8강–출석인증–테이블과 뷰 활용하기

연습문제 1

- 다양한 주기별 통계 뷰 만들기

1. 대여소별, **일자별 통계 뷰(view_stat_date)**를 만듭니다.

```
scala> spark.sql("SELECT * FROM view_stat_date").show(31)
21/04/28 19:17:18 WARN CSVHeaderChecker: CSV header does not conform to the schema.
 Header: 대여일시, 이용시간, 이용거리
 Schema: rentDate, useTime, useDistance
Expected: rentDate but found: 대여일시
CSV file: file:///Users/jonghyun/Workspace/SparkStudy/data/seoul_bike.csv
+----------+--------+-------+-------+-------------+-------+
|      date|useCount|timeSum|timeAvg|       distSum|distAvg|
+----------+--------+-------+-------+-------------+-------+
|2021-01-01|   22119| 692309|   31.3| 4.613528774E7|2085.78|
|2021-01-02|   23463| 686580|  29.26| 4.409618053E7|1879.39|
|2021-01-03|   21656| 609571|  28.15| 4.059830028E7|1874.69|
|2021-01-04|   32732| 750721|  22.94|  5.24964523E7|1603.83|
|2021-01-05|   28819| 610227|  21.17| 6.944855707E7|2409.82|
|2021-01-06|   22104| 463066|  20.95| 5.156691932E7|2332.92|
|2021-01-07|    5471| 105257|  19.24|    9772955.18|1786.32|
|2021-01-08|    7896| 154154|  19.52|  1.42234488E7|1801.35|
|2021-01-09|    9017| 204153|  22.64| 1.980416647E7|2196.31|
|2021-01-10|   11042| 280368|  25.39| 3.047287464E7|2759.72|
|2021-01-11|   19366| 414533|  21.41|  4.45637381E7|2301.13|
|2021-01-12|   12739| 271112|  21.28| 2.861225035E7|2246.04|
|2021-01-13|   19652| 491731|  25.02| 5.118574495E7|2604.61|
|2021-01-14|   32375| 797888|  24.65| 9.016120043E7| 2784.9|
|2021-01-15|   32218| 743448|  23.08| 8.325690373E7|2584.17|
|2021-01-16|   22370| 584137|  26.11| 6.683361112E7|2987.64|
|2021-01-17|   17951| 482819|   26.9| 5.646926759E7|3145.74|
|2021-01-18|   17983| 341429|  18.99| 3.828439548E7|2128.92|
|2021-01-19|   26706| 568605|  21.29| 6.426829344E7|2406.51|
|2021-01-20|   36728| 867109|  23.61|1.0169938074E8|2768.99|
|2021-01-21|   17329| 348146|  20.09| 4.018453435E7|2318.92|
|2021-01-22|   34749| 841057|   24.2|  9.51663558E7|2738.68|
|2021-01-23|   43127|1363827|  31.62|1.5846975228E8|3674.49|
|2021-01-24|   49763|1877677|  37.73|2.2250635301E8|4471.32|
|2021-01-25|   54324|1494838|  27.52|1.7902169596E8|3295.44|
|2021-01-26|   26066| 572431|  21.96| 6.922167318E7|2655.63|
|2021-01-27|   45370|1125197|   24.8|1.3369043384E8|2946.67|
|2021-01-28|   22598| 428268|  18.95| 4.939046847E7|2185.61|
|2021-01-29|   24228| 483034|  19.94|  5.31181781E7|2192.43|
|2021-01-30|   28198| 791971|  28.09| 8.945983334E7|3172.56|
|2021-01-31|   38344|1276238|  33.28| 1.504205305E8|3922.92|
+----------+--------+-------+-------+-------------+-------+
```

2. 대여소별, **요일별 통계 뷰(view_stat_week)**를 만듭니다.

```
scala> spark.sql("SELECT * FROM view_stat_week").show(7)
21/04/28 19:19:37 WARN CSVHeaderChecker: CSV header does not conform to the schema.
 Header: 대여일시, 이용시간, 이용거리
 Schema: rentDate, useTime, useDistance
Expected: rentDate but found: 대여일시
CSV file: file:///Users/jonghyun/Workspace/SparkStudy/data/seoul_bike.csv
+-------+--------+-------+-------+-------------+-------+
|dayNumb|useCount|timeSum|timeAvg|      distSum|distAvg|
+-------+--------+-------+-------+-------------+-------+
|      1|  138756|4526673|  32.62|5.0046732602E8|3606.82|
|      2|  124405|3001521|  24.13|3.1436628183E8|2526.96|
|      3|   94330|2022375|  21.44|2.3155077405E8|2454.69|
|      4|  123854|2947103|  23.79|3.3814247883E8|2730.17|
|      5|   77773|1679559|   21.6|1.8950915843E8| 2436.7|
|      6|  121210|2914002|  24.04|2.9190017417E8|2408.22|
|      7|  126175|3630668|  28.77|3.7866354374E8| 3001.1|
+-------+--------+-------+-------+-------------+-------+
```

3. 대여소별, **시간별 통계 뷰(view_stat_hour)**를 만듭니다.

```
scala> spark.sql("SELECT * FROM view_stat_hour").show(24)
21/04/28 19:19:57 WARN CSVHeaderChecker: CSV header does not conform to the schema.
 Header: 대여일시, 이용시간, 이용거리
 Schema: rentDate, useTime, useDistance
Expected: rentDate but found: 대여일시
CSV file: file:///Users/jonghyun/Workspace/SparkStudy/data/seoul_bike.csv
+----+--------+------+------+-------------+-------+
|hour|useCount|timeSum|timeAvg|      distSum|distAvg|
+----+--------+------+------+-------------+-------+
|   0|   12528| 254360|  20.3| 3.148547152E7|2513.21|
|   1|    9993| 206710| 20.69|  2.55476709E7|2556.56|
|   2|    6900| 141925| 20.57| 1.830724529E7|2653.22|
|   3|    4991|  97925| 19.62|  1.24666748E7|2497.83|
|   4|    4338|  81605| 18.81| 1.089382566E7|2511.26|
|   5|    5944|  99302| 16.71| 1.242315088E7|2090.03|
|   6|   11595| 202946|  17.5| 2.703124667E7|2331.28|
|   7|   27967| 466983|  16.7| 6.475198153E7| 2315.3|
|   8|   52632| 759845| 14.44|1.0021607944E8|1904.09|
|   9|   30628| 573172| 18.71| 6.317456516E7|2062.64|
|  10|   26141| 669573| 25.61| 6.627405473E7|2535.25|
|  11|   34880| 993804| 28.49| 9.092146694E7|2606.69|
|  12|   43401|1251758| 28.84|1.2006725827E8|2766.46|
|  13|   48457|1516495|  31.3|1.4884311933E8|3071.65|
|  14|   53756|1845069| 34.32|1.8736868919E8|3485.54|
|  15|   58149|1975721| 33.98|2.0168738333E8|3468.46|
|  16|   59871|1858163| 31.04|1.9515604613E8|3259.61|
|  17|   67785|1815492| 26.78|1.9640431712E8|2897.46|
|  18|   75131|1783587| 23.74| 2.010991999E8|2676.65|
|  19|   49514|1222610| 24.69|1.2946809676E8|2614.78|
|  20|   42378|1050426| 24.79|1.1975491561E8|2825.87|
|  21|   41238| 994670| 24.12|1.1774820088E8|2855.33|
|  22|   23140| 532512| 23.01| 6.434252416E7|2780.58|
|  23|   15146| 327248| 21.61| 3.916655286E7|2585.93|
+----+--------+------+------+-------------+-------+
```

4. 'show tables' 실행결과

```
scala> spark.sql("SHOW TABLES").show
+--------+-------------+-----------+
|database|    tableName|isTemporary|
+--------+-------------+-----------+
| default|      airobic|      false|
| default|view_stat_date|      false|
| default|view_stat_hour|      false|
| default|view_stat_week|      false|
+--------+-------------+-----------+
```

코드는 아래와 같습니다. rentDate, returnDate를 Timestamp 타입으로, useDistance를 float 타입으로 선언하였습니다.

```
spark.sql("""
  CREATE TABLE AiRoBiC (
     bicNumber      String    COMMENT '자전거번호',
     rentDate       Timestamp COMMENT '대여일자',
     rentStatId     String    COMMENT '대여소번호',
     rentStatName   String    COMMENT '대여소이름',
     rentParkId     String    COMMENT '거치대번호',
     returnDate     Timestamp COMMENT '반납일자',
     returnStatId   String    COMMENT '반납대여소번호',
     returnStatName String    COMMENT '반납대여소이름',
     returnParkId   String    COMMENT '반납거치대번호',
     useTime        Long      COMMENT '사용시간(분)',
     useDistance    Float     COMMENT '사용거리(미터)')
  USING csv OPTIONS (
     header true,
     path '../data/seoul_bike.csv')
""")

spark.sql("DESC AIRoBiC").show

spark.sql("""
  CREATE OR REPLACE VIEW view_stat_date AS
    SELECT DATE(rentDate) AS date,
          COUNT(*) AS useCount,
          SUM(useTime) AS timeSum,
          ROUND(AVG(useTime), 2) AS timeAvg,
          ROUND(SUM(useDistance), 2) AS distSum,
          ROUND(AVG(useDistance), 2) AS distAvg
    FROM AiRoBiC
    GROUP BY date
    ORDER BY date
""")

spark.sql("SELECT * FROM view_stat_date").show(31)
```

```
spark.sql("""
  CREATE OR REPLACE VIEW view_stat_week AS
    SELECT DAYOFWEEK(rentDate) AS dayNumb,
          COUNT(*) AS useCount,
          SUM(useTime) AS timeSum,
          ROUND(AVG(useTime), 2) AS timeAvg,
          ROUND(SUM(useDistance), 2) AS distSum,
          ROUND(AVG(useDistance), 2) AS distAvg
    FROM AiRoBiC
    GROUP BY dayNumb
    ORDER BY dayNumb
""")

spark.sql("SELECT * FROM view_stat_week").show(7)

spark.sql("""
  CREATE OR REPLACE VIEW view_stat_hour AS
    SELECT HOUR(rentDate) AS hour,
          COUNT(*) AS useCount,
          SUM(useTime) AS timeSum,
          ROUND(AVG(useTime), 2) AS timeAvg,
          ROUND(SUM(useDistance), 2) AS distSum,
          ROUND(AVG(useDistance), 2) AS distAvg
    FROM AiRoBiC
    GROUP BY hour
    ORDER BY hour
""")

spark.sql("SELECT * FROM view_stat_hour").show(24)

spark.sql("SHOW TABLES").show
```

연습문제 2

- 상기 통계 뷰를 활용한 이용패턴 찾기

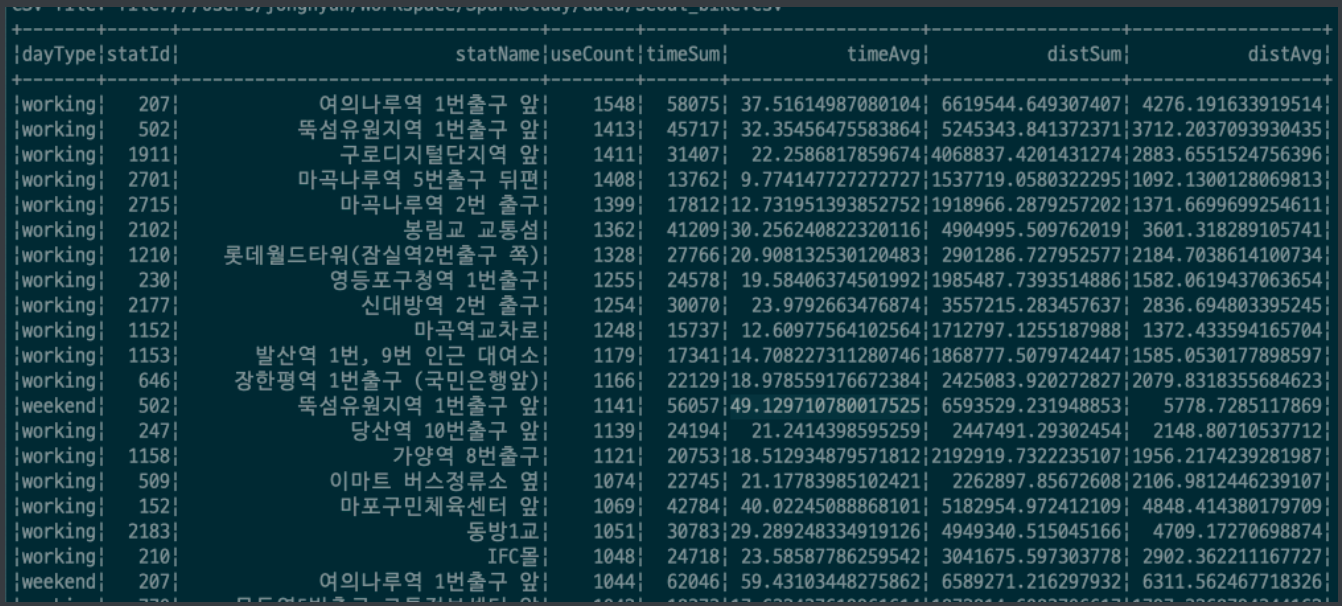
1. 전체 대여소의 일자별, 요일별, 시간별 이용패턴을 찾고, 설명해보자.

   일단 가장 눈에 띄는 것은 요일별 패턴이다. 합계보다는 평균에 주목해서 보았을 때, 일요일은 다른 요일에 비해 이용 시간 평균이 길다는 점을 볼 수 있다. 또한 평균 이용 거리는 주말에 해당하는 요일이 평일에 비해서 더 많다는 것을 확인할 수 있다. 아마도 평일에는 출퇴근 및 근처를 이동하기 위해 따릉이를 사용하는 반면 주말은 여가를 위해 한강변을 달리거나 하는 경우가 반영되었기 때문으로 추정된다.

   시간별 패턴 은 너무도 당연하게 심야 및 새벽 시간대 보다는 낮 시간대 이용이 활발함을 볼 수 있다. 그럼에도 새벽 시간대 이용이 적다고 할 수는 없는데, 대중교통 운행이 종료된 상황에서 따릉이를 대안으로 선택하기 때문으로 보인다. 또한 따릉이 이용이 가장 많은 시간은 17 ~ 18시이다. 또한 19시부터 21시까지 그 수요는 어느 정도 유지되는 것으로 보아 출근 전에 자전거를 타고자하는 수요보다는 퇴근 이후 자전거를 타고자하는 수요가 많다는 것을 추측해볼 수 있다. 물론 퇴근 길에 따릉이를 이용했을 수도 있으나 해당 데이터만으로 판단하기는 어려워서 보수적으로 주장하였다.

   마지막으로 일자별 패턴인데 1월 7일, 8일, 9일이 유독 따릉이 사용량이 적었음을 알 수 있다. 이 날은 서울에 강추위가 닥친 날로 8일에는 최저 기온이 -18.6 도 였음을 확인할 수 있다. 개인적으로 그럼에도 불구하고 수요가 8,000건이나 있었다는 사실이 놀랍기만 하다.

2. 일자별로 특이한 이용패턴을 보이는 대여소와 그 이유를 추정해보자.



dayType	statId	statName	useCount	timeSum	timeAvg	distSum	distAvg
working	207	여의나루역 1번출구 앞	1548	58075	37.51614987080104	6619544.649307407	4276.191633919514
working	502	뚝섬유원지역 1번출구 앞	1413	45717	32.35456475583864	5245343.841372371	3712.2037093930435
working	1911	구로디지털단지역 앞	1411	31407	22.2586817859674	4068837.4201431274	2883.6551524756396
working	2701	마곡나루역 5번출구 뒤편	1408	13762	9.774147727272727	1537719.0580322295	1092.1300128069813
working	2715	마곡나루역 2번 출구	1399	17812	12.731951393852752	1918966.2879257202	1371.6699699254611
working	2102	봉림교 교통섬	1362	41209	30.256240822320116	4904995.509762019	3601.318289105741
working	1210	롯데월드타워(잠실역2번출구 쪽)	1328	27766	20.908132530120483	2901286.727952577	2184.7038614100734
working	230	영등포구청역 1번출구	1255	24578	19.58406374501992	1985487.7393514886	1582.0619437063654
working	2177	신대방역 2번 출구	1254	30070	23.9792663476874	3557215.283457637	2836.694803395245
working	1152	마곡역교차로	1248	15737	12.60977564102564	1712797.1255187988	1372.433594165704
working	1153	발산역 1번, 9번 인근 대여소	1179	17341	14.708227311280746	1868777.5079742447	1585.0530177898597
working	646	장한평역 1번출구 (국민은행앞)	1166	22129	18.978559176672384	2425083.920272827	2079.8318355684623
weekend	502	뚝섬유원지역 1번출구 앞	1141	56057	49.129710780017525	6593529.231948853	5778.7285117869
working	247	당산역 10번출구 앞	1139	24194	21.2414398595259	2447491.29302454	2148.80710537712
working	1158	가양역 8번출구	1121	20753	18.512934879571812	2192919.7322235107	1956.2174239281987
working	509	이마트 버스정류소 옆	1074	22745	21.17783985102421	2262897.85672608	2106.9812446239107
working	152	마포구민체육센터 앞	1069	42784	40.02245088868101	5182954.972412109	4848.414380179709
working	2183	동방1교	1051	30783	29.289248334919126	4949340.515045166	4709.17270698874
working	210	IFC몰	1048	24718	23.58587786259542	3041675.597303778	2902.362211167727
weekend	207	여의나루역 1번출구 앞	1044	62046	59.43103448275862	6589271.216297932	6311.562467718326

   모든 일자를 확인하기 어려워 주말과 주중으로 비교하였다. 사용한 코드는 아래와 같다.

```
spark.sql("""
```

```
    SELECT CASE WHEN dayofweek(rentDate) = 1 THEN 'weekend'
                WHEN dayofweek(rentDate) = 2 THEN 'working'
                WHEN dayofweek(rentDate) = 3 THEN 'working'
                WHEN dayofweek(rentDate) = 4 THEN 'working'
                WHEN dayofweek(rentDate) = 5 THEN 'working'
                WHEN dayofweek(rentDate) = 6 THEN 'working'
                WHEN dayofweek(rentDate) = 7 THEN 'weekend'
                ELSE 'none'
           END AS dayType,
           rentStatId as statId,
           rentStatName as statName,
           COUNT(*) AS useCount,
           SUM(useTime) AS timeSum,
           AVG(useTime) AS timeAvg,
           SUM(useDistance) AS distSum,
           AVG(useDistance) AS distAvg
      FROM AiRoBiC
     GROUP BY 1, statId, statName
     ORDER BY useCount DESC
""").show(100)
```

이때 station ID가 207번 "여의나루역 1번출구 앞"과 502번 "뚝섬유원지역 1번출구 앞" 이 흥미로운 패턴을 보였다. working day에 여의나루역은 사용 횟수가 1,548회, 뚝섬유원지역은 1,413회를 기록하였다. 그러나 주말에는 여의나루역과 뚝섬유원지역 모두 1,044회, 1,141회로 감소하였다. 하지만 이용 평균 시간과 평균 거리는 주말이 주중에 비해 모두 높아졌다. 이는 주중과 주말에 따릉이 이용에 대한 목적이 다르다는 것을 잠정적으로 암시한다. 추가적으로 본 결과는 가장 따릉이가 활발하게 사용되고 있는 역에 대해서 조회를 했는데, 대부분 역 인근에 위치하고 있었고, 이는 대중교통의 보조 수단으로서 따릉이를 사용할 수 있다는 점을 암시한다. 주말과 주중 모두 유사하게 따릉이를 이용하는 역은 앞서 이야기한 여의나루역, 뚝섬유원지역처럼 주중 출퇴근 수요, 주말 여가 수요가 골고루 있음직한 곳으로 추정된다.

3. 요일별로 주목할 만한 이용패턴과 상관관계를 설명할 수 있는 요인을 추천해보자.

   요일별로 주목할만한 이용 패턴은 주말이 주중에 비해 더 오래, 더 길게 탄다는 점 등이 특징이다. 주말의 패턴은 날씨에 크게 영향을 받을 것으로 추측된다. 비가 오거나 추운 날 여가 시간을 자전거로 타고자 하는 사람은 많지 않을 것이다.

4. 시간별 이용패턴과 그에 대한 설명

새벽 시간대 수요는 대중 교통의 대체재로써의 성격이 강할 것으로 보인다. 또한 오전보다 오후에 따릉이의 수요가 높은데, 이는 퇴근 후 여가를 즐기거나 출근용으로 따릉이를 쓰는 것보다는 퇴근용으로 쓸 가능성이 높다는 점을 시사한다. 오전 8시에 갑작스레 사용 횟수가 증가한다. 또한 오전 8시는 평균 이동거리가 모든 시간과 비교했을 때, 가장 짧은데, 출근 시간이라는 점을 고려하면 이는 해당 시간에는 역에서 회사, 학교 등 걸어가기에는 멀지만 자전거 혹은 버스를 이용할 만한 거리에 대한 수요가 있을 것이라는 점을 추측해볼 수 있다.