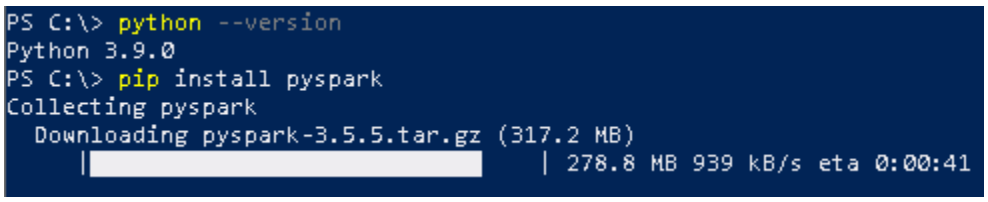
SPARK SQL EXERCISES

Cài đặt

Chú ý dùng Python3.9 - https://www.python.org/ftp/python/3.9.0/python-3.9.0-amd64.exe

Cài thư viện pyspark với pip



Bài tập 1

Download tập tin countries.csv từ link – https://github.com/vovanhai-ueh/resources/blob/master/countries.csv sau đó save vào thệ thống file HDFS.

Tạo data frame từ bộ dữ liệu này.

Tạo một TempView có tên "countries" sau đó dung Park SQL thực thi các yêu cầu sau

1. Hiển thị toàn bộ thông tin của một quốc gia khi biết tên quốc gia
2. Liệt kê các quốc gia thuộc vùng "ASIA (EX. NEAR EAST)"
3. Liệt kê 5 quốc gia có dân số đông nhất
4. Liệt kê 5 quốc gia có diện tích nhỏ nhất / lớn nhất
5. Liệt kê 5 quốc gia có mật độ dân số nhỏ nhất / lớn nhất
6. Liệt kê các quốc gia thuộc vùng "NEAR EAST" có dân số lớn hơn 15 triệu
7. Liệt kê 5 quốc gia châu Phi có tỷ lệ trẻ em tử vong (Infant mortality (per 1000 births)) cao nhất.
8. Hiển thị khu vực có trung bình GDP cao nhất
9. Và các câu hỏi tự sinh viên nghĩ ra dựa trên bộ dữ liệu này

Ngoài ra, có thể lấy thêm các thông tin về các thành phố, quốc gia, vùng lành thổ tại: https://github.com/dr5hn/countries-states-cities-database/blob/master/README.md

Bài tập 2

Download the online retail dataset in CSV format with the following link: https://github.com/vovanhai-ueh/resources/blob/master/new_retail_data.rar

Create a DataFrame from this dataset, and create a TempView with the name "Retails". Complete the following questions:

1. Get the order count per customer for January 2024.
    - Tables - orders and customers
    - Data should be sorted in descending order by count and ascending order by customer ID.

    - Output should contain customer_id, customer_first_name, customer_last_name and customer_order_count.
2. Get the customer details of those who have not placed any orders in January 2024.
    - Tables - orders and customers
    - Data should be sorted in ascending order by customer_id
    - Output should contain all the fields from the customers
3. Get the revenue generated by each customer for January 2024
    - Tables - orders, order_items, and customers
    - Data should be sorted in descending order by revenue and then ascending order by customer_id.
    - Output should contain customer_id, customer_first_name, customer_last_name, customer_revenue.
    - If the customer places no orders, then the corresponding revenue for a given customer should be 0.
    - Consider only COMPLETE and CLOSED orders
4. Get the revenue generated for each category for January 2024
    - Tables - orders, order_items, products and categories
    - Data should be sorted in ascending order by category_id.
    - Output should contain all the fields from the category along with the revenue as category_revenue.
    - Consider only COMPLETE and CLOSED orders
5. Get the products for each department.
    a. Tables - departments, categories, products
    b. Data should be sorted in ascending order by department_id
    c. The output should contain all the fields from the department and the product count as product_count

Bài tập 3

Source: https://sparksql.itversity.com/

Create a database with the name “retails” in SQLite/MariaDB/MS SQL Server.

Create a table with the following SQL code

```
USE retails;

DROP TABLE IF EXISTS users;

CREATE TABLE users(
    user_id INT,
    user_first_name VARCHAR(30),
    user_last_name VARCHAR(30),
    user_email_id VARCHAR(50),
    user_gender VARCHAR(1),
    user_unique_id VARCHAR(15),
    user_phone_no VARCHAR(20),
    user_dob DATE,
    created_ts TIMESTAMP
);
```

Insert some records using SQL

```
INSERT INTO users
VALUES
    (1, 'Giuseppe', 'Bode', 'gbode0@imgur.com', 'M', '88833-8759', '+86
(764) 443-1967', '1973-05-31', '2018-04-15 12:13:38'),
    (2, 'Lexy', 'Gisbey', 'lgisbey1@mail.ru', 'F', '262501-029', '+86 (751)
160-3742', '2003-05-31', '2020-12-29 06:44:09'),
    (3, 'Karel', 'Claringbold', 'kclaringbold2@yale.edu', 'F', '391-33-
2823', '+62 (445) 471-2682', '1985-11-28', '2018-11-19 00:04:08'),
    (4, 'Marv', 'Tanswill', 'mtanswill3@dedecms.com', 'F', '1195413-80',
'+62 (497) 736-6802', '1998-05-24', '2018-11-19 16:29:43'),
    (5, 'Gertie', 'Espinoza', 'gespinoza4@nationalgeographic.com', 'M',
'471-24-6869', '+249 (687) 506-2960', '1997-10-30', '2020-01-25 21:31:10'),
    (6, 'Saleem', 'Danneil', 'sdanneil5@guardian.co.uk', 'F', '192374-933',
'+63 (810) 321-0331', '1992-03-08', '2020-11-07 19:01:14'),
    (7, 'Rickert', 'O''Shiels', 'roshiels6@wikispaces.com', 'M', '749-27-47-
52', '+86 (184) 759-3933', '1972-11-01', '2018-03-20 10:53:24'),
    (8, 'Cybil', 'Lissimore', 'clissimore7@pinterest.com', 'M', '461-75-
4198', '+54 (613) 939-6976', '1978-03-03', '2019-12-09 14:08:30'),
    (9, 'Melita', 'Rimington', 'mrimington8@mozilla.org', 'F', '892-36-676-
2', '+48 (322) 829-8638', '1995-12-15', '2018-04-03 04:21:33'),
    (10, 'Benetta', 'Nana', 'bnana9@google.com', 'M', '197-54-1646', '+420
(934) 611-0020', '1971-12-07', '2018-10-17 21:02:51'),
    (11, 'Gregorius', 'Gullane', 'ggullanea@prnewswire.com', 'F', '232-55-
52-58', '+62 (780) 859-1578', '1973-09-18', '2020-01-14 23:38:53'),
    (12, 'Una', 'Glayzer', 'uglayzerb@pinterest.com', 'M', '898-84-336-6',
'+380 (840) 437-3981', '1983-05-26', '2019-09-17 03:24:21'),
    (13, 'Jamie', 'Vosper', 'jvosperc@umich.edu', 'M', '247-95-68-44', '+81
(205) 723-1942', '1972-03-18', '2020-07-23 16:39:33'),
    (14, 'Calley', 'Tilson', 'ctilsond@issuu.com', 'F', '415-48-894-3',
'+229 (698) 777-4904', '1987-06-12', '2020-06-05 12:10:50'),
    (15, 'Peadar', 'Gregorowicz', 'pgregorowicze@omniture.com', 'M', '403-
39-5-869', '+7 (267) 853-3262', '1996-09-21', '2018-05-29 23:51:31'),
    (16, 'Jeanie', 'Webling', 'jweblingf@booking.com', 'F', '399-83-05-03',
'+351 (684) 413-0550', '1994-12-27', '2018-02-09 01:31:11'),
    (17, 'Yankee', 'Jelf', 'yjelfg@wufoo.com', 'F', '607-99-0411', '+1 (864)
112-7432', '1988-11-13', '2019-09-16 16:09:12'),
    (18, 'Blair', 'Aumerle', 'baumerleh@toplist.cz', 'F', '430-01-578-5',
'+7 (393) 232-1860', '1979-11-09', '2018-10-28 19:25:35'),
    (19, 'Pavlov', 'Steljes', 'psteljesi@macromedia.com', 'F', '571-09-
6181', '+598 (877) 881-3236', '1991-06-24', '2020-09-18 05:34:31'),
    (20, 'Darn', 'Hadeke', 'dhadekej@last.fm', 'M', '478-32-02-87', '+370
(347) 110-4270', '1984-09-04', '2018-02-10 12:56:00'),
    (21, 'Wendell', 'Spanton', 'wspantonk@de.vu', 'F', null, '+84 (301) 762-
1316', '1973-07-24', '2018-01-30 01:20:11'),
    (22, 'Carlo', 'Yearby', 'cyearbyl@comcast.net', 'F', null, '+55 (288)
623-4067', '1974-11-11', '2018-06-24 03:18:40'),
    (23, 'Sheila', 'Evitts', 'sevittsm@webmd.com', null, '830-40-5287',
null, '1977-03-01', '2020-07-20 09:59:41'),
    (24, 'Sianna', 'Lowdham', 'slowdhamn@stanford.edu', null, '778-0845',
null, '1985-12-23', '2018-06-29 02:42:49'),
    (25, 'Phylys', 'Aslie', 'paslieo@qq.com', 'M', '368-44-4478', '+86 (765)
152-8654', '1984-03-22', '2019-10-01 01:34:28')
```

Test the results

```
SELECT * FROM users LIMIT 10
```

```
3  SELECT * FROM users LIMIT 10
```

users (10r × 9c)

#	us...	user_first_name	user_last_name	user_email_id	user_gender	user_unique_id	user_phone_no	user_dob	created_ts
1	1	Giuseppe	Bode	gbode0@imgur.com	M	88833-8759	+86 (764) 443-1967	1973-05-31	2018-04-15 12:13:38
2	2	Lexy	Gisbey	lgisbey1@mail.ru	F	262501-029	+86 (751) 160-3742	2003-05-31	2020-12-29 06:44:09
3	3	Karel	Claringbold	kclaringbold2@yale.edu	F	391-33-2823	+62 (445) 471-2682	1985-11-28	2018-11-19 00:04:08
4	4	Marv	Tanswill	mtanswill3@dedecms.com	F	1195413-80	+62 (497) 736-6802	1998-05-24	2018-11-19 16:29:43
5	5	Gertie	Espinoza	gespinoza4@nationalgeographic.com	M	471-24-6869	+249 (687) 506-2960	1997-10-30	2020-01-25 21:31:10
6	6	Saleem	Danneil	sdanneil5@guardian.co.uk	F	192374-933	+63 (810) 321-0331	1992-03-08	2020-11-07 19:01:14
7	7	Rickert	O'Shiels	roshiels6@wikispaces.com	M	749-27-47-52	+86 (184) 759-3933	1972-11-01	2018-03-20 10:53:24
8	8	Cybil	Lissimore	clissimore7@pinterest.com	M	461-75-4198	+54 (613) 939-6976	1978-03-03	2019-12-09 14:08:30
9	9	Melita	Rimington	mrimington8@mozilla.org	F	892-36-676-2	+48 (322) 829-8638	1995-12-15	2018-04-03 04:21:33
10	10	Benetta	Nana	bnana9@google.com	M	197-54-1646	+420 (934) 611-0020	1971-12-07	2018-10-17 21:02:51

Exercise 1

Get all the number of users created per year.

- Use the users table for this exercise.
- The output should contain a 4-digit year and count.
- Use date-specific functions to get the year using created_ts.
- Make sure you define aliases to the columns as created_year and user_count, respectively.
- Data should be sorted in ascending order by created_year.

Here is the sample output.

created_year	user_count
2018	13
2019	4
2020	8

Exercise 2

Get the day name of the birth days for all the users born in the month of June.

- Use the users table for this exercise.
- Output should contain user_id, user_dob, user_email_id and user_day_of_birth.
- Use date-specific functions to get the month using user_dob.
- user_day_of_birth should be a full day with the first character in upper case, such as Tuesday
- Data should be sorted by day within the month of May.

Sample output

user_id	user_dob	user_email_id	user_day_of_birth
4	1998-05-24	mtanswill3@dedecms.com	Sunday
12	1983-05-26	uglayzerb@pinterest.com	Thursday
1	1973-05-31	gbode0@imgur.com	Thursday
2	2003-05-31	lgisbey1@mail.ru	Saturday

Exercise 3

Get the names and email IDs of users added in the year 2019.

- Use the users table for this exercise.
- Output should contain user_id, user_name, user_email_id, created_ts, created_year.
- Use date-specific functions to get the year using created_ts.
- user_name is a derived column by concatenating user_first_name and user_last_name with a space in between.
- user_name should have values in uppercase.
- Data should be sorted in ascending order by user_name

Sample output:

user_id	user_name	user_email_id	created_ts	created_year
8	CYBIL LISSIMORE	clissimore7@pinterest.com	2019-12-09 14:08:30	2019.0
25	PHYLYS ASLIE	paslieo@qq.com	2019-10-01 01:34:28	2019.0
12	UNA GLAYZER	uglayzerb@pinterest.com	2019-09-17 03:24:21	2019.0
17	YANKEE JELF	yjelfg@wufoo.com	2019-09-16 16:09:12	2019.0

Exercise 4

Get the number of users by gender.

- Use the users table for this exercise.
- Output should contain gender and user_count.
- For males, the output should display Male, and for females, the output should display Female.
- If gender is not specified, then it should display Not Specified.
- Data should be sorted in descending order by user_count.

Sample output

user_gender	user_count
Female	13
Male	10
Not Specified	2

Exercise 5

Get the last 4 digits of unique IDs.

- Use the users table for this exercise.
- Output should contain user_id, user_unique_id and user_unique_id_last4
- Unique IDs are either null or not null.
- Unique IDs contain numbers and hyphens and are of different lengths.

- We need to get the last 4 digits, discarding hyphens only when the number of digits is at least 9.
- If the unique ID is null, then you should display Not Specified.
- After discarding hyphens, if the unique ID has fewer than 9 digits, then you should display Invalid Unique ID.
- Data should be sorted by user_id. You might see None or null for those user IDs where there is no unique ID for the user_unique_id

user_id	user_unique_id	user_unique_id_last4
1	88833-8759	8759
2	262501-029	1029
3	391-33-2823	2823
4	1195413-80	1380
5	471-24-6869	6869
6	192374-933	4933
7	749-27-47-52	4752
8	461-75-4198	4198
9	892-36-676-2	6762
10	197-54-1646	1646
11	232-55-52-58	5258
12	898-84-336-6	3366
13	247-95-68-44	6844
14	415-48-894-3	8943
15	403-39-5-869	5869
16	399-83-05-03	0503
17	607-99-0411	0411
18	430-01-578-5	5785
19	571-09-6181	6181
20	478-32-02-87	0287
21		Not Specified
22		Not Specified
23	830-40-5287	5287
24	778-0845	Invalid Unique Id
25	368-44-4478	4478

Exercise 6

Get the count of users based on country code.

- Use the users table for this exercise.
- Output should contain the country code and count.
- There should be no + in the country code. It should only contain digits.
- Data should be sorted as numbers by country code.
- We should discard user_phone_no with null values.

Here is the desired output:

country_code	user_count
1	1
7	2
48	1
54	1
55	1
62	3
63	1
81	1
84	1
86	4
229	1
249	1
351	1
370	1
380	1
420	1
598	1

Exercise 7

Let us validate if we have an invalid order_item_subtotal as part of the order_items table.

- The order_items table has 6 fields.
  - order_item_id
  - order_item_order_id
  - order_item_product_id
  - order_item_quantity
  - order_item_subtotal
  - order_item_product_price
- order_item_subtotal is nothing but the product of order_item_quantity and order_item_product_price. It means order_item_subtotal is computed by multiplying order_item_quantity and order_item_product_price for each item.
- You need to get the count of order_items where order_item_subtotal is not equal to the product of order_item_quantity and order_item_product_price.
- There can be issues related to rounding off. Make sure it is taken care of using the appropriate function.

The output should be 0 as there are no such records.

Exercise 8

Get the number of orders placed on weekdays and weekends in the month of January 2014.

- Orders have 4 fields

    - order_id
    - order_date
    - order_customer_id
    - order_status
- Use the order date to determine the day on which orders are placed.
- Output should contain 2 columns - day_type and order_count.
- day_type should have 2 values: Week days and Weekend days.

Here is the desired output.

day_type	**order_count**
Weekend days	1505
Week days	4403