In [1]:

```
import datetime
import time
import spotipy
import pandas as pd
import numpy as np
from spotipy.oauth2 import SpotifyClientCredentials
import logging

client_credentials_manager = SpotifyClientCredentials(client_id='c90e60e8ffa54b409e84d8
e8c55b7ad2',
                                                      client_secret='982ee7b8e5684d9e97
52e88c48c22764')
sp = spotipy.Spotify(client_credentials_manager=client_credentials_manager)
```

In [2]:

```
country_codes = {"AD": "Andorra", "AE": "United Arab Emirates", "AF": "Afghanistan", "AG": "Antigua and Barbuda",
                 "AI": "Anguilla", "AL": "Albania", "AM": "Armenia", "AO": "Angola", "AQ": "Antarctica",
                 "AR": "Argentina", "AS": "American Samoa", "AT": "Austria", "AU": "Australia", "AW": "Aruba",
                 "AX": "Åland Islands", "AZ": "Azerbaijan", "BA": "Bosnia and Herzegovina", "BB": "Barbados",
                 "BD": "Bangladesh", "BE": "Belgium", "BF": "Burkina Faso", "BG": "Bulgaria", "BH": "Bahrain",
                 "BI": "Burundi", "BJ": "Benin", "BL": "Saint Barthélemy", "BM": "Bermuda", "BN": "Brunei Darussalam",
                 "BO": "Bolivia: Plurinational State of", "BQ": "Bonaire: Sint Eustatius and Saba", "BR": "Brazil",
                 "BS": "Bahamas", "BT": "Bhutan", "BV": "Bouvet Island", "BW": "Botswana", "BY": "Belarus",
                 "BZ": "Belize", "CA": "Canada", "CC": "Cocos (Keeling) Islands",
                 "CD": "Congo: the Democratic Republic of", "CF": "Central African Republic", "CG": "Congo",
                 "CH": "Switzerland", "CI": "Côte d'Ivoire", "CK": "Cook Islands", "CL": "Chile", "CM": "Cameroon",
                 "CN": "China", "CO": "Colombia", "CR": "Costa Rica", "CU": "Cuba", "CV": "Cabo Verde", "CW": "Curaçao",
                 "CX": "Christmas Island", "CY": "Cyprus", "CZ": "Czech Republic", "DE": "Germany", "DJ": "Djibouti",
                 "DK": "Denmark", "DM": "Dominica", "DO": "Dominican Republic", "DZ": "Algeria", "EC": "Ecuador",
                 "EE": "Estonia", "EG": "Egypt", "EH": "Western Sahara", "ER": "Eritrea", "ES": "Spain",
                 "ET": "Ethiopia", "FI": "Finland", "FJ": "Fiji", "FK": "Falkland Islands (Malvinas)",
                 "FM": "Micronesia: Federated States of", "FO": "Faroe Islands", "FR": "France", "GA": "Gabon",
                 "GB": "United Kingdom of Great Britain and Northern Ireland", "GD": "Grenada", "GE": "Georgia",
                 "GF": "French Guiana", "GG": "Guernsey", "GH": "Ghana", "GI": "Gibraltar", "GL": "Greenland",
                 "GM": "Gambia", "GN": "Guinea", "GP": "Guadeloupe", "GQ": "Equatorial Guinea", "GR": "Greece",
                 "GS": "South Georgia and the South Sandwich Islands", "GT": "Guatemala", "GU": "Guam",
                 "GW": "Guinea-Bissau", "GY": "Guyana", "HK": "Hong Kong", "HM": "Heard Island and McDonalds Islands",
                 "HN": "Honduras", "HR": "Croatia", "HT": "Haiti", "HU": "Hungary", "ID": "Indonesia", "IE": "Ireland",
                 "IL": "Israel", "IM": "Isle of Man", "IN": "India", "IO": "British Indian Ocean Territory",
                 "IQ": "Iraq", "IR": "Iran: Islamic Republic of", "IS": "Iceland", "IT": "Italy", "JE": "Jersey",
                 "JM": "Jamaica", "JO": "Jordan", "JP": "Japan", "KE": "Kenya", "KG": "Kyrgyzstan", "KH": "Cambodia",
                 "KI": "Kiribati", "KM": "Comoros", "KN": "Saint Kitts and Nevis",
                 "KP": "Korea: Democratic People's Republic of", "KR": "Korea: Republic of", "KW": "Kuwait",
                 "KY": "Cayman Islands", "KZ": "Kazakhstan", "LA": "Lao People's Democratic Republic", "LB": "Lebanon",
                 "LC": "Saint Lucia", "LI": "Liechtenstein", "LK": "Sri Lanka", "LR": "Liberia", "LS": "Lesotho",
                 "LT": "Lithuania", "LU": "Luxembourg", "LV": "Latvia", "MA": "Morocco"
, "MC": "Monaco",
                 "MD": "Moldova: Republic of", "ME": "Montenegro", "MF": "Saint Martin
 (French part)",
                 "MG": "Madagascar", "MH": "Marshall Islands", "MK": "Macedonia: the fo
rmer Yugoslav Republic of",
                 "ML": "Mali", "MM": "Myanmar", "MN": "Mongolia", "MO": "Macao", "MP":
"Northern Mariana Islands",
                 "MQ": "Martinique", "MR": "Mauritania", "MS": "Montserrat", "MT": "Mal
ta", "MU": "Mauritius",
                 "MV": "Maldives", "MW": "Malawi", "MX": "Mexico", "MY": "Malaysia", "M
Z": "Mozambique",
                 "NA": "Namibia", "NC": "New Caledonia", "NE": "Niger", "NF": "Norfolk
 Island", "NG": "Nigeria",
                 "NI": "Nicaragua", "NL": "Netherlands", "NO": "Norway", "NP": "Nepal",
"NR": "Nauru", "NU": "Niue",
                 "NZ": "New Zealand", "OM": "Oman", "PA": "Panama", "PE": "Peru", "PF":
"French Polynesia",
                 "PG": "Papua New Guinea", "PH": "Philippines", "PK": "Pakistan", "PL":
"Poland",
                 "PM": "Saint Pierre and Miquelon", "PN": "Pitcairn", "PR": "Puerto Ric
o", "PS": "Palestine: State of",
                 "PT": "Portugal", "PW": "Palau", "PY": "Paraguay", "QA": "Qatar", "RE"
: "Réunion", "RO": "Romania",
                 "RS": "Serbia", "RU": "Russian Federation", "RW": "Rwanda", "SA": "Sau
di Arabia",
                 "SB": "Solomon Islands", "SC": "Seychelles", "SD": "Sudan", "SE": "Swe
den", "SG": "Singapore",
                 "SH": "Saint Helena: Ascension and Tristan da Cunha", "SI": "Slovenia"
, "SJ": "Svalbard and Jan Mayen",
                 "SK": "Slovakia", "SL": "Sierra Leone", "SM": "San Marino", "SN": "Sen
egal", "SO": "Somalia",
                 "SR": "Suriname", "SS": "South Sudan", "ST": "Sao Tome and Principe",
"SV": "El Salvador",
                 "SX": "Sint Maarten (Dutch part)", "SY": "Syrian Arab Republic", "SZ":
"Swaziland",
                 "TC": "Turks and Caicos Islands", "TD": "Chad", "TF": "French Southern
Territories", "TG": "Togo",
                 "TH": "Thailand", "TJ": "Tajikistan", "TK": "Tokelau", "TL": "Timor-Le
ste", "TM": "Turkmenistan",
                 "TN": "Tunisia", "TO": "Tonga", "TR": "Turkey", "TT": "Tuvalu", "TW":
"Taiwan: Province of China",
                 "TZ": "Tanzania: United Republic of", "UA": "Ukraine", "UG": "Uganda",
                 "UM": "United States Minor Outlying Islands", "US": "United States of
 America", "UY": "Uruguay",
                 "UZ": "Uzbekistan", "VA": "Holy See", "VC": "Saint Vincent and the Gre
nadines",
                 "VE": "Venezuela: Bolivarian Republic of", "VG": "Virgin Islands: Brit
ish",
                 "VI": "Virgin Islands: U.S.", "VN": "Viet Nam", "VU": "Vanuatu", "WF":
"Wallis and Futuna",
                 "WS": "Samoa", "YE": "Yemen", "YT": "Mayotte", "ZA": "South Africa",
"ZM": "Zambia", "ZW": "Zimbabwe",
                 "GLOBAL":"global"}
```

In [3]:

```
top_songs_countries = pd.read_csv('spotipy_countries.csv', encoding='utf-8').sample(100
0)
```

extracting the track_ids:

In [4]:

```
top_songs_countries['track_id'] = top_songs_countries.URL.apply(lambda x: str(x).split(
'/')[-1])
top_songs_countries.head()
```

Out[4]:

	Position	Track Name	Artist	Streams	
3063661	19	Ciao Adios	Anne-Marie	209858	https://open.spotify.com/track/3EfugazgSddQ
489606	8	Attention	Charlie Puth	128820	https://open.spotify.com/track/4iLqG9SeJSnt
3039552	6	Despacito (Featuring Daddy Yankee)	Luis Fonsi	1687	https://open.spotify.com/track/4aWmUDTfIPGks
2087299	140	My Way	Calvin Harris	2714	https://open.spotify.com/track/1vvNmPOiUuyCb
602802	4	Too Good At Goodbyes	Sam Smith	17178	https://open.spotify.com/track/1mXVgsBdtIVe

summing the streams per country for all songs:

In [5]:

```
summed_streams = top_songs_countries.groupby(['track_id','Track Name','Artist','Region'
])['Streams'].sum().reset_index()
summed_streams.sample(3)
```

Out[5]:

	track_id	Track Name	Artist	Region	Streams
47	0JOUrhZGMCx3lk9VosAQWl	El Desorden	Ozuna	do	1076
532	4COGASZEbAxmH6k5gYMo6G	0 Sentimientos (Remix) [feat. Noriel, Darkiel,...	Jon Z	ec	2116
859	6m2LNopVJKsvBB9l7Z1rwn	Olha a Explosão	MC Kevinho	pa	5152

adding the full region names:

In [6]:

```
def counry_code_to_country(country_code):
    upper_case = country_code.upper()
    if upper_case in country_codes:
        return country_codes[upper_case]
    return None

summed_streams['full_region_name'] = summed_streams.Region.apply(lambda x:counry_code_t
o_country(x))
summed_streams.sample(3)
```

Out[6]:

	track_id	Track Name	Artist	Region	Streams	full_region_name
606	4v094LQ9eF10k2O2Te2ngK	Too Much To Ask	Niall Horan	hu	2103	Hungary
82	0UeYCHOETPfai02uskjJ3x	Shooting Stars	Bag Raiders	pl	4208	Poland
221	1iTKkkgXj8kJkGdjGGBFqL	Two Princes - Radio Mix	Fake Pictures	de	97101	Germany

normalizing the number of streams by country:

In [7]:

```
countries_max_dict = summed_streams.groupby('full_region_name')['Streams'].max().to_dic
t()
countries_min_dict = summed_streams.groupby('full_region_name')['Streams'].min().to_dic
t()
summed_streams['normalized_popularity'] = summed_streams.apply(lambda x:(x.Streams-coun
tries_min_dict[x.full_region_name])/(countries_max_dict[x.full_region_name]-countries_m
in_dict[x.full_region_name]) ,axis =1)
summed_streams.sample(3)
```

Out[7]:

	track_id	Track Name	Artist	Region	Streams	full_region_name	n
150	13HVjjWUZFaWilh2QUJKsP	Let It Go	James Bay	ie	3393	Ireland	
597	4qknM1pQz53QOyfDVTjcM9	Me Enamoré	Shakira	pe	38542	Peru	
438	3NdDpSvN911VPGivFlV5d0	I Don't Wanna Live Forever (Fifty Shades Darke...	ZAYN	au	141598	Australia	

adding the musical features (extracted them speratly by track_id):

In [8]:

```
track_ids_with_features = pd.read_csv('track_ids_features.csv', encoding='utf-8').drop
(['Unnamed: 0'], axis=1)
```

In [9]:

```
# normalizes
def normalize(df):
    df['duration_ms_stand'] = np.where(df['duration_ms'] >= 10 * 60 * 1000, 10 * 60 * 1
000, df['duration_ms'])
    df['duration_norm'] = (df['duration_ms_stand']-df['duration_ms_stand'].min())/(df[
'duration_ms_stand'].max()-df['duration_ms_stand'].min())

    df['loudness_stand'] = np.where(df['loudness'] <= -40, -40 , df['loudness'])
    df['loudness_norm'] = (df['loudness_stand']-df['loudness_stand'].min())/(df['loudne
ss_stand'].max()-df['loudness_stand'].min())

    df['tempo_norm'] = (df['tempo']-df['tempo'].min())/(df['tempo'].max()-df['tempo'].m
in())
    return df

track_ids_with_features_normalized = normalize(track_ids_with_features)
songs_with_features = summed_streams.merge(track_ids_with_features_normalized, left_on=
'track_id',right_on = 'id')
songs_with_features.sample(3)
```

Out[9]:

	track_id	Track Name	Artist	Region	Streams	full_region_nai
962	7rdUtXasA973gmrr2Xxh3E	Homemade Dynamite (Feat. Khalid, Post Malone &...	Lorde	id	12489	Indone
879	6y2Kaz9QI01XBKJ8mTb7Pf	Skin	Rag'n'Bone Man	gb	112645	United Kingdom Great Britain a Northern
217	1dD3FPQw7qLdlPh1g1ZKRS	Ghetto Massari	Capital Bra	at	1901	Aust

3 rows × 30 columns

getting artist popularity from spotify (takes a while):

In [10]:

```
def get_artist_id(artist_name):
    results = sp.search(q='artist:' + artist_name, type='artist')
    try:
        result = results['artists']['items'][0]['id']
    except Exception:
        result = ''
    return result

artist_data = songs_with_features[['Artist']].drop_duplicates()
artist_data['artist_id'] = artist_data['Artist'].apply(lambda x: get_artist_id(str(x)))
.fillna('')
```

In [11]:

```
def get_popularity(id):
    try:
        result = sp.artist(id)['popularity']
    except Exception:
        result = ''
    return result

artist_data['artist_popularity'] = artist_data['artist_id'].apply(lambda x:  get_popula
rity(x))
top_songs_data_set = songs_with_features.merge(artist_data,on = 'Artist',how='left')
```

In [12]:

```
songs_with_features.to_csv('top_hits_for_gui.csv',index = False,encoding = 'utf-8')
```

In [13]:

```
countries_ranks = top_songs_data_set.pivot(index = 'track_id',columns = 'full_region_na
me',values='normalized_popularity').reset_index().fillna(-1)
countries_ranks.head()
```

Out[13]:

full_region_name	track_id	Argentina	Australia	Austria	Belgium	Plu
0	00vk0sEfb6mr2ehwSw0Uhn	-1.0	-1.0	-1.0	-1.0	
1	01vHGJRvJ6EZW9KO8NtZpi	0.0	-1.0	-1.0	-1.0	
2	02WacdrRpm4zlP8H7X6bnQ	-1.0	-1.0	-1.0	-1.0	
3	02djQtbRlUjbQGrJjaG0Oj	-1.0	-1.0	-1.0	-1.0	
4	03HsZP3FKOd0qWlRU01MxT	-1.0	-1.0	-1.0	-1.0	

5 rows × 55 columns

In [14]:

```
song_features = top_songs_data_set[['track_id', 'Track Name', 'Artist', 'danceability',
'energy','key', 'loudness', 'mode', 'speechiness',
                    'acousticness','instrumentalness', 'liveness', 'valence', 'tempo',
'duration_ms', 'time_signature',
                    'duration_ms_stand', 'duration_norm', 'loudness_stand', 'loudness_n
orm','tempo_norm', 'artist_id',
                    'artist_popularity']].drop_duplicates(subset='track_id')
processed_top_hits_dataset = song_features.merge(countries_ranks,on='track_id')
processed_top_hits_dataset.head()
```

Out[14]:

	track_id	Track Name	Artist	danceability	energy	key	loudness	m
0	00vk0sEfb6mr2ehwSw0Uhn	Don't Be So Shy - Filatov & Karas Remix	Imany	0.773	0.849	8.0	-8.345	
1	01vHGJRvJ6EZW9KO8NtZpi	Un Polvo	Maluma	0.811	0.681	6.0	-4.715	
2	02WacdrRpm4zlP8H7X6bnQ	Dancing On My Own	Calum Scott	0.608	0.174	1.0	-8.796	
3	02djQtbRlUjbQGrJjaG0Oj	Mina områden	Kartellen	0.517	0.671	6.0	-7.094	
4	03HsZP3FKOd0qWlRU01MxT	Wosk	Taco Hemingway	0.641	0.507	10.0	-12.010	

5 rows × 77 columns

In [15]:

```
processed_top_hits_dataset.to_csv('top_hits_processed_dataset_mini.csv',index = False,e
ncoding = 'utf-8')
```