

JUNHO 2022

# Relatório Trabalho Programa ção

PROGRAMAÇÃO E ALGORITMOS II  
COMUNICAÇÃO E DESIGN  
MULTIMÉDIA

# INTRODUÇÃO

No âmbito na unidade curricular de Programação e Algoritmos II, lecionada pelo docente João Orvalho e com os conhecimentos adquiridos ao longo do ano letivo e do semestre, foi-nos proposto o seguinte trabalho, no qual o objetivo principal é selecionar conjuntos de dados para processar, analisar e criar visualizações da informação extraída.

Para esse efeito decidimos usar o conjunto de dados do Top Hits do Spotify 2000-2019, pois é uma plataforma que usamos muito e nos é familiar.

## ANÁLISE DE DADOS

Para começar o nosso trabalho, fomos ao site kaggle e retirámos um conjunto de dados sobre o Top Hits do Spotify. Os dados que recolhemos estão apresentados num período de tempo de 19, ou seja, de 2000 a 2019, visto sendo esse o período de tempo mais recente que o site dispunha. O ficheiro já se encontrava em csv, o que tornou essa parte do trabalho mais fácil.

# DICIONÁRIO DE DADOS

Nome da Coluna	Significado
artist	artistas
song	músicas
duration_ms	duração das músicas
explicit	conteúdo explícito
year	ano
popularity	popularidade
danceability	dançabilidade
energy	energia
key	chave
loudness	volume
mode	modo
speechiness	fala
acousticness	acústica
instrumentalness	instrumentalidade
liveness	vivacidade
valence	valência
tempo	tempo
genre	gênero

# DICIONÁRIO DE DADOS (DESCRIÇÃO)

**artist:** Nome do artista.

**song:** Nome da música.

**duration\_ms:** Duração da música em milissegundos.

**explicit:** A letra ou conteúdo de uma música ou videoclipe contém um ou mais dos critérios que seriam classificados com impróprios ou ofensivos para crianças.

**year:** Ano de lançamento da música.

**popularity:** Quanto maior o valor, mais popularidade a música tem.

**danceability:** Dancibilidade descreve o quanto uma música é adequada para dançar, baseada numa combinação de elementos musicais, incluindo o tempo, estabilidade do ritmo, força do beat e regularidade geral. O valor de 0.0 tem menos dancibilidade e o valor de 1.0 tem mais dancibilidade.

**energy:** Energia é a medida de 0.0 a 1.0 e representa a medida perceptual de intensidade e atividade.

**key:** A chave em que a pista está. Os inteiros mapeiam para os arremessos usando a notação padrão pitch class. Por exemplo, 0 = C, 1 = C#/D ♭, 2 = D, e assim por diante. Se não for detetada chave, o valor é -1.

**loudness:** O ruído geral da música é em Decibels (dB). Os valores de ruído são médios em toda a pista e são úteis para comparar o ruído relativo das faixas. O ruído é a qualidade de um som que é a principal correlação psicológica da força física (amplitude). Os valores variam tipicamente entre -60 e 0 db.

**mode:** O modo indica a modalidade (maior ou menor) de uma faixa, o tipo de escala a partir do qual o seu conteúdo melódico é derivado. Major é representado por 1 e menor é 0.

**speechiness:** A fala deteta a presença de palavras faladas numa faixa. Quanto mais exclusivamente fala-like a gravação (por exemplo, talk show, audio book, poesia), mais perto de 1.0 o valor do atributo. Valores acima de 0,66 descrevem faixas que provavelmente são feitas inteiramente de palavras faladas. Valores entre 0,33 e 0,66 descrevem faixas que podem conter música e fala, quer em secções quer em camadas, incluindo casos como música rap. Valores abaixo de 0,33 provavelmente representam música e outras faixas não-fala- semelhantes.

**acousticness:** Uma medida de confiança de 0,0 a 1.0 de se a pista é acústica. 1.0 representa alta confiança a pista é acústica.

**instrumentalness:** Prevê se uma faixa não contém vocais. Os sons "Ooh" e "aah" são tratados como instrumentais neste contexto. As faixas de palavras rap ou faladas são claramente "vocais". Quanto mais próximo o valor da instrumentalidade for de 1.0, maior é a probabilidade de a faixa não conter conteúdo vocal. Valores acima de 0,5 destinam-se a representar faixas instrumentais, mas a confiança é maior à medida que o valor se aproxima de 1.0.

**liveness:** Deteta a presença de uma audiência na gravação. Os valores de maior vivacidade representam uma maior probabilidade de que a pista tenha sido realizada ao vivo. Um valor acima de 0,8 proporciona uma forte probabilidade de que a pista esteja ao vivo.

**valence:** Uma medida de 0.0 a 1.0 descrevendo a positiva musical transmitida por uma faixa. Faixas com alta valência soam mais positivas (por exemplo, felizes, alegres, eufóricas), enquanto faixas com baixa valência soam mais negativas (por exemplo, tristes, deprimidas, zangadas).

**tempo:** O tempo estimado global de uma faixa em batidas por minuto (BPM). Na terminologia musical, o tempo é a velocidade ou o ritmo de uma determinada peça e deriva diretamente da duração média do ritmo.

**genre:** género da música.

# CÓDIGO

*Como mencionado no início deste trabalho, o tema do nosso trabalho é a análise do Top Hits do Spotify de 2000 a 2019. Este código foi depois importado para o Jupyter, que deu origem a esta tabela.*

	artist	song	duration_ms	explicit	year	popularity	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	genre
0	Britney Spears	Oops!...I Did It Again	211160	False	2000	77	0.751	0.834	1	-5.444	0	0.0437	0.3000	0.000018	0.3550	0.894	95.053	pop
1	blink-182	All The Small Things	167066	False	1999	79	0.434	0.897	0	-4.918	1	0.0488	0.0103	0.000000	0.6120	0.684	148.726	rock, pop
2	Faith Hill	Breathe	250546	False	1999	66	0.529	0.496	7	-9.007	1	0.0290	0.1730	0.000000	0.2510	0.278	136.859	pop, country
3	Bon Jovi	It's My Life	224493	False	2000	78	0.551	0.913	0	-4.063	0	0.0466	0.0263	0.000013	0.3470	0.544	119.992	rock, metal
4	*NSYNC	Bye Bye Bye	200560	False	2000	65	0.614	0.928	8	-4.806	0	0.0516	0.0408	0.001040	0.0845	0.879	172.656	pop
***	***	***	***	***	***	***	***	***	***	***	***	***	***	***	***	***	***	***
1995	Jonas Brothers	Sucker	181026	False	2019	79	0.842	0.734	1	-5.065	0	0.0588	0.0427	0.000000	0.1060	0.952	137.958	pop
1996	Taylor Swift	Cruel Summer	178426	False	2019	78	0.552	0.702	9	-5.707	1	0.1570	0.1170	0.000021	0.1050	0.564	169.994	pop
1997	Blanco Brown	The Git Up	200593	False	2019	69	0.847	0.678	9	-8.635	1	0.1090	0.0669	0.000000	0.2740	0.811	97.984	hip hop, country
1998	Sam Smith	Dancing With A Stranger (with Normani)	171029	False	2019	75	0.741	0.520	8	-7.513	1	0.0656	0.4500	0.000002	0.2220	0.347	102.998	pop
1999	Post Malone	Circles	215280	False	2019	85	0.695	0.762	0	-3.497	1	0.0395	0.1920	0.002440	0.0863	0.553	120.042	hip hop

***Neste tópico apresentamos o nosso dicionário de dados, que contém os diferentes tipos de variáveis presentes no projeto, int, float, object e bool.***

```
In [4]: song.dtypes #dicionário de dados
```

```
Out[4]: artist          object
song                  object
duration_ms          int64
explicit              bool
year                  int64
popularity            int64
danceability          float64
energy                float64
key                   int64
loudness              float64
mode                  int64
speechiness           float64
acousticness          float64
instrumentalness      float64
liveness              float64
valence               float64
tempo                 float64
genre                 object
dtype: object
```

## Qual a canção mais popular?

```
In [14]: res=song[["song", "popularity"]]  
res1=res.sort_values("popularity", ascending= False)  
res1
```

```
Out[14]:
```

	song	popularity
1322	Sweater Weather	89
1311	Another Love	88
201	Without Me	87
1613	Wait a Minute!	86
6	The Real Slim Shady	86
...	...	...
1637	Side To Side	0
217	Rock The Boat	0
1631	Cool Girl	0
1626	Sex	0
126	Ain't It Funny	0

2000 rows x 2 columns

```
In [20]: mais_popular=res1.iloc[0]  
mais_popular=mais[0]  
print("a musica mais popular é a " + mais_popular)  
a musica mais popular é a Sweater Weather
```

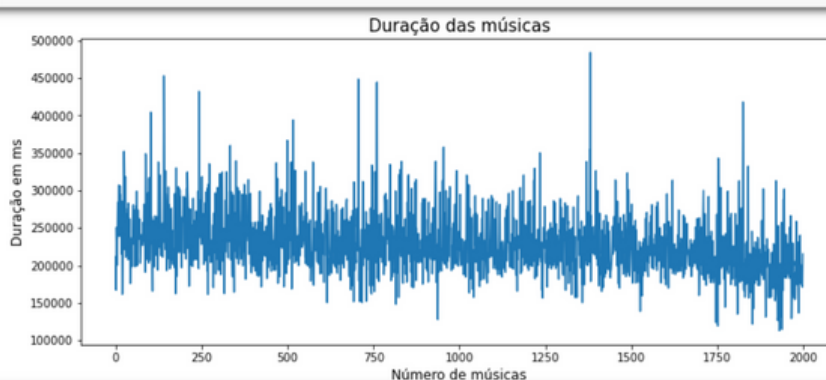
Depois de especificadas as variáveis, começamos a fazer a análise dos dados pelas perguntas colocadas. Nesta questão procurámos saber qual foi a canção mais popular do top hits no espaço de tempo entre 2000-2019, para a qual procurámos fazer uma tabela onde ordenasse as músicas de mais popular para menos, e depois especificámos com o `iloc` a 0 para que o resultado fosse só uma música, a mais popular. A resposta obtida foi Sweater Weather.

## Qual a duração da música com maior duração?

```
In [6]: max_time = song["duration_ms"].max()  
segundos = max_time/1000  
minutos = segundos/60  
print("A música com maior duração é de: " + str(segundos) + "s" + " ou " + str(minutos) + "min")  
A música com maior duração é de: 484.146s ou 8.0691min
```

Na seguinte questão quisemos saber qual a duração da música com maior duração, na qual a resposta obtida foi de 484.146s ou 8.069min. De seguida fizemos um gráfico para termos uma ideia visual mais clara dos dados da pergunta e da resposta obtida.

```
In [30]: #Gráfico da duração das músicas.  
fig, axs = plt.subplots(figsize = (12, 5))  
song2 = song["duration_ms"].plot()  
plt.title("Duração das músicas", fontsize=15)  
plt.xlabel('Número de músicas', fontsize=12)  
plt.ylabel('Duração em ms', fontsize=12)
```



### Qual o artista com mais músicas no "top hits"?

```
In [31]: result = song.groupby("artist").count()
ordem=result["song"].sort_values(ascending=False)
artistamais=ordem.index[0]
artistamais
print("O artista com mais musicas é: " + artistamais)

O artista com mais musicas é: Rihanna
```

Nesta questão procurámos saber qual o artista com mais músicas no top hits, na qual resolvemos por agrupamento e ordenação a questão á qual a resposta foi a artista Rihanna.

### Qual o número de músicas no top hits do artista com mais músicas?

```
In [82]: result = song.groupby("artist").count()
ordem=result["song"].sort_values(ascending=False)
num=ordem.iloc[0]
print("A " + artistamais + " tem " + str(num) + " músicas no top hits!")

A Rihanna tem 25 músicas no top hits!
```

Na seguinte análise usámos o agrupamento de dados e o sort\_values em ascending=False para determinar quantas músicas a Rihanna tinha no top hits entre 2000 e 2019, na qual a resposta foram 25 músicas.

### Quantas músicas existem com conteúdo explicit?

```
In [80]: result = song.groupby("explicit").count()
result

Out[80]:
```

	artist	song	duration_ms	year	popularity	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence
explicit															
False	1449	1449	1449	1449	1449	1449	1449	1449	1449	1449	1449	1449	1449	1449	1449
True	551	551	551	551	551	551	551	551	551	551	551	551	551	551	551

```
In [81]: nummusicasexp=result.iloc[1]["song"]
print("Existem " + str(nummusicasexp) + " músicas com o conteúdo explicit." )

Existem 551 músicas com o conteúdo explicit.
```

Nesta última questão, quisemos saber quantas músicas existem com conteúdo explicit. Para isso fizemos uma tabela onde dizia quantas músicas tinham conteúdo explicit e quantas não tinham. Depois fizemos iloc da coluna "song" e da linha 1 (que representa "True") o que daria 551, que foi a resposta dada.



# CONCLUSÃO

Durante o decorrer do trabalho apercebemo-nos de que este seria uma mais valia no aprofundamento dos nossos conhecimentos de programação, e de busca e análise de bases de dados, pelo facto de estarmos a pegar em algo que nos é familiar, como tabelas de dados, e transformar isso em perguntas, respostas e tabelas a partir de código.

Pegámos assim num tema fácil de interpretar e esprememo-lo ao nível da programação.