

- **To run the code :**

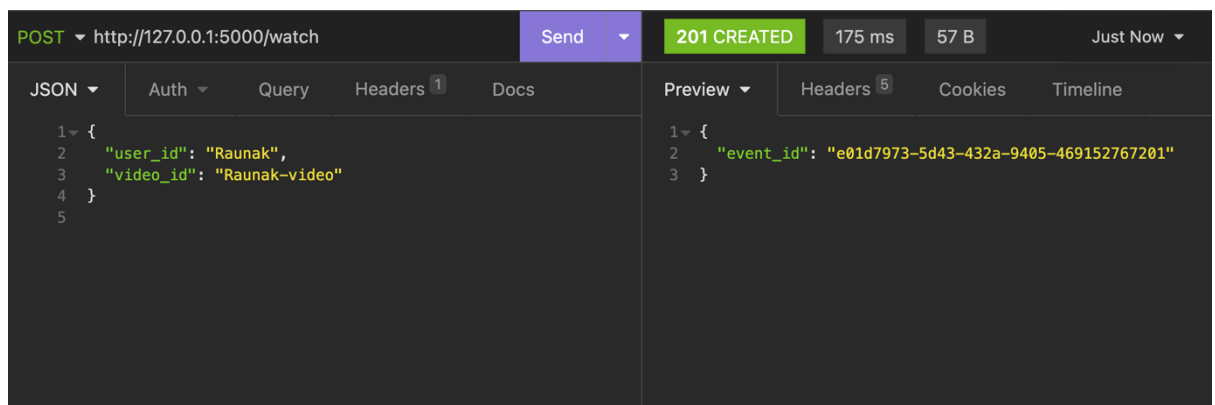
- aws configure [Put AWS access and secret key to communicate DynamoDB]
- python -m venv venv
- source venv/bin/activate
- pip install -r requirements.txt
- python app.py

- **To Test the API :**

- 1) **POST /watch:**

- URL: http://127.0.0.1:5000/watch
 - Body (JSON):

```
{  
  "user_id": "user1",  
  "video_id": "video1"  
}
```



2) Check watch-event record store in the DynamoDB

The screenshot shows the AWS DynamoDB console interface. On the left, a list of tables is displayed, with 'WatchEvents' selected. The main panel shows the 'Scan or query items' section, where 'Scan' is chosen. The table 'Table - WatchEvents' is selected, and 'All attributes' are projected. The 'Run' button is clicked, and the results are displayed in a table.

event_id (String)	timestamp	user_id	video_id
e01d7973-5d43-432a...	2024-05-2...	Raunak	Raunak-video
76e981e7-d237-495d...	2024-05-2...	sanjeev	sanjeevvideo1
1dea12e5-43c3-4610...	2024-05-2...	user1	video1
6804e2fe-0995-44bd...	2024-05-2...	user1	video1

3) GET /watch-history:

- URL: http://127.0.0.1:5000/watch-history?user_id=Raunak

The screenshot shows a web browser window with the URL `http://127.0.0.1:5000/watch-history?user_id=Raunak`. The status bar indicates a 200 OK response with a 44 ms response time and 174 B of data. The 'Preview' tab shows the JSON response:

```
[
  {
    "event_id": "e01d7973-5d43-432a-9405-469152767201",
    "timestamp": "2024-05-26T14:37:21.574113",
    "user_id": "Raunak",
    "video_id": "Raunak-video"
  }
]
```