

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

현재 하이브의 External에는 2020년 03월 22일 날짜로 “스마트카 상태 정보 데이터”가 적재돼 있다. 이 데이터를 우지 워크플로를 이용해 Managed 영역으로 매일 옮기고, 이때 “스마트카 마스터 데이터”와 Join을 통해 데이터를 통합하는 작업을 한다. 워크플로의 하이브 작업에 사용될 하이브 QL은 C://예제소스/bigdata2nd-master/CH06/HiveQL/ 경로에서 확인할 수 있다.

저사양 파일럿 환경: 우지 서비스를 시작한다.

▪ 우지 서비스: CM 홈 → [Oozie] → [시작]

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

01. 주제 영역 1에서 사용할 작업 디렉토리를 만든다.

- 휴의 좌측 드롭박스 메뉴에서 [문서] 메뉴를 선택해 내 문서 기능을 실행한다.



- 하이브 스크립트 파일을 저장하기 위한 작업 폴더를 "workflow"라는 이름으로 생성한다.



# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

- workflow 폴더 밑에 하위 작업 폴더로 "hive\_script"를 추가로 생성한다(그림 6.66 참고).
- hive\_script 폴더 밑에 하위 작업 폴더로 "subject1"을 생성한다.
- hive\_script 폴더 밑에 나머지 주제 영역에서도 사용할 subject2, subject3, subject4, subject5 폴더를 각각 생성한다.

<div> <div>내 문서 &gt; workflow &gt; hive_script</div> <div>모두 🔍</div> <div>📁</div> <div>🔍</div> <div>⋮</div> <div>🔄 휴지통</div> </div>			
이름	설명	유형	소유자
📁 subject1		디렉토리	admin
📁 subject2		디렉토리	admin
📁 subject3		디렉토리	admin
📁 subject4		디렉토리	admin
📁 subject4		디렉토리	admin

그림 6.66 주제 영역에 사용할 디렉터리 생성

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

02. 주제 영역 1에서 사용할 하이브 스크립트 파일 3개를 만든다. 먼저 내 문서에서 /workflow/hive\_script/subject1 위치로 이동한다. [새 문서] → [Hive 쿼리]를 선택한다.



그림 6.67 주제 영역 1에 사용할 하이브 파일 생성

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

03. 하이브 쿼리를 작성할 수 있는 에디터가 그림 6.68처럼 활성화된다.

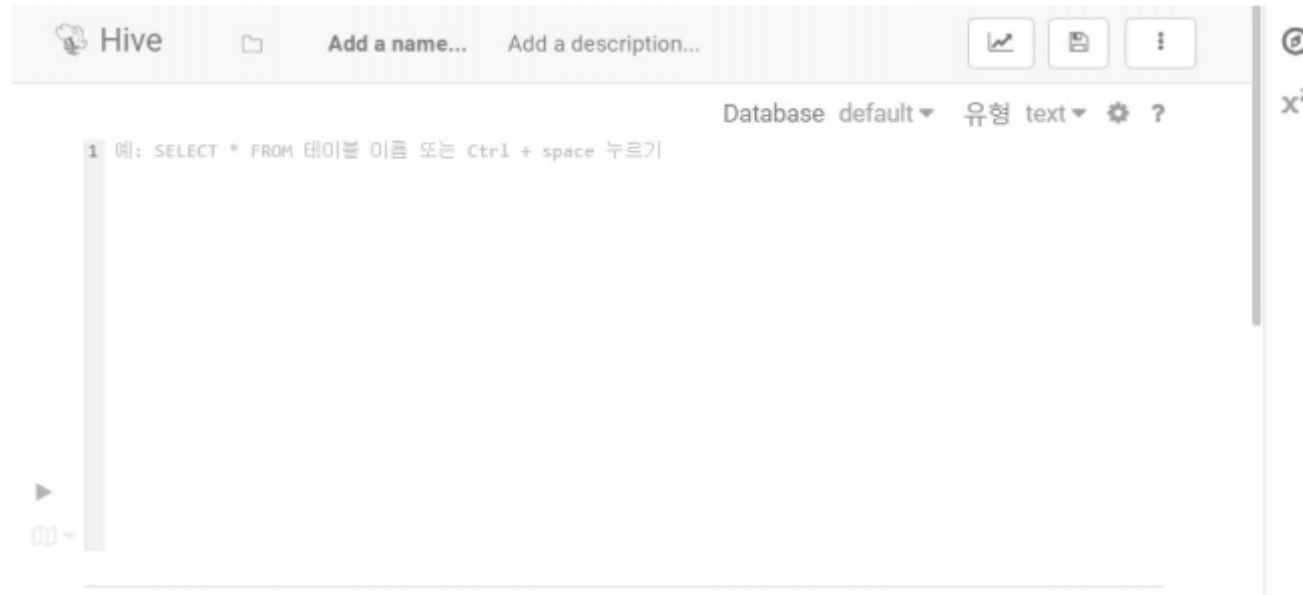


그림 6.68 하이브 쿼리 에디터

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

04. 하이브 에디트 창에 그림 6.69처럼 “Create Table..” DDL 문장을 입력하고 상단의 [저장] 버튼을 클릭한다. 파일명 입력창이 나타나면 파일명을 “create\_table\_managed\_smartcar\_status\_info.hql”로 입력하고 [Save] 버튼을 클릭한다. 하이브 QL은 C://예제소스/bigdata2nd-master/CH06/HiveQL/그림-6.69.hql로 제공되므로 참고한다.



Hive



Add a name...

Add a description...



```
1 create table if not exists Managed_SmartCar_Status_Info (  
2   car_number string,  
3   sex string,  
4   age string,  
5   marriage string,  
6   region string,  
7   job string,  
8   car_capacity string,  
9   car_year string,  
10  car_model string,  
11  tire_fl string,  
12  tire_fr string,  
13  tire_bl string,  
14  tire_br string,  
15  light_fl string,  
16  light_fr string,  
17  light_bl string,  
18  light_br string,  
19  engine string,  
20  break string,  
21  battery string,  
22  reg_date string  
23 )  
24 partitioned by( biz_date string )  
25 row format delimited  
26 fields terminated by ','  
27 stored as textfile;
```

그림 6.69 주제 영역 1에 사용할 하이브 테이블 쿼리 작성

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

05. 계속해서 내 문서 /workflow/hive\_script/subject1의 위치에 두 번째 하이브 스크립트 파일을 만들어 본다.  
subject1 디렉터리에서 [새 문서] → [Hive 쿼리]를 선택한다.



## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

06. 하이브 에디트 창에 External의 SmartCar\_Status\_Info 테이블에 작업 일자(오늘 날짜)를 기준으로 파티션 정보를 추가하는 스크립트를 작성한 후 저장한다. 파일 이름 입력창이 나타나면 이름을 "alter\_partition\_smartcar\_status\_info.hql"로 입력하고 [Save] 버튼을 클릭한다.

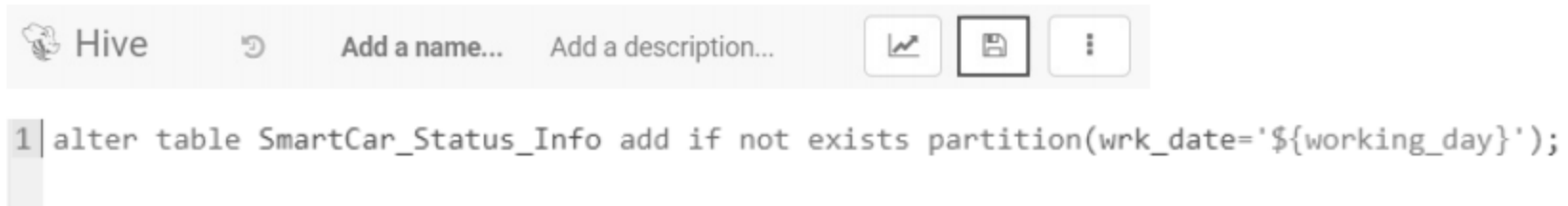


그림 6.70 주제 영역 1에 사용할 파티션 생성

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

07. 내 문서의 /workflow/hive\_script/subject1 위치에 세 번째 하이브 스크립트 파일을 만든다. [새 문서] → [Hive 쿼리]를 선택한다.

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

08. 하이브 에디트 창이 활성화되면 먼저 동적 파티션을 생성하기 위한 하이브 환경변수 값을 설정해야 한다. 동적 파티션은 지정된 특정 필드값을 기준으로 자동 파티션되는 기능인데, 다음의 옵션을 반드시 지정해야 한다.

- `set hive.exec.dynamic.partition = true;`
- `set hive.exec.dynamic.partition.mode = nonstrict;`

동적 파티션 설정에 이어서 External 영역에 생성돼 있는 두 테이블인 SmartCar\_Master\_Over18과 SmartCar\_Status\_Info를 조인해서 조회된 데이터를 앞서 만든 Managed 테이블인 Managed\_SmartCar\_Status\_Info에 삽입하는 하이브 스크립트를 작성하고 [저장] 버튼을 클릭한다. 파일명은 “insert\_table\_managed\_smartcar\_status\_info.hql”로 지정한다.



Hive



Add a name...

Add a description...



```
1 set hive.exec.dynamic.partition=true;
2 set hive.exec.dynamic.partition.mode=nonstrict;
3
4 insert overwrite table Managed_SmartCar_Status_Info partition(biz_date)
5 select
6   t1.car_number,
7   t1.sex,
8   t1.age,
9   t1.marriage,
10  t1.region,
11  t1.job,
12  t1.car_capacity,
13  t1.car_year,
14  t1.car_model,
15  t2.tire_fl,
16  t2.tire_fr,
17  t2.tire_bl,
18  t2.tire_br,
19  t2.light_fl,
20  t2.light_fr,
21  t2.light_bl,
22  t2.light_br,
23  t2.engine,
24  t2.break,
25  t2.battery,
26  t2.reg_date,
27  substring(t2.reg_date, 0, 8) as biz_date
28 from SmartCar_Master_Over18 t1 join SmartCar_Status_Info t2
29 on t1.car_number = t2.car_number and t2.wrk_date = '${working_day}';
```

그림 6.71 주제 영역 1의 Managed 테이블에 데이터를 생성하는 쿼리 작성

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

09. 상단의 쿼리 콤보박스에서 [스케줄러] → [Workflow]를 선택해서 실행할 워크플로를 만든다(좌측 드롭박스 메뉴의 [스케줄러]를 선택해도 된다).

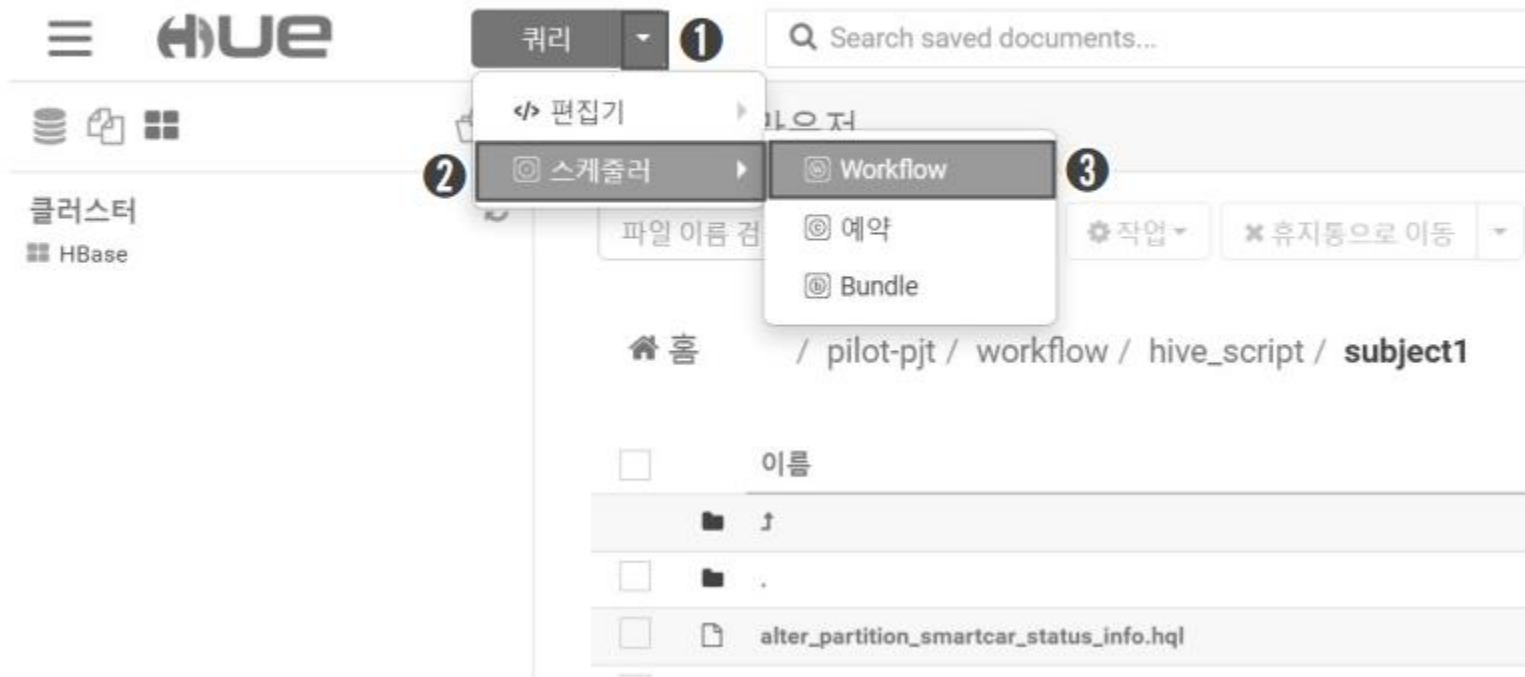
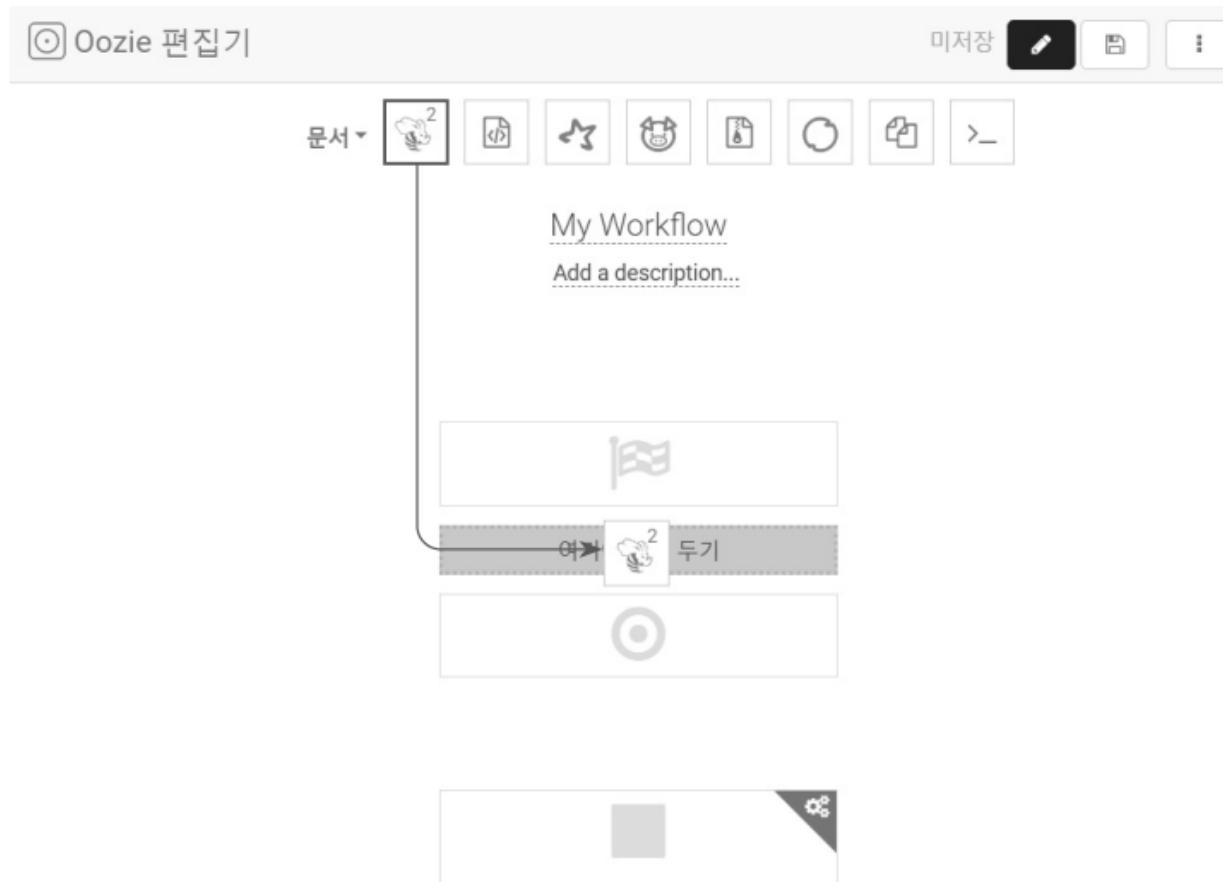


그림 6.72 휴의 워크플로 생성

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

10. 첫 번째 작업으로 주제 영역 1의 데이터를 관리하기 위해 Managed 영역에 하이브 테이블을 만드는 작업을 추가한다. 워크플로의 작업 툴 박스에서 "Hive 쿼리(HiveServer2 스크립트)" 작업을 워크플로의 첫 번째 작업 노드에 드래그 앤드 드롭한다.



## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

11. 사용할 Hive 쿼리 스크립트 파일을 선택한다. 앞 단계에서 만든 내 문서의 /workflow/hive\_script/subject1에 만들어 놓은 create\_table\_managed\_smartcar\_status\_info.hql을 선택한 후 [추가] 버튼을 누른다.

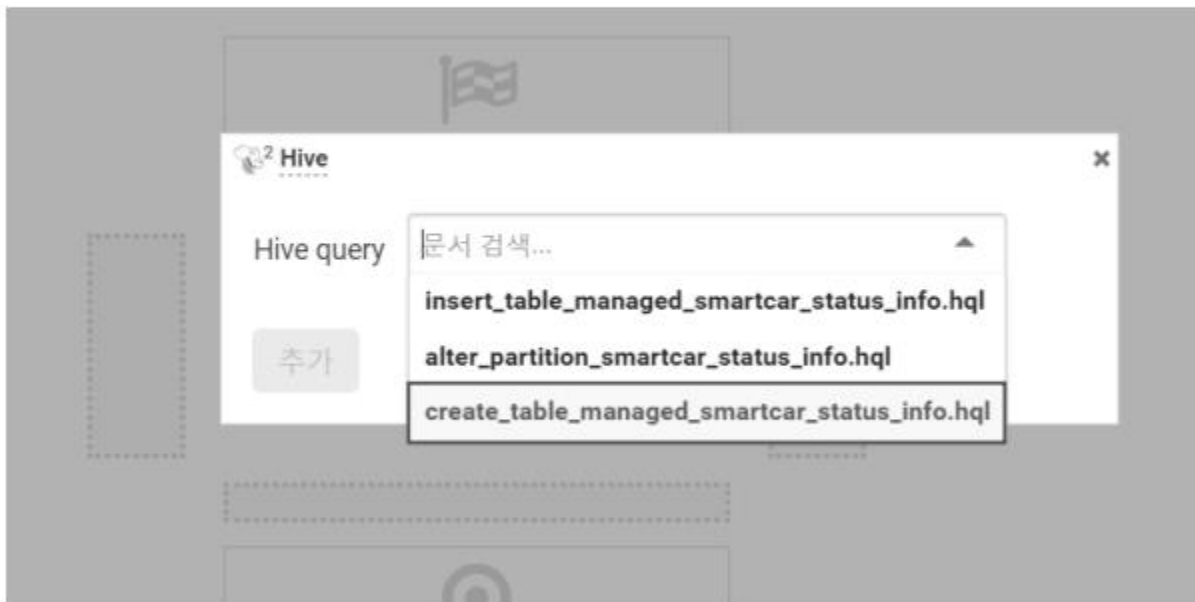


그림 6.74 Hive 쿼리 작업 파일 선택

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

12. 두 번째 작업으로 데이터를 가져올 External 영역의 SmartCar\_Status\_Info 테이블에 오늘 날짜로 파티션 정보를 설정하는 하이브 작업을 만든다. 워크플로의 작업 툴박스에서 “Hive 쿼리” 작업을 워크플로의 두 번째 작업 노드에 드래그 앤드 드롭한다.





## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

13. 사용할 Hive 쿼리 스크립트 파일을 선택한다. 앞 단계에서 만든 내 문서의 /workflow/hive\_script/subject1에 만들어 놓은 alter\_partition\_smartcar\_status\_info.hql을 선택한 후, [추가] 버튼을 클릭한다.

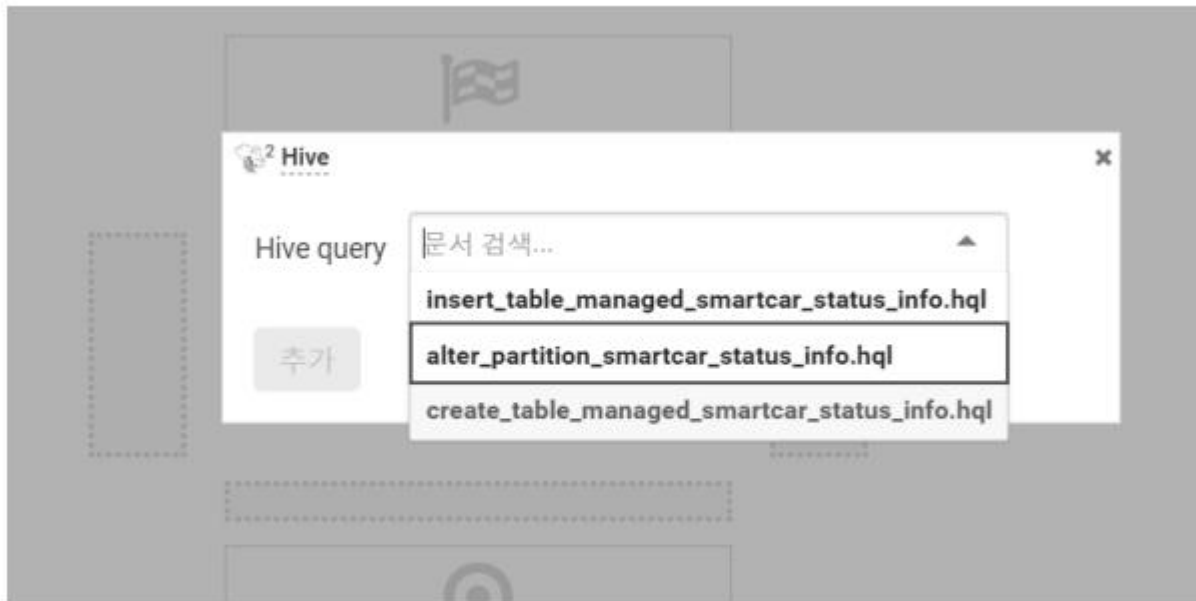


그림 6.76 Hive 쿼리 작업 파일 선택

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

14. 그림 6.71의 하이브 스크립트 안에서 정의한 작업일자(수집일자) 매개변수인 `${working_day}`의 값을 그림 6.77의 워크플로 매개변수와 연결한다. 이때 `working_day` 매개변수에 `${today}`를 설정하는데, 이 값은 잠시 후 우지의 예약 스케줄러에서 정의해 등록한다.

▪ `working_day=${today}`

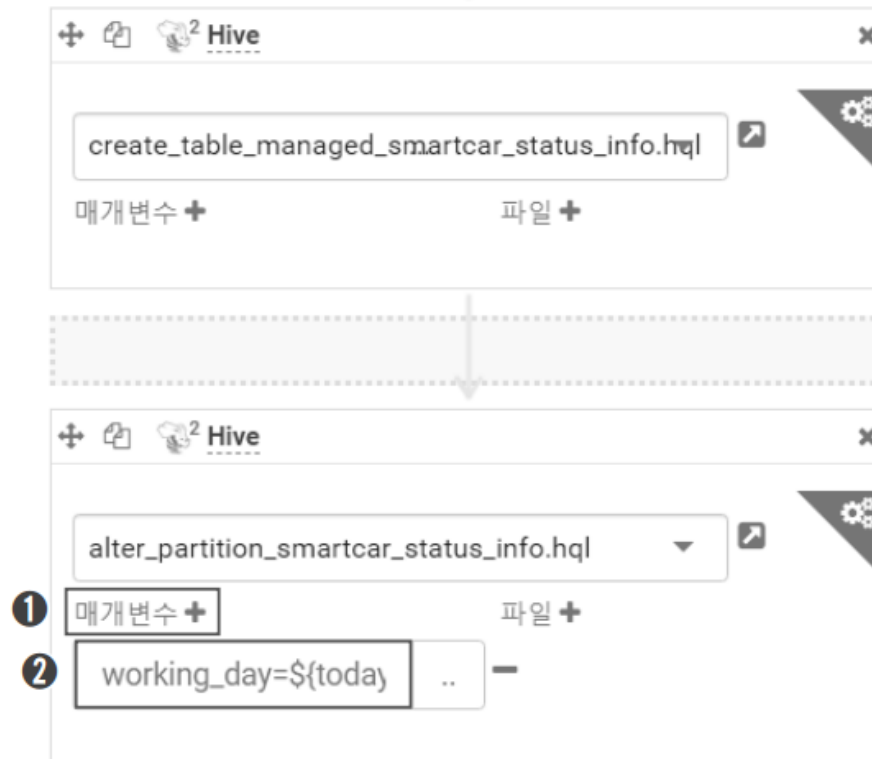


그림 6.77 Hive 쿼리 작업의 매개변수 설정

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

15. 마지막 작업으로 첫 번째 작업에서 만든 Managed\_SmartCar\_Status\_Info 테이블에 데이터를 저장하기 위한 하 이브 작업을 만든다. 워크플로의 작업 툴박스에서 “Hive 쿼리” 작업을 워크플로의 네 번째 작업 노드에 드래그 앤 드롭한다.



## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

16. 사용할 Hive 쿼리 스크립트 파일을 선택한다. 앞 단계에서 만든 내 문서의 /workflow/hive\_script/subject1에 있는 insert\_table\_managed\_smartcar\_status\_info.hql을 선택한 후, [추가] 버튼을 클릭한다.

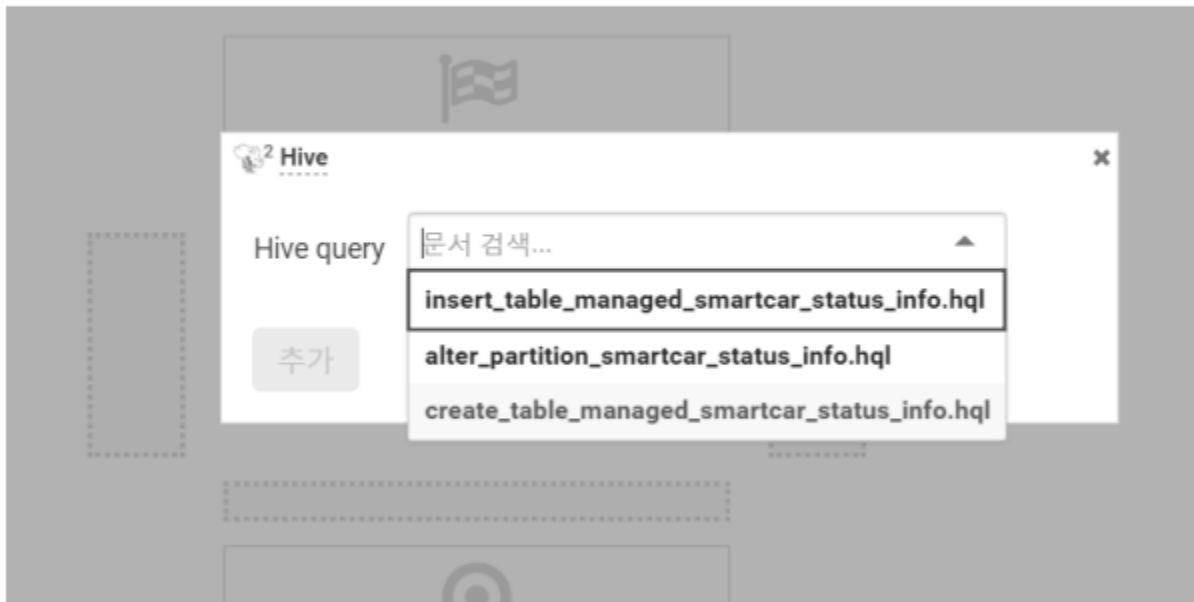


그림 6.79 Hive 쿼리 작업 파일 선택

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

17. 이번에도 하이브 스크립트 안에서 정의한 작업일자(수집일자) 매개변수인 `${working_day}`의 값을 워크플로의 매개변수와 연결한다. `working_day` 매개변수에 `${today}`를 설정하는데 이 값은 잠시 후 우지의 예약 작업의 Scheduler 설정 단계에서 정의한다.

- `working_day = ${today}`



그림 6.80 Hive 쿼리 작업의 매개변수 설정

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

18. 워크플로의 이름을 지정한다. 워크플로 상단의 “My Workflow”를 클릭하고, “Subject 1 - Workflow”로 변경한 후 [확인] 버튼을 클릭한다.

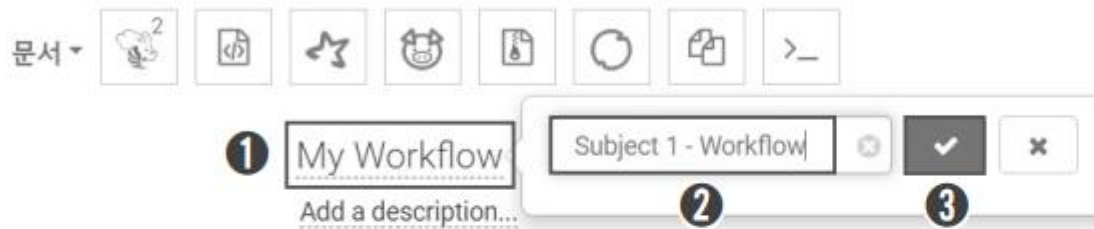


그림 6.81 주제 영역 1의 워크플로 이름 설정

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

19. 워크플로 작성을 완료한다. 우측 상단의 [저장] 버튼을 누른다.

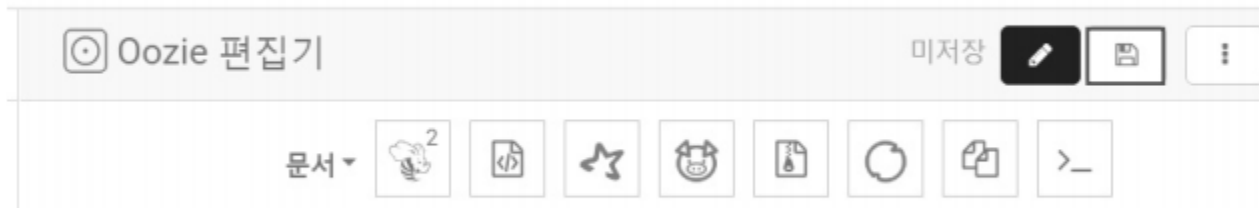


그림 6.82 주제 영역 1 워크플로 저장 및 완료

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

20. 이제 작성한 워크플로를 작동하기 위한 예약 작업을 생성한다. 그림 6.83처럼 상단의 쿼리 콤보박스에서 [스케줄러] → [예약]을 선택한다.



그림 6.83 휴의 예약 작업 생성



# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

21. 예약 작업의 이름을 "Subject 1 - 예약"으로 입력한다.

The screenshot shows the 'Oozie 편집기' (Oozie Editor) interface. At the top, there is a header 'Oozie 편집기'. Below it, a job configuration form is visible. The first field, labeled '1', is 'My Schedule'. To its right, a dropdown menu is open, showing 'Subject 1 - 예약' as the selected option, which is labeled '2'. The dropdown menu also contains a checkmark button labeled '3' and a close button 'x'. Below the job name field, there is a text input field for 'Add a description...'. Further down, the text '예정된 Workflow는 무엇입니까?' (Which workflow is scheduled?) is displayed, followed by 'Subject 1 - Workflow' with a small icon.

그림 6.84 주제 영역 1의 예약 이름 지정

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

22. 예약 작업에서 사용할 워크플로를 선택한다. 앞서 만든 주제 영역 1의 워크플로인 “Subject 1 – Workflow”를 선택한다.

Oozie 편집기

**Subject 1 - 예약**

Add a description...

Workflow 선택

Workflow 필터링

Subject 1 - Workflow

2

예정된 Workflow는 무엇입니까?

Workflow 선택... 1

그림 6.85 주제 영역 1의 예약 작업에서 작동시킬 워크플로 선택

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

23. 예약 작업의 워크플로를 주기적으로 실행하기 위한 일종의 배치 잡 스케줄러다. 그림 6.86처럼 스케줄러 실행 주기와 옵션 값을 설정한다. 시작일자와 종료일자는 파일럿 프로젝트 실습일자에 맞춰 입력한다.

간격은 얼마입니까?

다음 마다 일 다음에서 1 : 0

분 숨기기

☐ 고급 구문

시간대 Asia/Seoul

원본

->

- 실행 간격: 매일, 01시
- 시작 일자: 2020년 03월 23일, 00시 00분
- 종료 일자: 2020년 12월 31일, 23시 59분
- 시간대: Asia/Seoul

그림 6.86 주제 영역 1 예약 작업의 작동 스케줄링 설정

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

24. 워크플로에서 사용할 매개변수인 today 값을 설정한다.

매개 변수

working\_day

매개 변수

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

+ 매개 변수 추가

그림 6.88 주제 영역 1 예약 작업의 수행일자 매개변수 설정

앞서 워크플로의 하이브 작업에서 작업일자(수집일자) 매개변수를 “working\_day=\${today}”로 등록했다. today의 값을 예약 작업의 내장 함수를 통해 다음과 같이 설정한다.

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

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

25. 우지의 예약 작업 설정이 모두 끝났다. [저장] 버튼을 누르고 작성을 완료한다.

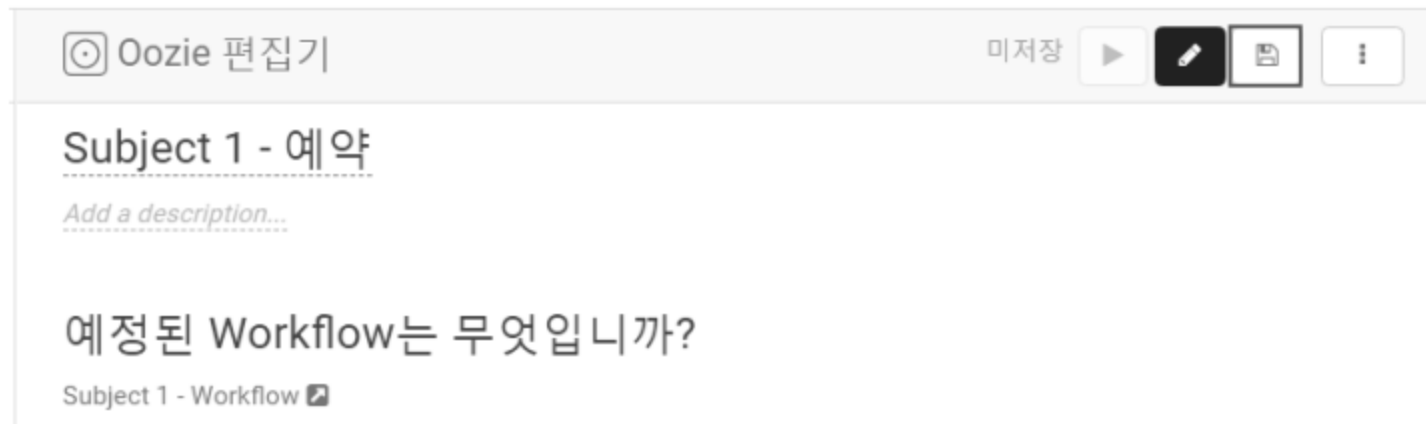


그림 6.89 주제 영역 1의 예약 기능 저장 및 완료

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

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

Oozie 편집기

Subject 1 - 예약

Add a description...

예정된 Workflow는

Subject 1 - Workflow

Subject 1 - 예약을(를) 제출하시겠습니까?

start\_date

2020-03-23T00:00

end\_date

2020-12-31T23:59

☐ Job을 제출하기 전에 모의 테스트 진행

취소 제출

그림 6.90 주제 영역 1의 예약 기능 제출 및 실행

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

27. 제출된 예약 작업 상태를 확인해 본다. 우측 상단의 [Job] 버튼을 클릭하고 잡 브라우저에서 [일정]을 선택한다. 앞서 등록한 “Subject 1 - 예약”이 설정한 스케줄러 시간에 따라 “PREP(준비)”, “Running(실행)” 상태 등으로 표기되며, 매일 새벽 01시가 되면 등록된 워크플로가 작동하게 된다.

Job Browser | Job | Workflow | **일정** | Bundle | SLAs

user:admin | ☐ 성공 ☐ 실행 중 ☐ 실패함 | ▶ 다시 시작 || 중단 ✕ 중지

실행 중

이름	사용자	유형	상태	진행률	그룹	수정됨	소요 시간	ID
<input type="checkbox"/> Subject 1 - 예약	admin	schedule	RUNNING	4%		2020년 4월 4일 오 전 10시 0분	283d, 23h, 59m, 0s	0000000-200322142437022- oozie-oozi-C

그림 6.91 예약 기능 모니터링 1

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

28. 잡 브라우저에서는 등록된 잡의 현황, 진행 상태, 처리 이력과 결과 등을 확인할 수 있다.

Job Browser

Job Workflow 일정 Bundle SLAs

user:admin ☒ 성공 ☐ 실행 중 ☐ 실패함

▶ 다시 시작 ||| 중단 ✕ 중지

실행 중

<input type="checkbox"/> 이름	사용자	유형	상태	진행률	그룹	수정됨	소요 시간	ID
<input type="checkbox"/> Subject 1 - 예약	admin	schedule	RUNNING	4%		2020년 4월 4일 오후 10시 0분	283d, 23h, 59m, 0s	0000000-200325213559078-oozie-oozi-C

Completed

<input type="checkbox"/> 이름	사용자	유형	상태	진행률	그룹	수정됨	소요 시간	ID
<input type="checkbox"/> Subject 1 - 예약	admin	schedule	KILLED	4%		2020년 4월 4일 오후 10시 0분	283d, 23h, 59m, 0s	0000004-200325213559078-oozie-oozi-C
<input type="checkbox"/> Subject 1 - 예약	admin	schedule	KILLED	4%		2020년 4월 4일 오후 10시 0분	283d, 23h, 59m, 0s	0000001-200323023611310-oozie-oozi-C

그림 6.92 잡 브라우저 모니터링



# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

특정 워크플로가 실행 중일 때는 그림 6.93처럼 작업 진행 상태를 프로그레스바로 곧바로 확인할 수 있다.

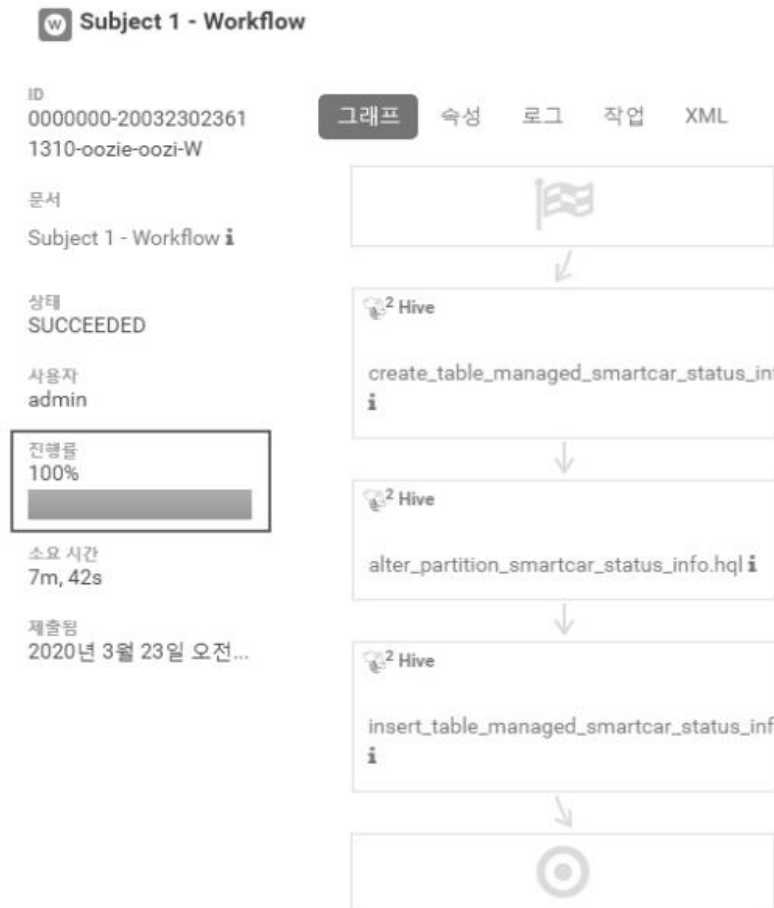


그림 6.93 실행 중인 워크플로 모니터링

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

예약된 작업들이 활성화되어 실행될 때는 우측 상단에 [Job Browser]의 버튼 우측에 실행 중인 잡의 개수가 표시된다. [작업 미리보기]를 클릭하면 실행 중이거나 종료된 잡의 상세 정보를 확인할 수 있다.

The screenshot shows the Job Browser interface. At the top, there's a 'Job' tab and a 'Workflow' tab. Below the tabs, there's a search bar with 'user:admin' and a filter section with checkboxes for '성공' (Success), '실행 중' (Running), and '실패함' (Failed). The main area displays a list of jobs:

Job Name	Status	Duration	User	Application ID
몇 초 전 - ACCEPTED	성공	3.70s	admin	application_1585139700601_0006
Oozie Launcher	실행 중	7.84s	admin	application_1585139700601_0005
16분 전 - KILLED	실패함	27.98s	admin	application_1585139700601_0004
Oozie Launcher	실행 중	19.3s	admin	application_1585139700601_0003
17분 전 - SUCCEEDED	성공			
Oozie Launcher	실행 중			
20분 전 - SUCCEEDED	성공			
Oozie Launcher	실행 중			

그림 6.94 작업 미리 보기

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

29. “Subject 1 – Workflow”가 정상적으로 작동했으면, 휴의 Hive Editor로 이동해서 그림 6.95와 같이 하이브 QL을 작성해서 실행한다. “biz\_date=20200322”의 날짜는 독자들의 파일럿 환경의 실행 일자에 맞춰 입력해야 한다.

참고로 워크플로를 통해 만들어지는 managed\_smartcar\_status\_info 같은 테이블이 하이브 편집기에서 곧바로 표기되지 않아 하이브 편집기에서 해당 테이블 관련 쿼리를 실행할 때 에러가 발생할 수 있다. 이때는 하이브 테이블 목록 상단의 [새로고침] 버튼을 클릭해 테이블 목록을 새롭게 갱신한다.

```
1 | select * from managed_smartcar_status_info
2 | where biz_date = '20200322' limit 10
```



그림 6.95 주제 영역 1 워크플로의 실행 결과 확인 1

# 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

## 스마트카 상태 정보 모니터링

실행 결과는 다음과 같다. 가장 우측 열의 biz\_date 칼럼에 날짜가 "20200322"인 것을 확인할 수 있다.

쿼리 기록	저장된 쿼리	결과 (10)	
	tatus_info.battery	managed_smartcar_status_info.reg_date	managed_smartcar_statu
1		20200322000000	20200322
2		20200322000004	20200322
3		20200322000008	20200322
4		20200322000012	20200322
5		20200322000016	20200322

그림 6.96 주제 영역 1 워크플로의 실행 결과 확인 2

## 6.6 탐색 파일럿 실행 4단계 - 주제영역 1

### 스마트카 상태 정보 모니터링

30. 한번 등록한 예약 작업은 종료 기간까지 계속 활성화된 상태로 남아 있게 되어 파일럿 환경의 리소스를 차지하게 된다. 파일럿 환경에서는 매일 워크플로가 실행될 필요가 없으니 실행 및 테스트가 끝난 작업은 휴 좌측 드롭박스 메뉴의 [Job] → [일정]을 선택해 실행 중인 예약 작업들을 모두 중지시킨다.

## Tip \_ 워크플로 즉시 실행하기

그림 6.86의 설정대로라면 익일 새벽 1시가 돼서야 워크플로의 작동 여부를 확인할 수 있다. 테스트를 위해 워크플로를 즉시 실행할 수 있는 좀 더 쉬운 방법이 있는데, [내 문서]에서 실행할 워크플로를 선택하고 들어가 상단의 [수정] 버튼을 클릭하고 매개변수의 작업 일자(매개변수: working\_day=YYYYMMDD)를 그림 6.87처럼 독자의 파일럿 상황에 맞춰 직접 입력한 후 상단의 [저장] → [제출] 버튼을 차례로 누르면 워크플로가 즉시 실행된다. 매개변수 설정이 필요없는 워크플로는 곧바로 [저장] → [제출]을 선택하면 된다.



그림 6.87 예약 기능 즉시 실행하기 설정 2

## Tip \_ 워크플로 즉시 실행하기

그림 6.86의 설정대로라면 익일 새벽 1시가 돼서야 워크플로의 작동 여부를 확인할 수 있다. 테스트를 위해 워크플로를 즉시 실행할 수 있는 좀 더 쉬운 방법이 있는데, [내 문서]에서 실행할 워크플로를 선택하고 들어가 상단의 [수정] 버튼을 클릭하고 매개변수의 작업 일자(매개변수: working\_day=YYYYMMDD)를 그림 6.87처럼 독자의 파일럿 상황에 맞춰 직접 입력한 후 상단의 [저장] → [제출] 버튼을 차례로 누르면 워크플로가 즉시 실행된다. 매개변수 설정이 필요없는 워크플로는 곧바로 [저장] → [제출]을 선택하면 된다.

### [주제 영역별 매개변수 설정]

- \* 주제영역1: 플럼이 하둡에 적재한 날짜 (개인별 실행중인 파일럿의 현재 작업일자 값으로, 모를경우 하둡 적재경로인 `/pilot-pjt/collect/car-batch-log/wrk_date=YYYYMMDD`에 디렉토리의 날짜값을 이용하면 된다. Server02에 접속해 아래의 명령으로 확인해 볼 수 있다.)

```
$ hdfs dfs -ls -R /pilot-pjt/collect/car-batch-log/
```

- \* 주제영역2: 스마트카 시뮬레이션 날짜(로그시뮬레이터 실행시 설정한 날짜로, 책기준 "20200322" 입력)
- \* 주제영역3: 스마트카 시뮬레이션 날짜(로그시뮬레이터 실행시 설정한 날짜로, 책기준 "20200322" 입력)
- \* 주제영역4: 스마트카 시뮬레이션 날짜(로그시뮬레이터 실행시 설정한 날짜로, 책기준 "20200322" 입력)
- \* 주제영역5: 미사용