

My REST APIs

Why I built my REST API

My api manages and allows viewing of a music related database.

All api's will have a simple example request and then give the actual output that the html test scripts for each outputted.

I would also like to add I did not add any bearer token to any of my API's, but I did add password hashing to employees and users.

My APIs

1 - Album Manager

- **GET /albums**
 - Gets all albums
 - If we did GET /albums we would get the response:

```
{"success":true,"data":[{"id":1,"albumName":"OK computer","artistName":"RadioHead","numOfSongs":12}, {"id":2,"albumName":"Bad Contestant","artistName":"Matt Maltese","numOfSongs":11}],"count":2}
```
- **GET /albums/{id}**
 - Gets a single album by its id
 - If we do /albums/1 we'd get

- **POST /albums**

- Creates a new album
- if we did POST /albums with albumName Bewitched, artistName Laufey and numOfSongs 18

```
{"success": true, "message": "Album created Successfully",  
  "data": {"id": 4,  
            "albumName": "Bewitched", "artistName": "Laufey", "numOfSongs": 18}}
```

- **PUT /albums/{id}**

- Updates a album by its id
- If we did PUT /albums/4 with albumName Bewitched, artistName Laufey and numOfSongs 17

```
{"success": true, "message": "Album updates successfully"}
```

- **DELETE /albums/{id}**
 - Deletes an album by id
 - If we did DELETE /albums/4 we'd get
`{"success": true, "message": "Album deleted successfully"}`

2 - Artists Manager

- GET /artists

- Gets all artists

```
{"success": true, "data": [ {"id": 1, "name": "artist1", "agency": "agency1", "genre":  
"pop"}, {"id": 2, "name": "artist2", "agency": "agency2", "genre": "rock"} ], "count":  
2}
```

- GET /artists/{id}

- Gets a single artist by id
- GET /artists/1

```
{"success": true, "data": {"id": 1, "name": "artist1", "agency": "agency1", "genre":  
"pop"}}
```

- **POST /artists**

- Creates a new artist
- POST /artists with name artist3, agency3 and hiphop genre
{"success": true, "message": "artist created Successfully", "data": {"id": "2", "name": "artist2", "agency": "agency2", "genre": "rock"}}

- **PUT /artists/{id}**

- Updates a artist by id
- We want to update artist 2 with agency4 PUT /artists/2
{"success": true, "message": "artist updates successfully"}

- **DELETE /artists/{id}**
 - Deletes a artist by id
 - We want to delete artist 2 DELETE /artists/2
{"success": true, "message": "artist deleted successfully"}

3 - Audiobook Manager

- **GET /audiobooks**

- Gets all audiobooks
- if we do GET /audiobooks:

```
{"success": true, "data": [{"id": 1, "bookName": "", "writerName": "Author1", "readerName": "reader1"}], "count": 1}
```

- **GET /audiobooks/{id}**

- Gets a single audiobook by id
- We do GET /audiobook/1

```
{"success": true, "data": {"id": 1, "bookName": "", "writerName": "Author1", "readerName": "reader1"}}
```

- **POST /audiobooks**

- Create a new audiobook
- We want to create a book Hunger Games 2, Author3, and reader2
`{"success": true, "message": "audiobook created Successfully", "data": {"id": "2", "bookName": "", "writerName": "Author3", "readerName": "reader2"}}`

- **PUT /audiobooks/{id}**

- Updates a audiobook
- If we do PUT /audiobooks/1 with update writerName to Author2
`{"success": true, "message": "audiobook updates successfully"}`

- **DELETE /audiobooks/{id}**
 - Deletes a audiobook by id
 - If we want to DELETE /audioboks/2
{"success": true, "message": "Audiobook deleted successfully"}

4 - Employee Manager

- **GET /employees**

- Gets all employees

```
{"success": true, "data": [{"id": 1, "name": "emp1", "email": "gmail1", "department": "dep1"}], "count": 1}
```

- **GET /employees{id}**

- Gets a single employee

- Getting /employees/1

```
{"success": true, "data": {"id": 1, "name": "emp1", "email": "gmail1", "department": "dep1"}}
```

- **POST /employees**

- Creates a new employee
- New employee emp2, email emp2@email, department dept, and password pass34

```
{"success": true, "message": "employee created Successfully", "data": {"id": "2", "name": "emp2", "email": "emp2@email", "department": "dept2"}}
```

- **PUT /employees/{id}**

- Updates an employee
- Update employee 2 to have dept3 for department

```
{"success": true, "message": "employee updates successfully"}
```

- **DELETE /employees/{id}**
 - Deletes a employee
 - We want to DELETE /employees/2
 - `{"success": true, "message": "employee deleted successfully"}`

5 - Live Event Manager

- **GET /events**

- Gets all live events
- {"success": true, "data": [{"id": 1, "venue": "venue1", "artist": "artist1", "day": "monday"}], "count": 1}

- **GET /events{id}**

- Get a single live event
- We want event 1 GET /events/1
{"success": true, "data": {"id": 1, "venue": "venue1", "artist": "artist1", "day": "monday"}}
}

- **POST /events**

- Creates a new live event
- I could not get this part to work in my html test, im not sure why

- **PUT /events/{id}**

- Updates a live event by id
- We want to update event 1 by changing day to tuesday
{"success": true, "message": "event updates successfully"}

- **DELETE /events/{id}**
 - Deletes a live event
 - We want to delete event 1 DELETE /events/1
{"success": true, "message": "event deleted successfully"}

6 - Playlist Manager

- **GET /playlists**

- Gets all playlists
- We do GET /playlists and get the response

```
{"success": true, "data": [{"id": 1, "userName": "user1", "playlistName": "playlist1", "genre": "pop"}, {"id": 2, "userName": "user2", "playlistName": "playlist2", "genre": "rock"}], "count": 2}
```

- **GET /playlists/{id}**

- Get a single playlist
- If we wanted only the first playlist GET /playlist/1

```
{"success": true, "data": {"id": 1, "userName": "user1", "playlistName": "playlist1", "genre": "pop"}}
```

- **POST /playlists**

- Creates a new playlist
- /POST playlists with userName user1, playlistName playlist1, and genre pop
{"success": true, "message": "User created Successfully", "data": {"id": "1", "userName": "user1", "playlistName": "playlist1", "genre": "pop"}}

- **PUT /playlists/{id}**

- Updates a playlist by id
- If we wanted to update genre of playlist 2 with hiphop PUT /playlists/2
{"success": true, "message": "Playlist updated successfully"}

- **DELETE /playlist/{id}**
 - Deletes a playlist by id
 - We want to delete playlist 2 DELETE /playlists/2
{"success": true, "message": "Playlist deleted successfully"}

7 - Podcast Manager

- GET /podcasts

- Gets all podcasts

```
{"success": true, "data": [{ "id": 1, "name": "podcast1", "hostName":  
"host1", "totalTime": 30}, {"id": 2, "name": "podcast2", "hostName":  
"host2", "totalTime": 12}], "count": 2}
```

- GET /podcasts/{id}

- Gets a single podcast
- We want to GET /podcasts/1

```
{"success": true, "data": {"id": 1, "name": "podcast1", "hostName":  
"host1", "totalTime": 30}}
```

- **POST /podcasts**

- Create a new podcast
- We want to create podcast1, host1, and total time of 30
`{"success": true, "message": "podcast created Successfully", "data": {"id": "1", "name": "podcast1", "hostName": "host1", "totalTime": 30 }}`

- **PUT /podcasts/{id}**

- Updates a podcast by id
- We want to update podcast 1 with a new
`{"success": true, "message": "Podcast updated successfully"}`

- **DELETE /podcasts/{id}**
 - Deletes a podcast by id
 - We want to delete podcast 2 DELETE /podcasts/2
{"success": true, "message": "Podcast deleted successfully"}

8 - Top Artists Manager

- **GET /topartists**

- Gets all top artists

```
{"success": true, "data": [{"id": 1, "name": "artist1", "genre": "pop", "rank": 1}, {"id": 2, "name": "artist2", "genre": "rock", "rank": 2}], "count": 2}
```

- **GET /topartists/{id}**

- Get a single top artist

- We want to get top artist 2 GET /topartists/2

```
{"success": true, "data": {"id": 2, "name": "artist2", "genre": "rock", "rank": 2}}
```


- **POST /topartists**

- Create a new top artist
- We want to create the artist1, genre pop, and rank 1

```
{"success": true, "message": "artist created Successfully", "data": {"id": "1", "name": "artist1", "genre": "pop", "rank": 1}}
```

- **PUT /topartists/{id}**

- Updates a top artist
- We want to update a top artists name to Jenny PUT /topartists/1

```
{"success": true, "message": "Topartist updated successfully"}
```

- **DELETE /topartists/{id}**
 - Deletes a top artist
 - We want to delete top artist 1 DELETE /topartists/1
{"success": true, "message": "Topartist deleted successfully"}

9 - Top Songs Manager

- **GET /topsongs**

- Gets all top songs

```
{"success": true, "data": [{"id": 1, "name": "song1", "artist": "artist1", "rank": 1}, {"id": 2, "name": "song2", "artist": "artist2", "rank": 2}], "count": 2}
```

- **GET /topsongs/{id}**

- Gets a top song

- GET /topsongs/2

```
{"success": true, "data": {"id": 2, "name": "song2", "artist": "artist2", "rank": 2}}
```

- **POST /topsongs**

- Creates a new top song
- We want to create the first top song: POST /topsongs song1, artist1, and rank 1

```
{"success": true, "message": "Top Song created Successfully", "data": {"id": "1", "name": "song1", "artist": "artist1", "rank": 1}}
```

- **PUT /topsongs/{id}**

- Updates a new top song
- We want to update top songs name to song3 PUT /topsongs/2

```
{"success": true, "message": "Topsong updated successfully"}
```

- **DELETE /topsongs/{id}**
 - Deletes a top song
 - We want to delete top song 1 DELETE /topsongs/1
{"success": true, "message": "Top Song deleted successfully"}

10 - Users Manager

- GET /users

- Gets all users

```
{"success": true, "data": [{"id": 1, "name": "user1", "email": "email1@gmail.com"}, {"id": 2, "name": "user2", "email": "email2@email"}], "count": 2}
```

- GET /users/{id}

- Get a single user

- GET /users/1

```
{"success": true, "data": {"id": 1, "name": "user1", "email": "email1@gmail.com"}}
```

- **POST /users**


- Create a new user
- Create new user2 with email2@email and pass78
{"success": true, "message": "User created Successfully", "data": {"id": "2", "name": "user2", "email": "email2@email"}}

- **PUT /users/{id}**

- Updates a user by id
- Update user2 to have email3@email
{"success": true, "message": "User updates successfully"}

- **DELETE /users/{id}**
 - Deletes a user by id
 - DELETE /users/2
{"success": true, "message": "User deleted successfully"}


HTML test working for employees

 alt text



alt text

HTML test working for users

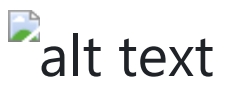
 alt text




alt text

HTML test for albums

 alt text




HTML test for Artists

 alt text



alt text


HTML Tst for audiobooks

 alt text



alt text

HTML for Live Events

 alt text

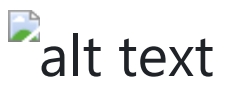


alt text

HTML for playlists



alt text




HTML for podcasts

 alt text



alt text

HTML for Top Artists

 alt text



alt text

HTML for top Songs

 alt text



alt text

