Github 地址:https://github.com/flower-pig/study.git

数据准备:

```
# 第一题
spark = SparkSession.builder\
    .appName('four')\
    .master('local[*]')\
    .getOrCreate()

sc = spark.sparkContext
data_rdd = sc.parallelize([('ABC17969(AB)', '1', 'ABC17969', 2022),
            ('ABC17969(AB)', '2', 'CDC52533', 2022),
            ('ABC17969(AB)', '3', 'DEC59161', 2023),
            ('ABC17969(AB)', '4', 'F43874', 2022),
            ('ABC17969(AB)', '5', 'MY06154', 2021),
            ('ABC17969(AB)', '6', 'MY4387', 2022),
            ('AE686(AE)', '7', 'AE686', 2023),
            ('AE686(AE)', '8', 'BH2740', 2021),
            ('AE686(AE)', '9', 'EG999', 2021),
            ('AE686(AE)', '10', 'AE0908', 2021),
            ('AE686(AE)', '11', 'QA402', 2022),
            ('AE686(AE)', '12', 'OM691', 2022)])

# peer_id, id_1, id_2, year.
schema = StructType()\
    .add('peer_id', StringType(), False)\
    .add('id_1', StringType(), False)\
    .add('id_2', StringType(), False)\
    .add('year', IntegerType(), False)

data_df = data_rdd.toDF(schema)
```

第一问代码截图:

```
# 1.For each peer_id, get the year when peer_id contains id_2, for example for 'ABC17969(AB)' year is 2022.
# 1. 对于每一个peer_id , 如果peer_id 中包含 id_2, 输出年份(这里多输出了'peer_id')
result_one = data_df.where(col('peer_id').contains(col('id_2'))).select('peer_id','year')
result_one.show()
```

第一问结果截图:

```
+------------+----+
|     peer_id|year|
+------------+----+
|ABC17969(AB)|2022|
|   AE686(AE)|2023|
+------------+----+


Process finished with exit code 0
```

第二问代码截图:

```
# 2.Given a size number, for example 3. For each peer_id count the number of each year (which is smaller or equal than the year in step1).
# 对每一个peer_id的年份计数，哪些是小于等于第一问中的年份
# For example, for 'ABC17969(AB)', the count should be:
# long 为 ABC17969(AB)
long_res_df = data_df.where((col('year') <= 2022) & (col('peer_id') == 'ABC17969(AB)')).groupby('peer_id', 'year').agg(
    count('peer_id').alias('num')).select('peer_id', 'year', 'num')
long_res_df.show()
# short 为 AE686(AE)
short_res_df = data_df.where((col('year') <= 2023) & (col('peer_id') == 'AE686(AE)')).groupby('peer_id', 'year').agg(
    count('peer_id').alias('num')).select('peer_id', 'year', 'num')
short_res_df.show()
```

第二问结果截图:

```
+------------+----+---+
|     peer_id|year|num|
+------------+----+---+
|ABC17969(AB)|2022|  4|
|ABC17969(AB)|2021|  1|
+------------+----+---+

+---------+----+---+
|  peer_id|year|num|
+---------+----+---+
|AE686(AE)|2023|  1|
|AE686(AE)|2021|  3|
|AE686(AE)|2022|  2|
+---------+----+---+


Process finished with exit code 0
```

第三问代码截图(数字给 3):

```
# 3.Order the value in step 2 by year and check if the count number of the first year is bigger or equal than the given size number. If yes, just return th
# If not, plus the count number from the biggest year to next year until the count number is bigger or equal than the given number.
# For example, for 'AE686(AE)', the year is 2023, and count are:

# 比如给的是3,5,7
give_number = 3
#窗口函数
window = Window.orderBy(desc('year')).rowsBetween(Window.unboundedPreceding, Window.currentRow)
# long 为 ABC17969(AB)
long_res_df = long_res_df.select('peer_id', 'year', 'num',
                                 when(sum('num').over(window).alias('s') >= give_number, give_number).otherwise(0).alias('partial_sum'),
                                 row_number().over(window).alias('rk'))

# 情况一: 对于ABC17969(AB)
res_one = long_res_df.where(col('partial_sum') == give_number).select(min('rk').alias('min_rk'))
min_rk = res_one.collect()[0]['min_rk']
if min_rk:
    result_three_partA = long_res_df.where(col('rk') <= min_rk).select('peer_id', 'year')
    result_three_partA.show()
else:
    long_res_df.select('peer_id', 'year').show()

# 情况二: 对于AE686(AE)
short_res_df = short_res_df.select('peer_id', 'year', 'num',
                                   when(sum('num').over(window).alias('s') >= give_number, give_number).otherwise(0).alias('partial_sum'),
                                   row_number().over(window).alias('rk'))
res_two = short_res_df.where(col('partial_sum') == give_number).select(min('rk').alias('min_rk'))
min_rk = res_two.collect()[0]['min_rk']
if min_rk:
    result_three_partB = short_res_df.where(col('rk') <= min_rk).select('peer_id', 'year')
    result_three_partB.show()
else:
    short_res_df.select('peer_id', 'year').show()
```

第三问结果截图:

```
+-------------+----+
|      peer_id|year|
+-------------+----+
|ABC17969(AB)|2022|
+-------------+----+
```

```
+----------+----+
|   peer_id|year|
+----------+----+
|AE686(AE)|2023|
|AE686(AE)|2022|
+----------+----+
```

如果数字改成 5:

```python
# 3.Order the value in step 2 by year and check if the count number of the first year is bigger or equal than the given size number. If yes, just return th
# If not, plus the count number from the biggest year to next year until the count number is bigger or equal than the given number.
# For example, for 'AE686(AE)', the year is 2023, and count are:

#比如给的是3,5,7
give_number = 3
# 窗口函数
window = Window.orderBy(desc('year')).rowsBetween(Window.unboundedPreceding, Window.currentRow)
# long 为 ABC17969(AB)
long_res_df = long_res_df.select('peer_id', 'year', 'num',
                                 when(sum('num').over(window).alias('s') >= give_number, give_number).otherwise(0).alias('partial_sum'),
                                 row_number().over(window).alias('rk'))

# 情况一: 对于ABC17969(AB)
res_one = long_res_df.where(col('partial_sum') == give_number).select(min('rk').alias('min_rk'))
min_rk = res_one.collect()[0]['min_rk']
if min_rk:
    result_three_partA = long_res_df.where(col('rk') <= min_rk).select('peer_id', 'year')
    result_three_partA.show()
else:
    long_res_df.select('peer_id', 'year').show()

# 情况二: 对于AE686(AE)
short_res_df = short_res_df.select('peer_id', 'year', 'num',
                                   when(sum('num').over(window).alias('s') >= give_number, give_number).otherwise(0).alias('partial_sum'),
                                   row_number().over(window).alias('rk'))
res_two = short_res_df.where(col('partial_sum') == give_number).select(min('rk').alias('min_rk'))
min_rk = res_two.collect()[0]['min_rk']
if min_rk:
    result_three_partB = short_res_df.where(col('rk') <= min_rk).select('peer_id', 'year')
    result_three_partB.show()
else:
    short_res_df.select('peer_id', 'year').show()
```

结果:

```
+-------------+----+
|      peer_id|year|
+-------------+----+
|ABC17969(AB)|2022|
|ABC17969(AB)|2021|
+-------------+----+
```

```
+---------+----+
|  peer_id|year|
+---------+----+
|AE686(AE)|2023|
|AE686(AE)|2022|
|AE686(AE)|2021|
+---------+----+
```

如果数字改成 7:

```
# 3.Order the value in step 2 by year and check if the count number of the first year is bigger or equal than the given size number. If yes, just return th
# If not, plus the count number from the biggest year to next year until the count number is bigger or equal than the given number.
# For example, for 'AE686(AE)', the year is 2023, and count are:

# 比如给的是7
give_number = 7
# 窗口函数
window = Window.orderBy(desc('year')).rowsBetween(Window.unboundedPreceding, Window.currentRow)
# long 为 ABC17969(AB)
long_res_df = long_res_df.select('peer_id', 'year', 'num',
                                 when(sum('num').over(window).alias('s') >= give_number, give_number).otherwise(0).alias('partial_sum'),
                                 row_number().over(window).alias('rk'))

# 情况一：对于ABC17969(AB)
res_one = long_res_df.where(col('partial_sum') == give_number).select(min('rk').alias('min_rk'))
min_rk = res_one.collect()[0]['min_rk']
if min_rk:
    result_three_partA = long_res_df.where(col('rk') <= min_rk).select('peer_id', 'year')
    result_three_partA.show()
else:
    long_res_df.select('peer_id', 'year').show()

# 情况二：对于AE686(AE)
short_res_df = short_res_df.select('peer_id', 'year', 'num',
                                   when(sum('num').over(window).alias('s') >= give_number, give_number).otherwise(0).alias('partial_sum'),
                                   row_number().over(window).alias('rk'))
res_two = short_res_df.where(col('partial_sum') == give_number).select(min('rk').alias('min_rk'))
min_rk = res_two.collect()[0]['min_rk']
if min_rk:
    result_three_partB = short_res_df.where(col('rk') <= min_rk).select('peer_id', 'year')
    result_three_partB.show()
else:
    short_res_df.select('peer_id', 'year').show()
```

结果是:

```
+-------------+----+
|      peer_id|year|
+-------------+----+
|ABC17969(AB)|2022|
|ABC17969(AB)|2021|
+-------------+----+
```

```
+----------+----+
|   peer_id|year|
+----------+----+
|AE686(AE)|2023|
|AE686(AE)|2021|
|AE686(AE)|2022|
+----------+----+
```