producer

November 17, 2023

[24]:

```
from kafka import KafkaProducer
from googleapiclient.discovery import build
import json
from datetime import datetime, timedelta
import isodate

# Set up the YouTube API client
api_key ="AIzaSyBM67yGeB8HtMCOkZ71D1c_K5Qg9VGvjN4"
youtube = build('youtube', 'v3', developerKey=api_key)

# Set up Kafka producer
bootstrap_servers = 'localhost:9092'
topic = 'yt7'
producer = KafkaProducer(bootstrap_servers=bootstrap_servers)

# Define the end time which is the current time
end_time = datetime.now()

# Define the start time whihc is 30 minutes before the end time
start_time = end_time - timedelta(minutes=30)

# Convert time to YouTube API format
start_time_str = start_time.isoformat('T') + 'Z'
end_time_str = end_time.isoformat('T') + 'Z'
```

[34]:

```
def get_video_categories():

    dt={}
    lt=[]
    request = youtube.videoCategories().list(
        part='snippet',
        regionCode='US'
    )
    response = request.execute()

    for item in response['items']:
        category_id = item['id']
```

```python
        lt.append(category_id)
        category_title = item['snippet']['title']
        dt[category_id]=category_title
        print(f"Category ID: {category_id}")
        print(f"Category Title: {category_title}")
        print()

    return dt,lt

category_dict,category_list=get_video_categories()
```

```
Category ID: 1
Category Title: Film & Animation

Category ID: 2
Category Title: Autos & Vehicles

Category ID: 10
Category Title: Music

Category ID: 15
Category Title: Pets & Animals

Category ID: 17
Category Title: Sports

Category ID: 18
Category Title: Short Movies

Category ID: 19
Category Title: Travel & Events

Category ID: 20
Category Title: Gaming

Category ID: 21
Category Title: Videoblogging

Category ID: 22
Category Title: People & Blogs

Category ID: 23
Category Title: Comedy

Category ID: 24
Category Title: Entertainment
Category ID: 25
Category Title: News & Politics

Category ID: 26
Category Title: Howto & Style

Category ID: 27
Category Title: Education

Category ID: 28
Category Title: Science & Technology

Category ID: 29
Category Title: Nonprofits & Activism

Category ID: 30
Category Title: Movies

Category ID: 31
Category Title: Anime/Animation

Category ID: 32
Category Title: Action/Adventure

Category ID: 33
Category Title: Classics

Category ID: 34
Category Title: Comedy

Category ID: 35
Category Title: Documentary

Category ID: 36
Category Title: Drama

Category ID: 37
Category Title: Family

Category ID: 38
Category Title: Foreign

Category ID: 39
Category Title: Horror

Category ID: 40
Category Title: Sci-Fi/Fantasy
Category ID: 41
Category Title: Thriller

Category ID: 42
Category Title: Shorts

Category ID: 43
Category Title: Shows

Category ID: 44
Category Title: Trailers
```

[6]:

```
def get_video_info(vid_id,category,api):

    api_key=api
    youtube = build('youtube', 'v3', developerKey=api_key)
    next_page_token = None

    while True:
        request = youtube.search().list(
            part='snippet',
            type='video',
            eventType='completed',
            publishedAfter=start_time_str,
            publishedBefore=end_time_str,
            pageToken=next_page_token,
            videoCategoryId=vid_id
        )
        response = request.execute()

        for item in response['items']:
            video_id = item['id']['videoId']
            video_title = item['snippet']['title']
            videos_cat=category
            video_request = youtube.videos().list(
                part='contentDetails',
                id=video_id
            )
            video_request = youtube.videos().list(
                part='contentDetails, statistics',
                id=video_id
            )
            video_response = video_request.execute()
            video_duration =␣
 ↪video_response['items'][0]['contentDetails']['duration']
            views = video_response['items'][0]['statistics'].get('viewCount', 0)
            if 'likeCount' in video_response['items'][0]['statistics']:
                likes = video_response['items'][0]['statistics']['likeCount']
            else:
                likes = 0
            duration = isodate.parse_duration(video_duration)
            duration= duration.total_seconds() / 60
            published_time = item['snippet']['publishedAt']


            video_info = {
                'video_id': video_id,
                'video_title': video_title,
                "duration":duration,
                "views":views,
                "likes":likes,
                "category": videos_cat,
                'published_time': published_time
            }
            producer.send(topic, json.dumps(video_info).encode('utf-8'))

        producer.flush()

        next_page_token = response.get('nextPageToken')

        if not next_page_token:
            break
```

[45]:

```
for j in [1,2,10,15,17,19]:
    i=str(j)
    print("Sending ",i,category_dict[i],"data")
    get_video_info(i,category_dict[i],"AIzaSyCYlRHo3ExW51-ZuzKstE1atYN4SV-xKRE")
```

```
Sending  1 Film & Animation data
Sending  2 Autos & Vehicles data
Sending  10 Music data
Sending  15 Pets & Animals data
Sending  17 Sports data
Sending  19 Travel & Events data
```

[ ]:

```

```