**일일교육내용**

**일자 : 2019.06.25**

|  |  |
| --- | --- |
| **교육과목** | DB |
| **교육내용** | 다차원 모델 실습 |
| **실습 내용** | **1. 분기별,월별,요일별, 시간대별 에러율을 확인하는 sql을 작성하여 에러율을 확인합니다.**  **분기별 에러율**  SELECT CQUATER, AVG(ERATE) FROM ERR\_RATE, TERM  WHERE ERR\_RATE.CDATE=TERM.CDATE GROUP BY CQUATER;    **월별 에러율**  SELECT CMM, AVG(ERATE) FROM ERR\_RATE, TERM  WHERE ERR\_RATE.CDATE=TERM.CDATE GROUP BY CMM ORDER BY CMM;    **요일별 에러율**  SELECT CDAY, AVG(ERATE) FROM ERR\_RATE, TERM  WHERE ERR\_RATE.CDATE=TERM.CDATE GROUP BY CDAY;    **시간대별 에러율**  SELECT CTIME, AVG(ERATE) FROM ERR\_RATE, TERM  WHERE ERR\_RATE.CDATE=TERM.CDATE  GROUP BY CTIME ORDER BY LENGTH(CTIME), CTIME;    **2. 라인이 추가된 다차원 모델에서 불량률을 2차원의 관점에서 분석할 수 있는 경우를 모두 나열하고, 그중에 10가지 정도만 SQL을 작성하여 결과 확인**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | 연도별 | 제조라인별 |  | 연도별 | 팀별 | | 반기별 | 제조라인별 |  | 반기별 | 팀별 | | 분기별 | 제조라인별 |  | 분기별 | 팀별 | | 월별 | 제조라인별 |  | 월별 | 팀별 | | 날짜별 | 제조라인별 |  | 날짜별 | 팀별 | | 요일별 | 제조라인별 |  | 요일별 | 팀별 | | 시간별 | 제조라인별 |  | 시간별 | 팀별 |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | 제조라인별 | 연도별 |  | 팀별 | 연도별 | | 제조라인별 | 반기별 |  | 팀별 | 반기별 | | 제조라인별 | 분기별 |  | 팀별 | 분기별 | | 제조라인별 | 월별 |  | 팀별 | 월별 | | 제조라인별 | 날짜별 |  | 팀별 | 날짜별 | | 제조라인별 | 요일별 |  | 팀별 | 요일별 | | 제조라인별 | 시간별 |  | 팀별 | 시간별 |   **1. 연도별, 제조라인별**  SELECT CYYYY, C.LNO, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY CYYYY, C.LNO ORDER BY CYYYY;  2018 L01 2.65376344086021505376344086021505376344  2018 L02 2.7  2018 L03 2.38494623655913978494623655913978494624  2018 L04 2.35752688172043010752688172043010752688  2018 L05 2.53655913978494623655913978494623655914  2019 L01 2.53657142857142857142857142857142857143  2019 L02 2.52457142857142857142857142857142857143  2019 L03 2.57314285714285714285714285714285714286  2019 L04 2.73085714285714285714285714285714285714  2019 L05 2.38514285714285714285714285714285714286  **2. 제조라인별 연도별**  SELECT C.LNO, CYYYY, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY C.LNO, CYYYY ORDER BY C.LNO;  L01 2018 2.65376344086021505376344086021505376344  L01 2019 2.53657142857142857142857142857142857143  L02 2018 2.7  L02 2019 2.52457142857142857142857142857142857143  L03 2018 2.38494623655913978494623655913978494624  L03 2019 2.57314285714285714285714285714285714286  L04 2018 2.35752688172043010752688172043010752688  L04 2019 2.73085714285714285714285714285714285714  L05 2018 2.53655913978494623655913978494623655914  L05 2019 2.38514285714285714285714285714285714286  **3. 연도별 팀별**  SELECT CYYYY, WTEAM, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY CYYYY, WTEAM ORDER BY CYYYY;  2018 T01 2.67688172043010752688172043010752688172  2018 T02 2.46075268817204301075268817204301075269  2018 T03 2.35752688172043010752688172043010752688  2019 T01 2.53057142857142857142857142857142857143  2019 T02 2.47914285714285714285714285714285714286  2019 T03 2.73085714285714285714285714285714285714  **4. 팀별 연도별**  SELECT WTEAM, CYYYY, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY WTEAM, CYYYY ORDER BY WTEAM;  T01 2018 2.67688172043010752688172043010752688172  T01 2019 2.53057142857142857142857142857142857143  T02 2018 2.46075268817204301075268817204301075269  T02 2019 2.47914285714285714285714285714285714286  T03 2018 2.35752688172043010752688172043010752688  T03 2019 2.73085714285714285714285714285714285714  **5. 분기별 제조라인별**  SELECT CQUATER, C.LNO, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY CQUATER, C.LNO ORDER BY CQUATER;  1 L01 2.49444444444444444444444444444444444444  1 L02 2.67666666666666666666666666666666666667  1 L03 2.64333333333333333333333333333333333333  1 L04 2.83888888888888888888888888888888888889  1 L05 2.35222222222222222222222222222222222222  2 L01 2.55287356321839080459770114942528735632  2 L02 2.38735632183908045977011494252873563218  2 L03 2.53103448275862068965517241379310344828  2 L04 2.58160919540229885057471264367816091954  2 L05 2.45287356321839080459770114942528735632  3 L01 2.66195652173913043478260869565217391304  3 L02 2.72826086956521739130434782608695652174  3 L03 2.52391304347826086956521739130434782609  3 L04 2.14565217391304347826086956521739130435  3 L05 2.61195652173913043478260869565217391304  4 L01 2.67391304347826086956521739130434782609  4 L02 2.65652173913043478260869565217391304348  4 L03 2.21304347826086956521739130434782608696  4 L04 2.59673913043478260869565217391304347826  4 L05 2.43260869565217391304347826086956521739  **6. 라인별 분기별**  SELECT C.LNO, CQUATER, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY C.LNO, CQUATER ORDER BY C.LNO;  L01 1 2.49444444444444444444444444444444444444  L01 2 2.55287356321839080459770114942528735632  L01 3 2.66195652173913043478260869565217391304  L01 4 2.67391304347826086956521739130434782609  L02 1 2.67666666666666666666666666666666666667  L02 2 2.38735632183908045977011494252873563218  L02 3 2.72826086956521739130434782608695652174  L02 4 2.65652173913043478260869565217391304348  L03 1 2.64333333333333333333333333333333333333  L03 2 2.53103448275862068965517241379310344828  L03 3 2.52391304347826086956521739130434782609  L03 4 2.21304347826086956521739130434782608696  L04 1 2.83888888888888888888888888888888888889  L04 2 2.58160919540229885057471264367816091954  L04 3 2.14565217391304347826086956521739130435  L04 4 2.59673913043478260869565217391304347826  L05 1 2.35222222222222222222222222222222222222  L05 2 2.45287356321839080459770114942528735632  L05 3 2.61195652173913043478260869565217391304  L05 4 2.43260869565217391304347826086956521739  **7. 팀별 분기별**  SELECT B.WTEAM, CQUATER, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY B.WTEAM, CQUATER ORDER BY B.WTEAM;  T01 1 2.58555555555555555555555555555555555556  T01 2 2.47011494252873563218390804597701149425  T01 3 2.69510869565217391304347826086956521739  T01 4 2.66521739130434782608695652173913043478  T02 1 2.49777777777777777777777777777777777778  T02 2 2.4919540229885057471264367816091954023  T02 3 2.56793478260869565217391304347826086957  T02 4 2.32282608695652173913043478260869565217  T03 1 2.83888888888888888888888888888888888889  T03 2 2.58160919540229885057471264367816091954  T03 3 2.14565217391304347826086956521739130435  T03 4 2.59673913043478260869565217391304347826  **8. 분기별 팀별**  SELECT CQUATER, B.WTEAM , AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY CQUATER, B.WTEAM ORDER BY CQUATER;  1 T01 2.58555555555555555555555555555555555556  1 T02 2.49777777777777777777777777777777777778  1 T03 2.83888888888888888888888888888888888889  2 T01 2.47011494252873563218390804597701149425  2 T02 2.4919540229885057471264367816091954023  2 T03 2.58160919540229885057471264367816091954  3 T01 2.69510869565217391304347826086956521739  3 T02 2.56793478260869565217391304347826086957  3 T03 2.14565217391304347826086956521739130435  4 T01 2.66521739130434782608695652173913043478  4 T02 2.32282608695652173913043478260869565217  4 T03 2.59673913043478260869565217391304347826  **9. 요일별 라인별**  SELECT CDAY, B.LNO , AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATE AND B.LNO=C.LNO  GROUP BY CDAY, B.LNO ORDER BY CDAY;  금요일 L01 2.94615384615384615384615384615384615385  금요일 L02 2.45576923076923076923076923076923076923  금요일 L03 2.54615384615384615384615384615384615385  금요일 L04 2.59807692307692307692307692307692307692  금요일 L05 2.42115384615384615384615384615384615385  목요일 L01 2.31568627450980392156862745098039215686  목요일 L02 2.58431372549019607843137254901960784314  목요일 L03 2.91960784313725490196078431372549019608  목요일 L04 2.31372549019607843137254901960784313725  목요일 L05 2.67647058823529411764705882352941176471  수요일 L01 2.56470588235294117647058823529411764706  수요일 L02 2.81960784313725490196078431372549019608  수요일 L03 2.81176470588235294117647058823529411765  수요일 L04 2.66666666666666666666666666666666666667  수요일 L05 2.36666666666666666666666666666666666667  월요일 L01 2.58461538461538461538461538461538461538  월요일 L02 2.68076923076923076923076923076923076923  월요일 L03 2.26730769230769230769230769230769230769  월요일 L04 2.59038461538461538461538461538461538462  월요일 L05 2.48846153846153846153846153846153846154  일요일 L01 2.50192307692307692307692307692307692308  일요일 L02 2.33076923076923076923076923076923076923  일요일 L03 2.11923076923076923076923076923076923077  일요일 L04 2.58653846153846153846153846153846153846  일요일 L05 2.34038461538461538461538461538461538462  토요일 L01 2.74423076923076923076923076923076923077  토요일 L02 2.69230769230769230769230769230769230769  토요일 L03 2.60961538461538461538461538461538461538  토요일 L04 2.36538461538461538461538461538461538462  토요일 L05 2.65  화요일 L01 2.51372549019607843137254901960784313725  화요일 L02 2.74705882352941176470588235294117647059  화요일 L03 2.06666666666666666666666666666666666667  화요일 L04 2.6490196078431372549019607843137254902  화요일 L05 2.29803921568627450980392156862745098039  **10. 라인별 요일별**  SELECT B.LNO, CDAY, AVG(ERATE) FROM TERM A, LINE B, ERR\_RATE C  WHERE C.CDATE=A.CDATEAND B.LNO=C.LNO  GROUP BY B.LNO, CDAY ORDER BY B.LNO;  L01 금요일 2.94615384615384615384615384615384615385  L01 목요일 2.31568627450980392156862745098039215686  L01 수요일 2.56470588235294117647058823529411764706  L01 월요일 2.58461538461538461538461538461538461538  L01 일요일 2.50192307692307692307692307692307692308  L01 토요일 2.74423076923076923076923076923076923077  L01 화요일 2.51372549019607843137254901960784313725  L02 금요일 2.45576923076923076923076923076923076923  L02 목요일 2.58431372549019607843137254901960784314  L02 수요일 2.81960784313725490196078431372549019608  L02 월요일 2.68076923076923076923076923076923076923  L02 일요일 2.33076923076923076923076923076923076923  L02 토요일 2.69230769230769230769230769230769230769  L02 화요일 2.74705882352941176470588235294117647059  L03 금요일 2.54615384615384615384615384615384615385  L03 목요일 2.91960784313725490196078431372549019608  L03 수요일 2.81176470588235294117647058823529411765  L03 월요일 2.26730769230769230769230769230769230769  L03 일요일 2.11923076923076923076923076923076923077  L03 토요일 2.60961538461538461538461538461538461538  L03 화요일 2.06666666666666666666666666666666666667  L04 금요일 2.59807692307692307692307692307692307692  L04 목요일 2.31372549019607843137254901960784313725  L04 수요일 2.66666666666666666666666666666666666667  L04 월요일 2.59038461538461538461538461538461538462  L04 일요일 2.58653846153846153846153846153846153846  L04 토요일 2.36538461538461538461538461538461538462  L04 화요일 2.6490196078431372549019607843137254902  L05 금요일 2.42115384615384615384615384615384615385  L05 목요일 2.67647058823529411764705882352941176471  L05 수요일 2.36666666666666666666666666666666666667  L05 월요일 2.48846153846153846153846153846153846154  L05 일요일 2.34038461538461538461538461538461538462  L05 토요일 2.65  L05 화요일 2.29803921568627450980392156862745098039  **3. 콩다방 매출자료를 이용해 DataMart를 구성하고자 하며 다음과 같은 접근이 이루어질 수 있어야 합니다.**  **1 기간 : 년도별, 분기별, 월별, 요일, 일별 매출을 Drill down 분석할 수 있어야합니다.**  **2 회원 : 지역별(우편번호 앞 두자리), 나이별(주민번호앞 두자리생년이용),**  **성별(주민번호 7번째 자리)**  **3 지점 :구역(강남/강북)별, 지점별, 직원성(sex)별, 직원별**    **4 상품품분류별(커피,빵류,음료수류),**  **상품특성별(커피: 얼음여부, 우유여부, 첨가종류별 분류 단위로 ),**  **상품별**  **5 subject area(sale\_analysis)에서 다차원 모델링을 수행합니다**    **6 모델링 결과에 맞게 샘플데이터를 생성합니다**  **시간**  INSERT INTO TIME\_D  select to\_char(saledate, 'YYYYMMDDHH24'),  to\_char(saledate, 'YYYY'), to\_char(saledate, 'Q'),  to\_char(saledate, 'MM'), to\_char(saledate, 'D'), to\_char(saledate, 'HH24')  from sale  where saledate >= to\_date('20151231','YYYYMMDD')  group by to\_char(saledate, 'YYYYMMDDHH24'),  to\_char(saledate, 'YYYY'), to\_char(saledate, 'Q'),  to\_char(saledate, 'MM'), to\_char(saledate, 'D'), to\_char(saledate, 'HH24');  **회원 : 지역별(우편번호 앞 두자리), 나이별(주민번호앞 두자리생년이용),**  **성별(주민번호 7번째 자리)**  INSERT INTO MEMB\_D (MEMBNO, AREA,AGE, SEX)  SELECT MEMBNO,  SUBSTR(ZIPNO,1,2), --- 우편번호 앞 두자리  2019- TO\_NUMBER('19' || SUBSTR(CTZNNO, 1,2)),  --- 나이별(주민번호앞 두자리생년이용)  DECODE(MOD(SUBSTR(CTZNNO,7,1),2),1,'M','F')  --- 성별(주민번호 7번째 자리)  FROM MEMB;  **상품**  INSERT INTO PROD\_D(PRODCD, PRODTY, ICE, MILK,ADDTY)  SELECT PRODCD,  SUBSTR(PRODCD,1,2),  DECODE(SUBSTR(PRODCD ,1,1), '1', SUBSTR(PRODCD,4,1),''),  DECODE(SUBSTR(PRODCD ,1,1), '1', SUBSTR(PRODCD,4,1),''),  DECODE(SUBSTR(PRODCD ,1,1), '1', SUBSTR(PRODCD,4,1),'')  FROM PROD;  **SALEHISTORY\_F 샘플 데이터 생성**  DECLARE  BEGIN  FOR I IN (SELECT PRODCD FROM PROD)  LOOP  INSERT INTO SALEHISTORY\_F(SALEDATE, MEMBNO, EMPNO, PRODCD, QTY, AMT)  SELECT TO\_CHAR(SALEDATE, 'YYYYMMDDHH24'),  B.MEMBNO, B.EMPNO, A.PRODCD,  SUM(A.SALEQTY) QTY, SUM(A.SALEAMT) AMT  FROM SALELIST A, SALE B  where B.SALENO = A.SALENO AND A.PRODCD = I.PRODCD  AND B.SALEDATE >= to\_date('20151231','YYYYMMDD')  group by TO\_CHAR(SALEDATE, 'YYYYMMDDHH24'),  B.MEMBNO, B.EMPNO, A.PRODCD;  COMMIT;  END LOOP;  END;  7 기간별, 회원정보, 지점, 재품별 drill down 조회 기능을 이용해 mart 데이터를 검색합니다  8 동일 검색을 마트가 아닌 데이터 DB 에서 수행 할 경우 쿼리의 복잡도와 수행시간의 차이를 비교합니다  9 fact table의 첫번째 조건이 주어지는 경우와 그렇지 않은 경우의 검색속도 비교 |
| **기타** |  |
|  |  |