E1D9/

Request 2:

http GET http://127.0.0.1:8000/juceer/0.1/songs/SOABLAF12AB018E1D/

Request 3:

http DELETE

http://127.0.0.1:8000/juceer/0.1/songs/SOOLVIB12A8C1408E2/

### Responses:

Success:

Request 1:

HTTP/1.1 200 OK

{

"artists\_of\_the\_same\_ganre":

"http://127.0.0.1:8000/juceer/0.1/genres/grunge/artists/",

"google\_artist": "https://www.google.com/?q=Alice In Chains/",

"object": {

"artist\_hottnesss": 0.582921577,

"artist\_id": "ARSW5F51187FB4CFC9",

"artist\_info":

"http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/",

"artist\_name": "Alice In Chains",

"artist\_terms": "grunge",

"release\_id": 583091,

"song\_hottnesss": 0.824432869,

"song\_id": "SOABLAF12AB018E1D9",

```
"song_info":  
"http://127.0.0.1:8000/juceer/0.1/songs/SOABLAF12AB018E1D9/",  
  "song_year": 2009  
},  
  "songs_of_the_same_artist":  
"http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/songs/",  
  "songs_of_the_same_year":  
"http://127.0.0.1:8000/juceer/0.1/years/2009/"  
}
```

Request 2:

HTTP/1.1 204 No Content

Error:

HTTP/1.1 404 Not Found

### Show all songs of the artist of the year

GET /juceer/0.1/artists/{ {artist\_id} }/songs/year/{ {song\_year} }/ HTTP/1.1

This feature returns the list of the songs of the artist of the given year.

Resource URL:

http://127.0.0.1:8000/juceer/0.1/artists/{ {artist\_id} }/songs/year/{ {song\_year} }/

### Examples:

Request 1:

http GET

http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/songs/year/1992/

Request 2:

http GET

http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/songs/year/1991/

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_hottnesss": 0.582921577,
      "artist_id": "ARSW5F51187FB4CFC9",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/",
      "artist_name": "Alice In Chains",
      "artist_terms": "grunge",
      "release_id": 385038,
      "song_hottnesss": 0.708746211,
      "song_id": "SOOLVIB12A8C1408E2",
```

```
    "song_info":  
    "http://127.0.0.1:8000/juceer/0.1/songs/SOOLVIB12A8C1408E2/",  
    "song_year": 1992  
  }  
],  
  "title": "List of songs of Alice In Chains in 1992"  
}
```

Error:

HTTP/1.1 404 Not Found

### Show all list of artists sorted by popularity with possibility to subset

GET /juceer/0.1/artists/hottest/{number}}/ HTTP/1.1

This feature returns the list of the songs of the artists sorted by popularity with possibility to subset.

Resource URL: <http://127.0.0.1:8000/juceer/0.1/artists/hottest/{number}}/>

### Examples:

Request 1:

http GET http://127.0.0.1:8000/juceer/0.1/artists/hottest/25/

Request 2:

http GET http://127.0.0.1:8000/juceer/0.1/artists/hottest/0/

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_id": "ARTDQRC1187FB4EFD4",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARTDQRC1187FB4EFD4/",
      "artist_name": "Black Eyed Peas"
    },
    {
      "artist_id": "ARKU3Z61187FB51DCA",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARKU3Z61187FB51DCA/",
      "artist_name": "Rihanna"
    },
    ...
    {
      "artist_id": "ARFWL8S1187B9B4B44",
```

```
    "artist_info":  
    "http://127.0.0.1:8000/juceer/0.1/artists/ARFWL8S1187B9B4B44/",  
    "artist_name": "CA(c)line Dion"  
  }  
],  
  "title": "List of artists by popularity"  
}
```

Error:

HTTP/1.1 404 Not Found



### Show all list of artists sorted by popularity for given year with possibility to subset

GET /juceer/0.1/artists/hottest/{number}/year/{song\_year}/ HTTP/1.1

This feature returns the list of the artists sorted by popularity for given year with possibility to subset.

Resource URL:

http://127.0.0.1:8000/juceer/0.1/artists/hottest/{number}/year/{song\_year}/

### Examples:

Request 1:

http GET http://127.0.0.1:8000/juceer/0.1/artists/hottest/25/year/2010/

Request 2:

http GET http://127.0.0.1:8000/juceer/0.1/artists/hottest/0/year/2011/

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_id": "ARVBP9K1187FB48254",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARVBP9K1187FB48254/",
      "artist_name": "Cancer Bats"
    },
    {
      ...
    },
    {
      "artist_id": "ARDDHGI126E2B3AA76",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARDDHGI126E2B3AA76/",
      "artist_name": "Pascal Sangla"
    }
  ]
}
```

```
],  
  "title": "List of artists by popularity"  
}
```

Error:

HTTP/1.1 404 Not Found

### Show statistics of artist

GET /juceer/0.1/statistics/{type\_of\_stat}/artists/{artist\_id}/ HTTP/1.1

This feature returns the statistic of artist.

Resource URL:

http://127.0.0.1:8000/juceer/0.1/statistics/{type\_of\_stat}/artists/{artist\_id}/

### Examples:

Request 1:

http GET

http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARSW5F51187FB4CFC9/

Request 2:

http GET

http://127.0.0.1:8000/juceer/0.1/statistics/median/artists/ARSW5F51187FB4CFC9/

Request 3:

http GET

http://127.0.0.1:8000/juceer/0.1/statistics/std\_dev/artists/ARSW5F51187FB4CFC9/

Request 4:

http GET

http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARS5F51187FB4CFC9/

### Responses:

Success:

Request 1:

HTTP/1.1 200 OK

```
{
  "mean": 0.76658954,
  "median":
"http://127.0.0.1:8000/juceer/0.1/statistics/median/artists/ARSW5F51187FB4CFC9/",
  "object": {},
  "songs_of_the_same_artist":
"http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/songs/",
  "std_deviation":
"http://127.0.0.1:8000/juceer/0.1/statistics/std_dev/artists/ARSW5F51187FB4CFC9/",
  "year_by_year_mean":
"http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARSW5F51187FB4CFC9/years/"
}
Request 2:
```

HTTP/1.1 200 OK

```
{
  "mean":
"http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARSW5F51187FB4CFC9/",
  "median": 0.76658954,
  "object": {},
  "songs_of_the_same_artist":
"http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/songs/",
  "std_deviation":
"http://127.0.0.1:8000/juceer/0.1/statistics/std_dev/artists/ARSW5F51187FB4CFC9/",
  "year_by_year_mean":
"http://127.0.0.1:8000/juceer/0.1/statistics/median/artists/ARSW5F51187FB4CFC9/years/"
}
```

Request 3:

HTTP/1.1 200 OK

```
{
  "mean":
"http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARSW5F51187FB4CFC9/",
  "median":
"http://127.0.0.1:8000/juceer/0.1/statistics/median/artists/ARSW5F51187FB4CFC9/",
  "object": {},
  "songs_of_the_same_artist":
"http://127.0.0.1:8000/juceer/0.1/artists/ARSW5F51187FB4CFC9/songs/",
  "std_deviation": "0.08180282036460895",
  "year_by_year_mean":
"http://127.0.0.1:8000/juceer/0.1/statistics/std_dev/artists/ARSW5F51187FB4CFC9/years/"
}
```

Error:

Request 4:

HTTP/1.1 404 Not Found

### Show statistics of artists for given year

GET /juceer/0.1/statistics/mean/artists/{artist\_id}/years/{song\_year}/  
HTTP/1.1

This feature returns the statistic of artist for given year.

Resource URL:

http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/{artist\_id}/years/{song\_year}/

### Examples:

Request 1:

http GET

http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARVBP9K1187FB48254/years/2010/

Request 2:

http GET

http://127.0.0.1:8000/juceer/0.1/statistics/mean/artists/ARVBP9K1187FB48254/years/209/

### Responses:

Success:

Request 1:

HTTP/1.1 200 OK

```
{
  "artist_id": "ARVBP9K1187FB48254",
  "artist_name": "Cancer Bats",
  "mean": 0.726267401,
  "median":
"http://127.0.0.1:8000/juceer/0.1/statistics/median/artists/ARVBP9K1187F
B48254/",
  "object": {},
  "songs_of_the_same_artist":
"http://127.0.0.1:8000/juceer/0.1/artists/ARVBP9K1187FB48254/songs/",
  "stat": [
```

```
    2010
  ],
  "std_deviation":
"http://127.0.0.1:8000/juceer/0.1/statistics/std_dev/artists/ARVBP9K1187F
B48254/",
  "year": "2010"
}
```

Error:

HTTP/1.1 404 Not Found

## Show list of songs with matching parameter

GET

/juceer/0.1/statistics/mean/artists/{ {artist\_id} }/years/{ {song\_year} }.json

HTTP/1.1

This feature returns the list of songs which artist\_name, artist\_term or artist\_year contain given query.

Resource URL: <http://127.0.0.1:8000/juceer/0.1/search/?q={{query}}>

### Examples:

Request 1:

http GET <http://127.0.0.1:8000/juceer/0.1/search/?q=black>

Request 2:

http GET <http://127.0.0.1:8000/juceer/0.1/search/?q=>

### Responses:

Success:

HTTP/1.1 200 OK

```
{
  "object_list": [
    {
      "artist_hottnesss": 0.408801073,
      "artist_id": "AR4CH531187FB4EA21",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/AR4CH531187FB4EA21/",
      "artist_name": "Abysmal Dawn",
      "artist_terms": "blackened_death_metal",
      "release_id": 237545,
      "song_hottnesss": 0.538200452,
      "song_id": "SOMTZYN12A8AE471C0",
      "song_info":
"http://127.0.0.1:8000/juceer/0.1/songs/SOMTZYN12A8AE471C0/",
      "song_year": 2008
    }
  ]
}
```



```
    },
    {
      "artist_hottnesss": 0.460853145,
      "artist_id": "ARV1RMI1187FB48F94",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARV1RMI1187FB48F94/",
      "artist_name": "Arkona",
      "artist_terms": "black_metal",
      "release_id": 573497,
      "song_hottnesss": -1.0,
      "song_id": "SONQTUH12AF72A47A7",
      "song_info":
"http://127.0.0.1:8000/juceer/0.1/songs/SONQTUH12AF72A47A7/",
      "song_year": 0
    },
    ...
    {
      "artist_hottnesss": 0.262798505,
      "artist_id": "ARO2EDF1187B993BD8",
      "artist_info":
"http://127.0.0.1:8000/juceer/0.1/artists/ARO2EDF1187B993BD8/",
      "artist_name": "The Black Velvet Band",
      "artist_terms": "irish_folk",
      "release_id": 570453,
      "song_hottnesss": -1.0,
      "song_id": "SOAMXQV12AB018775F",
      "song_info":
"http://127.0.0.1:8000/juceer/0.1/songs/SOAMXQV12AB018775F/",
      "song_year": 0
    }
  ],
  "title": "You searched for black"
}
```

Request 2:

HTTP/1.1 200 OK

```
{  
  "object_list": [],  
  "title": "You search is empty"  
}
```

## **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"