Proyecto SQL!

Para el proyecto de SQL recordaremos como el coronavirus tomó al mundo entero por sorpresa, cambiando la rutina diaria de todas las personas. Los habitantes de las ciudades ya no pasaban su tiempo libre fuera, yendo a cafés y centros comerciales; sino que más gente se quedaba en casa, leyendo libros. Eso atrajo la atención de las startups que se apresuraron a desarrollar nuevas aplicaciones para los amantes de los libros.

Entramos a una base de datos de uno de los servicios que compiten en dicho mercado. Contiene datos sobre libros, editoriales, autores, calificaciones de clientes y reseñas de libros. Esta información la utilizararemos para generar una propuesta de valor para un nuevo producto.

Se nos solicita que desarrollemos los siguientes ejercicios:

- Encuentrar el número de libros publicados después del 1 de enero de 2000.
- Encuentrar el número de reseñas de usuarios y la calificación promedio para cada libro.
- Identificar la editorial que ha publicado el mayor número de libros con más de 50 páginas (esto nos ayudará a excluir folletos y publicaciones similares de nuestro análisis).
- Identificar al autor que tiene la más alta calificación promedio del libro: mira solo los libros con al menos 50 calificaciones.
- Encuentra el número promedio de reseñas de texto entre los usuarios que calificaron más de 50 libros.

In [14]:
```
# Importar librerías
import pandas as pd
from sqlalchemy import create_engine

# Establecer la conexión a la base de datos.
db_config = {'user': 'practicum_student',         # nombre de usuario
             'pwd': 's65BlTKV3faNIGhmvJVzOqhs', # contraseña
             'host': 'rc1b-wcoijxj3yxfsf3fs.mdb.yandexcloud.net',
             'port': 6432,              # puerto de conexión
             'db': 'data-analyst-final-project-db'}          # nombre de la

connection_string = 'postgresql://{}:{}@{}:{}/{}'.format(db_config['user'],
                                                                 db_conf
                                                                   db_cc
                                                                   db_cc
                                                                   db_cc


engine = create_engine(connection_string, connect_args={'sslmode':'require'}
```

Estudiamos las tablas.

In [15]:
```
# Análisis de la tabla de los libros.

libros = 'SELECT * FROM books'
pd.io.sql.read_sql(libros, con = engine).head()
```

Out[15]:

	book_id	author_id	title	num_pages	publication_date	publisher_id
0	1	546	'Salem's Lot	594	2005-11-01	93
1	2	465	1 000 Places to See Before You Die	992	2003-05-22	336
2	3	407	13 Little Blue Envelopes (Little Blue Envelope...	322	2010-12-21	135
3	4	82	1491: New Revelations of the Americas Before C...	541	2006-10-10	309
4	5	125	1776	386	2006-07-04	268

La tabla libros contiene los siguientes datos:

- `Clave primaria (PK) book_id` : identificación del libro
- `Claves foránea (FK) author_id` : identificación del autor o autora
- `Claves foránea (FK) publisher_id` : identificación de la editorial
- `title` : título
- `num_pages` : número de páginas
- `publication_date` : fecha de la publicación

Relaciones: Cada libro está asociado a un autor mediante author_id. Cada libro pertenece a una editorial mediante publisher_id. Esta tabla es clave para conectar con reviews y ratings a través de book_id.

In [16]:
```
# Análisis de la tabla de los autores.

autores = 'SELECT * FROM authors'
pd.io.sql.read_sql(autores, con = engine).head()
```

Out[16]:

	author_id	author
0	1	A.S. Byatt
1	2	Aesop/Laura Harris/Laura Gibbs
2	3	Agatha Christie
3	4	Alan Brennert
4	5	Alan Moore/David Lloyd

La tabla autores contiene los siguientes datos:

- `Clave primaria (PK) author_id` : identificación del autor o autora
- `author` : el autor o la autora

Relación con otras tablas: Se puede conectar con la tabla books mediante author_id.
Relaciones: Cada autor puede tener uno o varios libros en la tabla books.

In [17]:
```
# Análisis de la tabla de los ratings.

ratings = """
SELECT
    *
FROM
    ratings
"""
pd.io.sql.read_sql(ratings, con = engine).head()
```

Out[17]:

	rating_id	book_id	username	rating
0	1	1	ryanfranco	4
1	2	1	grantpatricia	2
2	3	1	brandtandrea	5
3	4	2	lorichen	3
4	5	2	mariokeller	2

La tabla ratings contiene los siguientes datos:

- `Clave primaria (PK) rating_id` : identificación de la calificación
- `Clave foránea (FK) book_id` : identificación del libro
- `username` : el nombre del usuario que revisó el libro
- `rating` : calificación

Relaciones: Cada calificación está asociada a un libro mediante book_id. Un libro puede tener una o varias calificaciones.

In [18]:
```
# Análisis de la tabla de las reseñas.

reviews = """
SELECT
    *
FROM
    reviews
"""
pd.io.sql.read_sql(reviews, con = engine).head()
```

Out[18]:

	review_id	book_id	username	text
0	1	1	brandtandrea	Mention society tell send professor analysis. ...
1	2	1	ryanfranco	Foot glass pretty audience hit themselves. Amo...
2	3	2	lorichen	Listen treat keep worry. Miss husband tax but ...
3	4	3	johnsonamanda	Finally month interesting blue could nature cu...
4	5	3	scotttamara	Nation purpose heavy give wait song will. List...

La tabla reviews contiene los siguientes datos:

- `Clave primaria (PK) review_id` : identificación de la reseña
- `Clave foránea (FK) book_id` : identificación del libro
- `username` : el nombre del usuario que revisó el libro
- `text` : el texto de la reseña

Relaciones: Cada reseña está asociada a un libro mediante book_id. Un libro puede tener una o varias reseñas.

In [19]:
```
# Análisis de la tabla de los editores.

editores = """
SELECT
    *
FROM
    publishers
"""
pd.io.sql.read_sql(editores, con = engine).head()
```

Out[19]:

	publisher_id	publisher
0	1	Ace
1	2	Ace Book
2	3	Ace Books
3	4	Ace Hardcover
4	5	Addison Wesley Publishing Company

La tabla editores contiene los siguientes datos:

- `Clave primaria (PK) publisher_id`: identificación de la editorial
- `publisher`: la editorial

Relaciones: Cada editorial puede haber publicado uno o varios libros en la tabla books.

In [20]:
```sql
# Encontrar el número de libros publicados después del 1 de enero de 2000.

libros_publicados = """
SELECT
    COUNT(*)
FROM
    books
WHERE
    publication_date > '2000-01-01'
"""

pd.io.sql.read_sql(libros_publicados, con = engine)
```

Out[20]:

	count
0	819

Se identificó que el total de libros publicados después del 1 de enero del 2020 fueron 819 libros.

In [21]:
```sql
# Encuentra el número de reseñas de usuarios y la calificación promedio para

libros_con_numero_de_resenias_y_calificacion_promedio = """
SELECT
    b.book_id,
    b.title,
    COUNT(r.review_id) AS review_count,
    ROUND(AVG(rt.rating), 2) AS average_rating
FROM
    books b
JOIN
    reviews r ON b.book_id = r.book_id
JOIN
    ratings rt ON b.book_id = rt.book_id
GROUP BY
    b.book_id
ORDER BY
    review_count DESC
LIMIT 10
"""
pd.io.sql.read_sql(libros_con_numero_de_resenias_y_calificacion_promedio, cc
```

Out[21]:

	book_id	title	review_count	average_rating
0	948	Twilight (Twilight #1)	1120	3.66
1	750	The Hobbit or There and Back Again	528	4.13
2	673	The Catcher in the Rye	516	3.83
3	302	Harry Potter and the Prisoner of Azkaban (Harr...	492	4.41
4	299	Harry Potter and the Chamber of Secrets (Harry...	480	4.29
5	75	Angels & Demons (Robert Langdon #1)	420	3.68
6	301	Harry Potter and the Order of the Phoenix (Har...	375	4.19
7	779	The Lightning Thief (Percy Jackson and the Oly...	372	4.08
8	722	The Fellowship of the Ring (The Lord of the Ri...	370	4.39
9	79	Animal Farm	370	3.73

Se detectaron 994 libros que cuentan con reseñas, se imprimen los primeros 10 libros:

- El primer libro, Twilight tiene 1120 reviews y un promedio de rating de 3.66.
- El segundo y tercer libro, The Hobbit o There and Back Again con 528 reviews y 4.12 de rating.
- El cuarto libro, The Catcher in the Rye tiene 516 reviews y 3.82 en promedio de rating.

In [22]:
```
# Identifica la editorial que ha publicado el mayor número de libros con más

editorial_con_mas_publicaciones = """
SELECT
        p.publisher,
    COUNT(b.book_id) AS num_books
FROM
        publishers p
JOIN
        books b ON p.publisher_id = b.publisher_id
WHERE
        b.num_pages > 50
GROUP BY
    p.publisher
ORDER BY
        num_books DESC
LIMIT 1
"""
pd.io.sql.read_sql(editorial_con_mas_publicaciones , con = engine).head()
```

Out[22]:

	publisher	num_books
0	Penguin Books	42

Se ha identificado que la editorial que ha publicado el mayor número de libros es Penguin Books con 42 libros.

In [23]:
```
# Identifica al autor que tiene la más alta calificación promedio del libro:

autor_con_mas_alta_calificacion_promedio = """
SELECT
    a.author_id,
    a.author,
    ROUND(AVG(rt.rating), 2) AS calificacion_promedio
FROM
    authors a
JOIN
    books b ON a.author_id = b.author_id
JOIN
    ratings rt ON b.book_id = rt.book_id
WHERE
    b.book_id IN (
        SELECT
            b.book_id
        FROM
            books b
        JOIN
            ratings rt ON b.book_id = rt.book_id
        GROUP BY
            b.book_id
        HAVING
            COUNT(rt.rating) >= 50
    )
GROUP BY
    a.author_id
ORDER BY
    calificacion_promedio DESC
LIMIT 1
"""
pd.io.sql.read_sql(autor_con_mas_alta_calificacion_promedio, con = engine).h
```

Out[23]:

	author_id	author	calificacion_promedio
0	236	J.K. Rowling/Mary GrandPré	4.29

Se logro identificar que los autores que tuvieron la calificación más alta promedio fueron J.K. Rowling/Mary GrandPré con 4.29%.

In [28]:
```
# Encuentra el número promedio de reseñas de texto entre los usuarios que ca

numero_promedio_de_resenias = """
SELECT
        ROUND(AVG(conteo_resenias), 2) AS promedio_de_resenias
```

```
FROM
        (SELECT
                rt.username,
                COUNT(r.review_id) AS conteo_resenias
        FROM
                ratings rt
    JOIN
            reviews r ON rt.book_id = r.book_id
        GROUP BY
                rt.username
        HAVING
                COUNT(r.review_id) > 50
        ) AS subquery
"""

pd.io.sql.read_sql(numero_promedio_de_resenias , con = engine)
```

Out[28]:

	promedio_de_resenias
0	163.54

Se encontro que el numero promedio de reseñas, solo entre los usuarios que calificaron más de 50 libros, fue de 163 reseñas.

Conclusiones

- Encontrar el número de libros publicados después del 1 de enero de 2000.
  Se identificó que el total de libros publicados después del 1 de enero del 2020 fueron 819 libros. Esto demuestra que la producción de libros continua vigente hoy en día a pesar de las diferentes evoluciones tecnologicas con las que se cuenta, ya que también se ha implementado el uso de los e-books.

- Encuentra el número de reseñas de usuarios y la calificación promedio para cada libro.
  Se detectaron 994 libros que cuentan con reseñas, se enlista el top 3:

  - El primer libro, Twilight tiene 1120 reviews y un promedio de rating de 3.66.
  - El segundo y tercer libro, The Hobbit o There and Back Again con 528 reviews y 4.12 de rating.
  - El cuarto libro, The Catcher in the Rye tiene 516 reviews y 3.82 en promedio de rating.

  No porque un libro tenga el mayor número de reseñas querra decir que tenga la mejor calificación del público.

- Identifica la editorial que ha publicado el mayor número de libros con más de 50 páginas.
  Se ha identificado que la editorial que ha publicado el mayor número de libros es

Penguin Books con 42 libros, mostrandose como una de las principales editoriales con una fuerte presencia en el mercado.

- Identifica al autor que tiene la más alta calificación promedio del libro: mira solo los libros con al menos 50 calificaciones.
  Se logro identificar que los autores que tuvieron la calificación más alta promedio fueron J.K. Rowling/Mary GrandPré con 4.29%.

  Vaya que la saga de Harry Potter ha sido una de las preferidas en esta última epoca.

- Encuentra el número promedio de reseñas de texto entre los usuarios que calificaron más de 50 libros.
  Se encontro que el numero promedio de reseñas, solo entre los usuarios que calificaron más de 50 libros, fue de 163 reseñas.

Recomendaciones:

- Promover libros a través de redes sociales: Hoy en día el hecho de hacer que una pelicula o algún comentario se haga viral puede ayudar mucho a fomentar la lectura del libro.

- Creación de pequeños adelantos a través de e-books: Que apoye a despertar ese interés en el lector para que lo cautive y con eso lo incline a adquirir el libro en físico o dígital.

- Enviar muestras gratis a autores reconocidos para que den su reseña. Si la reseña de un autor reconocido es buena sobre algún libro esto puede incitar a otros lectores a leerlo también.