



20220428

파일럿 탐색 실행 4단계 - 데이터 탐색 기능 구현 테스트

- 주제 영역 1 : 스마트카 상태 모니터 정보
- 주제 영역 2 : 스마트카 운전자 운행기록 정보
- 주제 영역 3 : 이상 운전 패턴 스마트카 정보
- 주제 영역 4 : 긴급 점검이 필요한 스마트카 정보
- 주제 영역 5 : 운전자의 차량용품 구매 이력 정보

	주제 영역 1	주제 영역 2	주제 영역 3	주제 영역 4	주제 영역 5
주제 영역	스마트카 상태 모니터 정보	스마트카 운전자 운행 기록 정보	이상 운전 패턴 스마트카 정보	긴급 점검이 필요한 스마트카 정보	운전자의 차량용품 구매 이력 정보
이용할 테이블	스마트카 마스터 데이터(SmartCar_Master_Over18)	스마트카 마스터 데이터(SmartCar_Master_Over18)	스마트카 운전자 운행기록 정보(Managed_SmartCar_Drive_Info)	스마트카 상태 모니터링 정보(Managed_SmartCar_Status_Info)	스마트카 마스터 데이터(SmartCar_Master_Over18)
	스마트카 상태 정보 데이터(SmartCar_Status_Info)	스마트카 운전자 운행 기록 데이터(SmartCar_Drive_Info_2)			스마트카 차량용품 구매 이력 데이터(SmartCar_Item_BuyList)
워크플로 이름	Subject 1 – Workflow	Subject 2 – Workflow	Subject 3 – Workflow	Subject 4 – Workflow	Subject 5 – Workflow
스케줄러 이름	Subject 1 – Coordinator	Subject 2 – Coordinator	Subject 3 – Coordinator	Subject 4 – Coordinator	Subject 5 – Coordinator
수행 주기	01:00 / 1Day	02:00 / 1Day	03:00 / 1Day	04:00 / 1Day	05:00 / 1Day
생성할 Mart 테이블	Managed_SmartCar_Status_Info	Managed_SmartCar_Drive_Info	Managed_SmartCar_Symptom_Info	Managed_SmartCar_Emergency_Check_Info	Managed_SmartCar_Item_BuyList_Info

스마트카 상태 정보 데이터 생성

- Server02에 SSH를 통해 접속하고 명령어 실행(root로 접속)
 - HDFS 시작
 - `cd /home/pilot-pjt/working`
 - `java -cp bigdata.smartcar.loggen-1.0.jar com.wikibook.bigdata.smartcar.loggen.CarLogMain 20220428 100 &` 실행
 - `cd /home/pilot-pjt/working/SmartCar/` 로 경로 이동
 - `ls -ltrh SmartCarStatusInfo_20220428.txt` 로 100MB 파일이 생성된 것을 확인
 - `ps -ef | grep smartcar.log`를 이용하고
 - `kill -9 [pid]` 로 삭제 (pid : 프로세스 번호)
 - `mv /home/pilot-pjt/working/SmartCarStatusInfo_20220428.txt /home/pilot-pjt/working/car-batch-log/`로 데이터 수집
 - `hdfs dfs -ls -R /pilot-pjt/collect/car-batch-log/` 로 정상적으로 적재됐는지 확인.

스마트카 운전자 운행 로그 생성

- 플럼, 카프카, 스톰, 레디스, HBase가 정지돼 있다면 모두 시작
 - `cd /home/pilot-pjt/working` 으로 경로 이동
 - `java -cp bigdata.smartcar.loggen-1.0.jar com.wikibook.bigdata.smartcar.loggen.DriverLogMain 20220428 5 &` 로 데이터 생성
 - `cd /home/pilot-pjt/working/driver-realtime-log`로 경로이동
 - `tail -f SmartCarDriverInfo.log`를 통하여 로그가 계속 기록되는지 확인
 - Hue 웹 UI 에서 메뉴 → HBase → DriverCarInfo 테이블을 선택해서 20220428의 운행 데이터가 생성됐는지 확인
 - 레디스 CLI를 이용하여 과속한 스마트카 차량 정보를 확인
 - `redis-cli` 실행

- smembers 20220428 명령어 입력
- ps -ef | grep smartcar.log 를 이용하여 pid 확인
- kill -9 [pid] 로 시뮬레이터 종료
- 이후 플럼, 카프카, 스톰, 레디스 종료

주제 영역 1. 스마트카 상태 정보 모니터링 - 워크플로 작성

- Hue 웹 UI 에서 좌측 드롭박스 메뉴에서 [문서] 메뉴 선택
- workflow라는 새 폴더 디렉토리 생성
- workflow 폴더 안에 hive_script 새 디렉토리 생성
- hive_script 폴더 안에 subject1 ,2, 3, 4, 5 폴더를 각각 생성
- 세부 Hive 쿼리 생성
 - subject1 폴더안에 새로운 Hive 쿼리 생성

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/f0038762-2efa-44bc-a93f-c32928b04974/그림-6.69.hql>

- “create_table_managed_smartcar_status_info.hql”로 저장

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/45bebca0-73c5-4371-a248-47440e0e6acc/그림-6.70.hql>

- “alter_partition_smartcar_status_info.hql” 로 저장
- 동적 파티션을 생성하기 위한 하이브 환경변수 값 설정
 - set hive.exec.dynamic.partition = true;
 - set hive.exec.dynamic.partition.mode = nonstrict;

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/ae53df28-2571-4448-a67f-4887ddb3429/그림-6.71.hql>

“insert_table_managed_smartcar_status_info.hql”로 저장

- 상단 쿼리 콤보박스에서 [스케줄러] → [workflow] 선택
- 워크플로의 작업 툴 박스에서 “Hive 쿼리” 작업을 워크플로의 첫 번째 작업노드에 드래그 앤드 드롭한다..
- create, alter, insert 순으로 넣기
- alter, insert에서 매개변수로 working_day=\${today} 설정
 - 워크플로 즉시 실행하고 싶다면 working_day=현재 날짜로 입력
- 워크플로 상단의 “My Workflow”를 클릭후 “Subject 1 - Workflow”로 변경 후 저장
- [스케줄러] → [예약] 으로 이동
- “My Schedule”을 “Subject 1 - 예약” 으로 입력
- Workflow 선택에서 방금 만든 워크플로 선택
- 시간대를 Asia/Seoul로 맞추고 시간 설정 00:00 ~ 23:59 , 14:0지정한 후 저장
- \${coord:formatTime(coord:dateTzoffset(coord:nominalTime(), "Asia/Seoul"), 'yyyyMMdd')}

주제 영역 2. 스마트카 운전자 운행 기록 정보 - 워크플로 작성

서브 Hive 쿼리 생성

- 휴 내 문서 : /workflow/hive_script/subject2에 새 문서로 Hive쿼리 생성

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/c36ae509-16f3-4b76-813d-5b6aff1d5fd3/그림-6.97.hql>

- “create_table_smartcar_drive_info_2.hql” 로 저장

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/2bc9b2a9-fd7a-42ea-a213-f15536ca3ea2/그림-6.98.hql>

- “insert_table_smartcar_drive_info_2.hql” 로 저장

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/9f10a40b-9816-473e-bed5-87c3a80a9ef9/그림-6.99.hql>

- “create_table_managed_smartcar_dirve_info.hql” 로 저장

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e71eb5d6-0faf-4ef2-a70b-c3afd0c443d1/그림-6.100.hql>

- “insert_table_managed_smartcar_drive_info.hql”로 저장
- 주제 영역 1과 같이 workflow 설정과 예약을 진행

주제 영역 3. 이상 운전 패턴 스마트카 정보 - 워크플로 작성

서브 Hive 쿼리 생성

- 휴 내 문서 : /workflow/hive_script/subject3에 새 문서로 Hive쿼리 생성

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/f1aa2f6a-6422-4c05-b94e-e1fbfd992f44/그림-6.102.hql>

- “create_table_managed_smartcar_symptom_info.hql” 로 저장

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/cd501324-1fa4-430d-91d7-6cf38e2f637e/그림-6.103.hql>

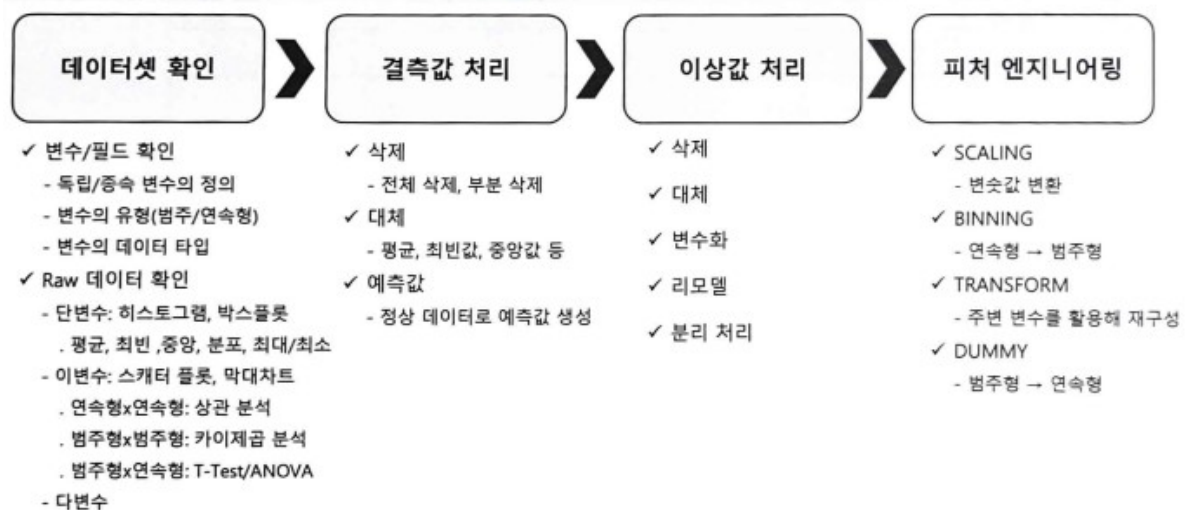
- “insert_table_managed_smartcar_symptom_info.hql”로 저장
- 주제 영역 1과 같이 workflow 설정과 예약을 진행

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/d978b270-648e-46ee-8b03-644a10c958f8/그림-6.105.hql>

- sql 쿼리로 실행 결과 확인

빅데이터 분석을 위한 탐색 및 전처리 작업

- 탐색 단계에서는 데이터의 전처리 작업의 비중이 매우 높는데 일련의 과정들을 크게 4 단계로 나눌 수 있다.



주제 영역 4. 긴급 점검이 필요한 스마트카 정보 - 워크플로 작성

서브 Hive 쿼리 생성

- 휴 내 문서 : /workflow/hive_script/subject3에 새 문서로 Hive쿼리 생성

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/6fdeea1f-e640-4e05-87d7-c535f0977840/그림-6.110.hql>

- “create_table_managed_smartcar_emergency_check_info.hql” 로 저장

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/dc715612-e792-4ebf-947e-aca7913c95f7/그림-6.111.hql>

- “insert_table_managed_smartcar_emergency_check_info.hql”로 저장
- 주제 영역 1 과 같이 workflow와 예약을 진행
- select * from managed_smartcar_emergency_check_info where biz_date = '20220428' 로 확인