

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 4

 긴급 점검이 필요한 스마트카 정보

# 과제 III

다음 페이지 부터 내용들을 실습해서 주제영역 4

**“긴급 점검이 필요한 스마트카 정보”** 마트를 생성 하세요.

## 주제 영역 4. 긴급 점검이 필요한 스마트카 정보 - 워크플로 작성 \_ 실습

주제 영역 4의 워크플로는 2020년 03월 22일 스마트카의 다양한 센서로부터 수집된 데이터(타이어, 라이트, 브레이크, 엔진, 배터리)를 분석해 긴급 점검이 필요한 스마트카 차량 리스트를 찾아내는 것이다. 이번 하이브 쿼리 역시, 파일럿 프로젝트에서 실행하기에는 다소 무겁고 시간이 많이 소요되는 맵리듀스 잡들을 만들어낸다. 하지만 이번 작업을 통해 “긴급 점검 차량” 마트 데이터를 만들면 이후부터는 단순 쿼리로 빠르게 긴급 점검 차량들을 조회해 볼 수 있게 된다. 워크플로의 하이브 작업에 사용되는 하이브 QL은 C://예제소스/bigdata2nd-master/CH06/HiveQL/에서 확인할 수 있으므로 필요 시 해당 파일을 열어서 복사한 후 붙여넣는 식으로 활용한다.

01. 휴의 좌측 드롭박스 메뉴에서 [문서]를 선택해 [내 문서]에 생성해 놓은 주제 영역 4의 작업 디렉터리로 이동한다.

- 휴 내 문서: /workflow/hive\_script/subject4

02. 주제 영역 4에서는 하이브 스크립트 파일 2개를 작성한다. 먼저 내 문서의 /workflow/hive\_script/subject4로 이동한 후 [새 문서] → [Hive 쿼리]를 선택한다.

03. 하이브 에디터가 나타나면 스마트카 장비의 상태를 관리하기 위한 테이블 생성 스크립트를 작성하고 저장한다. 파일 이름은 create\_table\_managed\_smartcar\_emergency\_check\_info.hql로 입력한다.

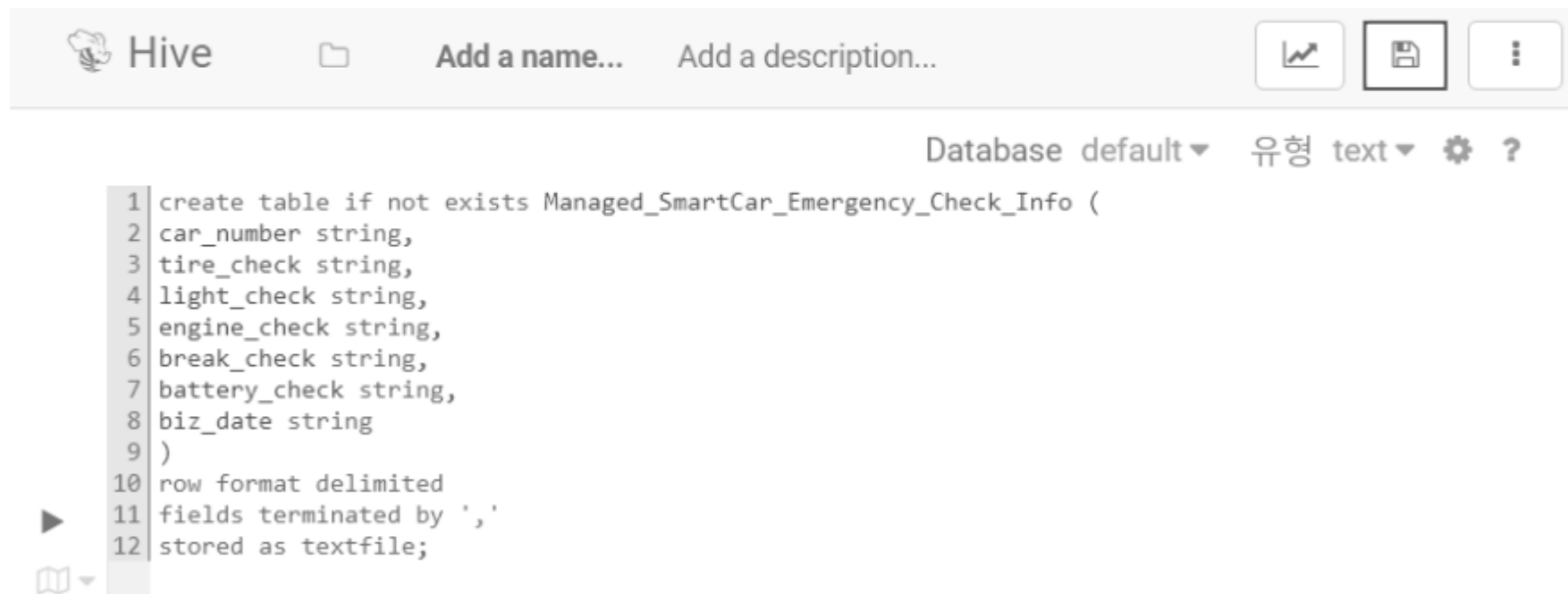



그림 6.110 주제 영역 4의 Managed 테이블을 생성하는 하이브 쿼리




04. 계속해서 내 문서의 /workflow/hive\_script/subject4에 두 번째 하이브 스크립트 파일을 만든다. “subject4” 디렉터리에서 [새 문서] → [Hive 쿼리]를 선택한다.

05. 하이브 에디트 창이 나타나면 긴급 점검이 필요한 스마트카를 Select/Insert하는 쿼리를 그림 6.111처럼 작성한 후 저장한다. 파일 이름은 insert\_table\_managed\_smartcar\_emergency\_check\_info.hql로 입력한다.

그림 6.111의 하이브 쿼리를 실행하면 스마트카의 6가지 점검 상태를 분석해서 그 결과를 “스마트카 긴급 점검 차량 정보(Managed\_SmartCar\_Emergency\_Check\_Info)” 마트 테이블에 저장한다. 쿼리의 내용을 자세히 살펴보면, 주요 필드들을 가공해 스마트카 상태 점검을 위한 새로운 변수를 만들어 내는 피처 엔지니어링(Feature Engineering) 쿼리로 구성돼 있다. 또한 RDBMS에서 자주 사용하는 Left Outer 조인 쿼리가 사용되는 것을 확인할 수 있다. 참고로 해당 하이브 QL은 대용량 쿼리로서, 실행 중에 8개 이상의 맵리듀스 잡과 태스크가 만들어지고 총 실행 시간은 10여 분 정도 된다.

 Hive

Add a nam... Add a des...



```
1 insert into table Managed_SmartCar_Emergency_Check_Info
2
3 select
4     t1.car_number,
5     t2.symptom as tire_symptom,
6     t3.symptom as light_symptom,
7     t4.symptom as engine_symptom,
8     t5.symptom as break_symptom,
9     t6.symptom as battery_symptom,
10    t1.biz_date
11 from
12     (select distinct car_number as car_number, biz_date
13      from managed_smartcar_status_info where biz_date = '${working_day}') t1
```

(계속)

```

14
15 left outer join ( select
16     car_number,
17     avg(tire_fl) as tire_fl_avg ,
18     avg(tire_fr) as tire_fr_avg ,
19     avg(tire_bl) as tire_bl_avg ,
20     avg(tire_br) as tire_br_avg ,
21     'Tire Check' as symptom
22     from managed_smartcar_status_info where biz_date = '${working_day}'
23     group by car_number
24     having  tire_fl_avg < 80
25            or tire_fr_avg < 80
26            or tire_bl_avg < 80
27            or tire_br_avg < 80 ) t2
28 on t1.car_number = t2.car_number

```

타이어 점검

```

29
30 left outer join ( select
31     distinct car_number,
32     'Light Check' as symptom
33     from managed_smartcar_status_info
34     where biz_date = '${working_day}' and (light_fl = '2'
35                                            or light_fr = '2'
36                                            or light_bl = '2'
37                                            or light_br = '2')) t3
38 on t1.car_number = t3.car_number
39

```

라이트 점검

(계속)



06. 이제 워크플로를 만든다. 휴의 상단 쿼리 콤보박스에서 [스케줄러] → [Workflow]를 선택해서 우지 편집기를 실행한다.
07. 첫 번째 작업으로 워크플로의 작업 툴박스에서 “Hive 쿼리” 작업을 워크플로의 첫 번째 작업 노드에 드래그 앤드 드롭한다.
08. 사용할 Hive 스크립트 파일을 선택한다. 앞 단계에서 만든 내 문서의 /workflow/hive\_script/subject4에 create\_table\_smartcar\_emergency\_check\_info.hql을 선택한 후 [추가] 버튼을 클릭한다.
09. 두 번째 작업을 위해 워크플로의 작업 툴박스에서 “Hive 쿼리” 작업을 워크플로의 두 번째 작업 노드에 드래그 앤드 드롭한다.
10. 사용할 Hive 스크립트 파일을 선택한다. 앞 단계에서 만든 내 문서의 /workflow/hive\_script/subject4에 insert\_table\_managed\_smartcar\_emergency\_check\_info.hql을 선택한 후 [추가] 버튼을 클릭한다.
11. [매개변수+]를 누르고 working\_day의 매개변수에 우지의 예약 스케줄러에서 정의할 \${today} 매개변수를 할당한다.
  - working\_day=\${today} ※ 즉시 실행시 - 스마트카 시뮬레이션 날짜(교재기준: 20200322)로 설정  
ex) working\_day=20200322



12. 워크플로의 이름을 작성한다. 워크플로 상단의 [My Workflow]를 클릭하고, “Subject 4 – Workflow”로 변경한 후 [확인] 버튼을 클릭한다.
13. Workflow 작성을 완료 한다. 우측 상단의 [저장] 버튼을 클릭한다.
14. 이제 작성한 워크플로를 작동하기 위한 예약 작업을 생성한다. 상단의 쿼리 콤보박스에서 [스케줄러] → [예약] 메뉴를 선택한다.
15. 먼저 예약 작업 이름을 입력한다. 상단의 [My Scheduler]를 클릭하고 “Subject 4 – 예약”으로 입력한다.
16. 예약 작업이 사용할 워크플로를 선택한다. “예정된 Workflow는 무엇입니까?”라는 메시지 하단에 있는 “Workflow 선택...”을 클릭해 앞서 만든 주제 영역 4의 워크플로인 “Subject 4 – Workflow”를 선택한다.
17. 예약 작업이 워크플로를 실행하기 위한 스케줄 값을 입력한다.
  - 실행 간격: 매일, 04시
  - 시작 일자: 2020년 03월 23일, 00시 00분
  - 종료 일자: 2020년 12월 31일, 23시 59분
  - 시간대: Asia/Seoul



18. 워크플로에서 사용할 매개변수인 today 값을 예약 작업의 매개변수로 정의한다.

앞서 워크플로의 하이브 작업에서는 매개변수를 “working\_day=\${today}”로 등록했다. today의 값을 예약 작업의 내장 함수를 통해 설정한다.






```
${coord:formatTime(coord:dateTzOffset(coord:nominalTime(), "Asia/Seoul"), 'yyyyMMdd')}
```

19. 우지의 예약 작업 설정이 모두 끝났다. [저장] 버튼을 클릭해 작성을 완료한다.


20. 작성이 완료된 예약 작업을 우측 상단의 [제출] 버튼을 클릭해 실행한다.


21. 제출된 예약 작업 상태를 확인해 보자. 좌측 상단의 드롭박스 메뉴를 클릭해 [Job] → [일정]을 차례로 선택한다. 앞서 등록한 “Subject 4 – 예약”이 “Running” 상태로, 매일 새벽 04시가 되면 등록된 워크플로를 작동시키게 된다. 새벽 4시까지 기다릴 수 없으니 앞서 주제 영역 1에서 소개한 “워크플로 즉시 실행해 보기”를 참고해 곧바로 실행해 본다.

22. “Subject 4 – Workflow”가 정상적으로 작동했는지 확인한다. 휴의 Hive Editor로 이동해서 그림 6.112와 같이 하 이브 QL을 작성해서 실행한다. “2020년 03월 22일 긴급 점검이 필요한 스마트카 차량”의 정보가 조회될 것이다. 필자의 경우 33대의 차량에 대해 긴급 점검 대상으로 조회됐다. “biz\_date=20220322”의 날짜는 독자들의 파일럿 환경의 날짜로 맞춰야 한다.


 Hive  Add a name... Add a description...   

```
1 select * from managed_smartcar_emergency_check_info
2 where biz_date = '20200322'
```





쿼리 기록

저장된 쿼리 

결과 (35)

	car_number	tire_check	light_check	engine_check	break_check	battery_check
1	A0090	NULL	NULL	Engine Check	NULL	NULL
2	C0078	Tire Check	NULL	NULL	NULL	NULL
3	E0001	NULL	NULL	Engine Check	NULL	NULL
4	H0049	Tire Check	NULL	NULL	NULL	NULL
5	H0094	Tire Check	NULL	NULL	NULL	NULL

그림 6.112 주제 영역 4 워크플로의 실행 결과 확인

주제 영역 4에서는 긴급 점검이 필요한 스마트카 정보를 찾아봤다. 이때 긴급 점검 대상 차량을 판단하기 위해 기준값들이 사용되며, 과거 운행 데이터로부터 탐색적 분석을 진행해 관련 지표를 정의한다. 지표의 신뢰도를 높이기 위해 대규모 데이터를 확보하고, 관련 업무 전문가로부터 영향력 있는 변수를 도출하는 작업도 진행된다.