## Back-end Engineer.

## Notes on my task.

The code is in the musicbrainz\_api\_func.py . First, make sure you have requests installed with pip install requests and python 3.6+ (I used python 3.9)

The function is run as a python file. In terminal run python musicbrainz\_api\_func.py or if you using an IDE run it.

In the terminal you will prompted with inputs and you will be asked to enter the values.

> Enter limit value:(enter int value of your choice) Enter offset value:(enter int value of your choice) Enter artist id:(enter the artist id)

The limit is for pagination, from the required 50 to 150, more or less.

as shown below:

```
(projects-django) (base) 10:42[nyaribari@foscraft BMAT-APIS]$ python musicbrainz_api_func.py
Enter limit value :5
Enter offset value :0
Enter artist id :65f4f0c5-ef9e-490c-aee3-909e7ae6b2ab
```

Then the result should show

```
10:42[nyaribari@foscraft BMAT-APIS]$ python musicbrainz_api_func.py
. (id:: '0da580f2-6768-498f-af9d-2becaddf15e0', 'title': 'Ride the Lightning', 'year': '1984-07-30',
a8e1-b4bfc4d66913', 'title': 'Master of Puppets', 'year': '1986-02-21', 'release count': 807}, ('id'
'.And Justice for All', 'year': '1988-08-25', 'release_count': 807), ('id': 'e8f70201-8899-370c-9e0
191-08-12', 'release_count': 807}, ('id': 'f44f4f73-a714-31a1-a4b8-bfcaaf311f50', 'title': 'Kill 'Em
```

In the example above, I used  ${\bf 5}$  as the  ${\bf limit},\ {\bf 0}$  as the  ${\bf offset}$  and the 65f4f0c5-ef9e-490c-aee3-909e7ae6b2ab. as the id The code should give the anticipated result.

Shown below is for the 100 pagination limit: