

최과장 보고서

Oracle —> mySQL로 수정

1. 전체 상품의 주문 완료 건 총 매출, 평균 매출, 최고매출, 최저매출을 출력하시오

```
SELECT COUNT(*) 전체주문건,  
SUM(B.sales) 총매출,  
AVG(B.sales) 평균매출,  
MAX(B.sales) 최고매출,  
MIN(B.sales) 최저매출  
FROM reservation A, order_info B  
WHERE A.reserv_no = B.reserv_no;
```

```
SELECT COUNT(*) 전체주문건,  
SUM(B.sales) 총매출,  
AVG(B.sales) 평균매출,  
MAX(B.sales) 최고매출,  
MIN(B.sales) 최저매출  
FROM reservation A  
INNER JOIN order_info B ON A.reserv_no = B.reserv_no;
```

2. 전체 상품의 총 판매량과 총 매출액, 전용상품의 판매량과 매출액을 출력하시오

```
SELECT COUNT(*) 총판매량,  
SUM(B.sales) 총매출,  
SUM(DECODE(B.item_id, 'M0001', 1, 0)) 전용상품판매량,  
SUM(DECODE(B.item_id, 'M0001', B.sales, 0)) 전용상품매출  
FROM reservation A, order_info B  
WHERE A.reserv_no = B.reserv_no  
AND A.cancel = 'N';
```

```
SELECT COUNT(*) 총판매량,  
SUM(B.sales) 총매출,  
SUM(CASE WHEN B.item_id = 'M0001' THEN 1 ELSE 0 END) 전용상품판매량,  
SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품매출  
FROM reservation A  
INNER JOIN order_info B ON A.reserv_no = B.reserv_no  
WHERE A.cancel = 'N';
```

3. 각 상품별 전체 매출액을 내림차순으로 출력하시오

```
SELECT C.item_id 상품아이디,  
C.product_name 상품명,  
SUM(B.sales) 상품매출  
FROM reservation A  
INNER JOIN order_info B ON A.reserv_no = B.reserv_no  
INNER JOIN item C ON B.item_id = C.item_id  
WHERE A.cancel = 'N'  
GROUP BY C.item_id, C.product_name  
ORDER BY SUM(B.sales) DESC;
```

```
SELECT C.item_id 상품아이디,  
C.product_name 상품명,  
SUM(B.sales) 상품매출  
FROM reservation A  
INNER JOIN order_info B ON A.reserv_no = B.reserv_no  
INNER JOIN item C ON B.item_id = C.item_id
```

```
WHERE A.cancel = 'N'
GROUP BY C.item_id, C.product_name
ORDER BY SUM(B.sales) DESC;
```

4. 모든 상품의 월별 매출액을 출력하시오

```
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
SUM(DECODE(B.item_id,'M0001',B.sales,0)) SPECIAL_SET,
SUM(DECODE(B.item_id,'M0002',B.sales,0)) PASTA,
SUM(DECODE(B.item_id,'M0003',B.sales,0)) PIZZA,
SUM(DECODE(B.item_id,'M0004',B.sales,0)) SEA_FOOD,
SUM(DECODE(B.item_id,'M0005',B.sales,0)) STEAK,
SUM(DECODE(B.item_id,'M0006',B.sales,0)) SALAD_BAR,
SUM(DECODE(B.item_id,'M0007',B.sales,0)) SALAD,
SUM(DECODE(B.item_id,'M0008',B.sales,0)) SANDWICH,
SUM(DECODE(B.item_id,'M0009',B.sales,0)) WINE,
SUM(DECODE(B.item_id,'M0010',B.sales,0)) JUICE
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no
AND A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUBSTR(A.reserv_date,1,6);
```

```
SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) SPECIAL_SET,
SUM(CASE WHEN B.item_id = 'M0002' THEN B.sales ELSE 0 END) PASTA,
SUM(CASE WHEN B.item_id = 'M0003' THEN B.sales ELSE 0 END) PIZZA,
SUM(CASE WHEN B.item_id = 'M0004' THEN B.sales ELSE 0 END) SEA_FOOD,
SUM(CASE WHEN B.item_id = 'M0005' THEN B.sales ELSE 0 END) STEAK,
SUM(CASE WHEN B.item_id = 'M0006' THEN B.sales ELSE 0 END) SALAD_BAR,
SUM(CASE WHEN B.item_id = 'M0007' THEN B.sales ELSE 0 END) SALAD,
SUM(CASE WHEN B.item_id = 'M0008' THEN B.sales ELSE 0 END) SANDWICH,
SUM(CASE WHEN B.item_id = 'M0009' THEN B.sales ELSE 0 END) WINE,
SUM(CASE WHEN B.item_id = 'M0010' THEN B.sales ELSE 0 END) JUICE
FROM reservation A
INNER JOIN order_info B ON A.reserv_no = B.reserv_no
WHERE A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date, 1, 6)
ORDER BY SUBSTR(A.reserv_date, 1, 6);
```

5. 월별 총 매출액과 전용 상품 매출액을 출력하시오

```
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
SUM(B.sales) 총매출,
SUM(DECODE(B.item_id,'M0001',B.sales,0)) 전용상품매출
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no
AND A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUBSTR(A.reserv_date,1,6);
```

```
SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
SUM(B.sales) 총매출,
SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품매출
FROM reservation A
INNER JOIN order_info B ON A.reserv_no = B.reserv_no
WHERE A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date, 1, 6)
ORDER BY SUBSTR(A.reserv_date, 1, 6);
```

6. 월별 총 매출액과 전용 상품 매출액을 출력에 매출 기여율을 추가하고 기여율은 소수점 아래 두번째 자리에서 반올림 하여 출력하시오

```
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
SUM(b.sales)
- SUM(decode(b.item_id,'M0001',b.sales,0)) 전용상품외매출,
```

```
SUM(decode(b.item_id,'M0001',b.sales,0)) 전용상품매출,
ROUND(SUM(DECODE(B.item_id,'M0001',B.sales,0))/SUM(B.sales)*100,1) 매출기여율
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no
AND A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUBSTR(A.reserv_date,1,6);
```

```
SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
SUM(B.sales) - SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품외매출,
SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품매출,
ROUND(SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) / SUM(B.sales) * 100, 1) 매출기여율
FROM reservation A
INNER JOIN order_info B ON A.reserv_no = B.reserv_no
WHERE A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date, 1, 6)
ORDER BY SUBSTR(A.reserv_date, 1, 6);
```

7. 6번에 총 예약건수, 예약 취소 건수를 추가하시오

```
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
SUM(B.sales) 총매출,
SUM(B.sales)
- SUM(decode(B.item_id,'M0001',B.sales,0)) 전용상품외매출,
SUM(DECODE(B.item_id,'M0001',B.sales,0)) 전용상품매출,
ROUND(SUM(DECODE(B.item_id,'M0001',B.sales,0))/SUM(B.sales)*100,1) 매출기여율,
COUNT(A.reserv_no) 총예약건,
SUM(DECODE(A.cancel,'N',1,0)) 예약완료건,
SUM(DECODE(A.cancel,'Y',1,0)) 예약취소건
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no
-- AND A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUBSTR(A.reserv_date,1,6);
```

```
SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
SUM(B.sales) 총매출,
SUM(B.sales) - SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품외매출,
SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품매출,
ROUND(SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) / SUM(B.sales) * 100, 1) 매출기여율,
COUNT(A.reserv_no) 총예약건,
SUM(CASE WHEN A.cancel = 'N' THEN 1 ELSE 0 END) 예약완료건,
SUM(CASE WHEN A.cancel = 'Y' THEN 1 ELSE 0 END) 예약취소건
FROM reservation A
INNER JOIN order_info B ON A.reserv_no = B.reserv_no
GROUP BY SUBSTR(A.reserv_date, 1, 6)
ORDER BY SUBSTR(A.reserv_date, 1, 6);
```

8. 7번에 총 매출 대비 전용 상품의 판매율, 총 예약건 대비 예약 취소율을 추가 소숫점이 나오면 두번째 자리에서 반올림하여 00.0%형식으로 출력하시오

```
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
SUM(B.sales) 총매출,
SUM(B.sales)
- SUM(DECODE(B.item_id,'M0001',B.sales,0)) 전용상품외매출,
SUM(DECODE(B.item_id,'M0001',B.sales,0)) 전용상품매출,
ROUND(SUM(DECODE(B.item_id,'M0001',B.sales,0))/SUM(B.sales)*100,1)|| '%' 전용상품판매율,
COUNT(A.reserv_no) 총예약건,
SUM(DECODE(A.cancel,'N',1,0)) 예약완료건,
SUM(DECODE(A.cancel,'Y',1,0)) 예약취소건,
ROUND(SUM(DECODE(A.cancel,'Y',1,0))/COUNT(A.reserv_no)*100,1)|| '%' 예약취소율
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no(+)
-- AND A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUBSTR(A.reserv_date,1,6);
```

```

SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
SUM(B.sales) 총매출,
SUM(B.sales) - SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품외매출,
SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품매출,
CONCAT(ROUND(SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) / SUM(B.sales) * 100, 1), '%') 전용상품판매율,
COUNT(A.reserv_no) 총예약건,
SUM(CASE WHEN A.cancel = 'N' THEN 1 ELSE 0 END) 예약완료건,
SUM(CASE WHEN A.cancel = 'Y' THEN 1 ELSE 0 END) 예약취소건,
CONCAT(ROUND(SUM(CASE WHEN A.cancel = 'Y' THEN 1 ELSE 0 END) / COUNT(A.reserv_no) * 100, 1), '%') 예약취소율
FROM reservation A
LEFT JOIN order_info B ON A.reserv_no = B.reserv_no
GROUP BY SUBSTR(A.reserv_date, 1, 6)
ORDER BY SUBSTR(A.reserv_date, 1, 6);

```

9. 월별 전용 상품 매출액을 일요일부터 월요일까지 구분해 출력해보세요.

```

SELECT SUBSTR(reserv_date,1,6) 날짜,
A.product_name 상품명,
SUM(DECODE(A.WEEK, '1', A.sales, 0)) 일요일,
SUM(DECODE(A.WEEK, '2', A.sales, 0)) 월요일,
SUM(DECODE(A.WEEK, '3', A.sales, 0)) 화요일,
SUM(DECODE(A.WEEK, '4', A.sales, 0)) 수요일,
SUM(DECODE(A.WEEK, '5', A.sales, 0)) 목요일,
SUM(DECODE(A.WEEK, '6', A.sales, 0)) 금요일,
SUM(DECODE(A.WEEK, '7', A.sales, 0)) 토요일
FROM
(
SELECT A.reserv_date,
C.product_name,
TO_CHAR(TO_DATE(A.reserv_date, 'YYYYMMDD'), 'd') WEEK,
B.sales
FROM reservation A, order_info B, item C
WHERE A.reserv_no = B.reserv_no
AND B.item_id = C.item_id
AND B.item_id = 'M0001'
) A
GROUP BY SUBSTR(reserv_date,1,6), A.product_name
ORDER BY SUBSTR(reserv_date,1,6);

```

```

SELECT SUBSTR(reserv_date,1,6) 날짜,
A.product_name 상품명,
SUM(CASE A.WEEK WHEN '1' THEN A.sales ELSE 0 END) 일요일,
SUM(CASE A.WEEK WHEN '2' THEN A.sales ELSE 0 END) 월요일,
SUM(CASE A.WEEK WHEN '3' THEN A.sales ELSE 0 END) 화요일,
SUM(CASE A.WEEK WHEN '4' THEN A.sales ELSE 0 END) 수요일,
SUM(CASE A.WEEK WHEN '5' THEN A.sales ELSE 0 END) 목요일,
SUM(CASE A.WEEK WHEN '6' THEN A.sales ELSE 0 END) 금요일,
SUM(CASE A.WEEK WHEN '7' THEN A.sales ELSE 0 END) 토요일
FROM(
SELECT A.reserv_date,
C.product_name,
DATE_FORMAT(STR_TO_DATE(A.reserv_date, '%Y%m%d'), '%w') WEEK,
B.sales
FROM reservation A, order_info B, item C
WHERE A.reserv_no = B.reserv_no
AND B.item_id = C.item_id
AND B.item_id = 'M0001'
) A
GROUP BY SUBSTR(reserv_date,1,6), A.product_name
ORDER BY SUBSTR(reserv_date,1,6);

```

1. 월별 전용 상품 매출 1위부터 3위까지 지점이 어딘지 확인하시오

```

SELECT *
FROM

```

```
(
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
A.branch 지점,
SUM(B.sales) 전용상품매출,
RANK() OVER(PARTITION BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUM(B.sales) DESC) 지점순위
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no
AND A.cancel = 'N'
AND B.item_id = 'M0001'
GROUP BY SUBSTR(A.reserv_date,1,6), A.branch
ORDER BY SUBSTR(A.reserv_date,1,6)
) A
WHERE A.지점순위 <= 3;
```

```
SELECT *
FROM
(
SELECT
SUBSTR(A.reserv_date, 1, 6) 매출월,
A.branch 지점,
SUM(B.sales) 전용상품매출,
RANK() OVER (PARTITION BY SUBSTR(A.reserv_date, 1, 6) ORDER BY SUM(B.sales) DESC) 지점순위
FROM
reservation A
INNER JOIN
order_info B ON A.reserv_no = B.reserv_no
WHERE
A.cancel = 'N'
AND B.item_id = 'M0001'
GROUP BY
SUBSTR(A.reserv_date, 1, 6), A.branch
ORDER BY
SUBSTR(A.reserv_date, 1, 6)
) AS A
WHERE
A.지점순위 <= 3;
```

11. 분석8의 결과와 분석10의 결과항목을 월별로 합쳐서 리포트로 작성하시오

```
SELECT A.매출월, 매출월,
MAX(총매출) 총매출,
MAX(전용상품외매출) 전용상품외매출,
MAX(전용상품매출) 전용상품매출,
MAX(전용상품판매율) 전용상품판매율,
MAX(총예약건) 총예약건,
MAX(예약완료건) 예약완료건,
MAX(예약취소건) 예약취소건,
MAX(예약취소율) 예약취소율,
MAX(최대매출지점) 최대매출지점,
MAX(지점매출액) 지점매출액
FROM
(
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
SUM(B.sales) 총매출,
SUM(B.sales)
- SUM(DECODE(B.item_id, 'M0001', B.sales, 0)) 전용상품외매출,
SUM(DECODE(B.item_id, 'M0001', B.sales, 0)) 전용상품매출,
ROUND(SUM(DECODE(B.item_id, 'M0001', B.sales, 0))/SUM(B.sales)*100, 1)|| '%' 전용상품판매율,
COUNT(A.reserv_no) 총예약건,
SUM(DECODE(A.cancel, 'N', 1, 0)) 예약완료건,
SUM(DECODE(A.cancel, 'Y', 1, 0)) 예약취소건,
ROUND(SUM(DECODE(A.cancel, 'Y', 1, 0))/COUNT(A.reserv_no)*100, 1)|| '%' 예약취소율,
' ' 최대매출지점,
0 지점매출액
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no(+)
-- AND A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date,1,6), ' ', 0
UNION
SELECT A.매출월,
0 총매출,
0 전용상품외매출,
0 전용상품매출,
```

```

    ' '      전용상품판매율,
    0      총예약건,
    0      예약완료건,
    0      예약취소건,
    ' '      예약취소율,
A.지점      최대매출지점,
A.전용상품매출 지점매출액
FROM
(
SELECT SUBSTR(A.reserv_date,1,6) 매출월,
A.branch      지점,
SUM(B.sales)      전용상품매출,
ROW_NUMBER() OVER(PARTITION BY SUBSTR(A.reserv_date,1,6)
ORDER BY SUM(B.sales) DESC) 지점순위,
DECODE(A.branch, '강남', 'A', '종로', 'A', '영등포', 'A', 'B') 지점등급
FROM reservation A, order_info B
WHERE A.reserv_no = B.reserv_no
AND A.cancel = 'N'
AND B.item_id = 'M0001'
GROUP BY SUBSTR(A.reserv_date,1,6), A.branch,
DECODE(A.branch, '강남', 'A', '종로', 'A', '영등포', 'A', 'B')
ORDER BY SUBSTR(A.reserv_date,1,6)
) A
WHERE A.지점순위 = 1
-- AND 지점등급 = 'A'
) A
GROUP BY A.매출월
ORDER BY A.매출월;

```

```

SELECT A.매출월 매출월,
      MAX( 총매출 ) 총매출,
      MAX( 전용상품외매출 ) 전용상품외매출,
      MAX( 전용상품매출 ) 전용상품매출,
      MAX( 전용상품판매율 ) 전용상품판매율,
      MAX( 총예약건 ) 총예약건,
      MAX( 예약완료건 ) 예약완료건,
      MAX( 예약취소건 ) 예약취소건,
      MAX( 예약취소율 ) 예약취소율,
      MAX( 최대매출지점 ) 최대매출지점,
      MAX( 지점매출액 ) 지점매출액
FROM
(
SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
      SUM(B.sales) 총매출,
      SUM(B.sales) - SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품외매출,
      SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) 전용상품매출,
      CONCAT(ROUND(SUM(CASE WHEN B.item_id = 'M0001' THEN B.sales ELSE 0 END) / SUM(B.sales) * 100, 1), '%') 전용상품판매율,
      COUNT(A.reserv_no) 총예약건,
      SUM(CASE WHEN A.cancel = 'N' THEN 1 ELSE 0 END) 예약완료건,
      SUM(CASE WHEN A.cancel = 'Y' THEN 1 ELSE 0 END) 예약취소건,
      CONCAT(ROUND(SUM(CASE WHEN A.cancel = 'Y' THEN 1 ELSE 0 END) / COUNT(A.reserv_no) * 100, 1), '%') 예약취소율,
      ' ' 최대매출지점,
      0 지점매출액
FROM reservation A
LEFT JOIN order_info B ON A.reserv_no = B.reserv_no
WHERE A.cancel = 'N'
GROUP BY SUBSTR(A.reserv_date, 1, 6)
UNION
SELECT A.매출월,
      0 총매출,
      0 전용상품외매출,
      0 전용상품매출,
      ' ' 전용상품판매율,
      0 총예약건,
      0 예약완료건,
      0 예약취소건,
      ' ' 예약취소율,
      A.지점 최대매출지점,
      A.전용상품매출 지점매출액
FROM
(
SELECT SUBSTR(A.reserv_date, 1, 6) 매출월,
      A.branch 지점,
      SUM(B.sales) 전용상품매출,
      ROW_NUMBER() OVER (PARTITION BY SUBSTR(A.reserv_date, 1, 6) ORDER BY SUM(B.sales) DESC) 지점순위,
      CASE

```

```

        WHEN A.branch IN ('강남', '종로', '영등포') THEN 'A'
        ELSE 'B'
    END 지점등급
FROM reservation A
JOIN order_info B ON A.reserv_no = B.reserv_no
WHERE A.cancel = 'N' AND B.item_id = 'M0001'
GROUP BY SUBSTR(A.reserv_date, 1, 6), A.branch,
        CASE
            WHEN A.branch IN ('강남', '종로', '영등포') THEN 'A'
            ELSE 'B'
        END
ORDER BY SUBSTR(A.reserv_date, 1, 6)
) A
WHERE A.지점순위 = 1
) A
GROUP BY A.매출월
ORDER BY A.매출월;

```

12. 고객의 수 ,남녀숫자,평균나이,평균거래기간을 출력하시오

```

SELECT COUNT(customer_id) 고객수,
SUM(DECODE(sex_code, 'M', 1, 0)) 남자,
SUM(DECODE(sex_code, 'F', 1, 0)) 여자,
ROUND(AVG(MONTHS_BETWEEN(TO_DATE('20171231', 'YYYYMMDD'), TO_DATE(birth, 'YYYYMMDD'))/12), 1) 평균나이,
ROUND(AVG(MONTHS_BETWEEN(TO_DATE('20171231', 'YYYYMMDD'), first_reg_date)), 1) 평균거래기간
FROM customer;

```

```

SELECT
COUNT(customer_id) 고객수,
SUM(CASE WHEN sex_code = 'M' THEN 1 ELSE 0 END) 남자,
SUM(CASE WHEN sex_code = 'F' THEN 1 ELSE 0 END) 여자,
ROUND(AVG(DATEDIFF('2017-12-31', STR_TO_DATE(birth, '%Y%m%d')) / 365.25), 1) 평균나이,
ROUND(AVG(DATEDIFF('2017-12-31', first_reg_date), 1)) 평균거래기간
FROM customer;

```

13. 개인별 전체 상품 주문 건수, 총 매출,전용상품주문건수,전용상품매출을 출력하여 전용상품의 매출 기준으로 내림차순 정렬하시오

```

SELECT A.customer_id      고객아이디,
A.customer_name      고객이름,
COUNT(C.order_no) 전체상품주문건수,
SUM(C.sales) 총매출,
SUM(DECODE(C.item_id, 'M0001', 1, 0)) 전용상품주문건수,
SUM(DECODE(C.item_id, 'M0001', C.sales, 0)) 전용상품매출
FROM customer A, reservation B, order_info C
WHERE A.customer_id = B.customer_id
AND B.reserv_no = C.reserv_no
AND B.cancel = 'N'
GROUP BY A.customer_id, A.customer_name
ORDER BY SUM(DECODE(C.item_id, 'M0001', C.sales, 0)) DESC;

```

```

SELECT
A.customer_id 고객아이디,
A.customer_name 고객이름,
COUNT(C.order_no) 전체상품주문건수,
SUM(C.sales) 총매출,
SUM(CASE WHEN C.item_id = 'M0001' THEN 1 ELSE 0 END) 전용상품주문건수,
SUM(CASE WHEN C.item_id = 'M0001' THEN C.sales ELSE 0 END) 전용상품매출
FROM
customer A
JOIN
reservation B ON A.customer_id = B.customer_id
JOIN
order_info C ON B.reserv_no = C.reserv_no
WHERE
B.cancel = 'N'
GROUP BY
A.customer_id, A.customer_name

```

```
ORDER BY
SUM(CASE WHEN C.item_id = 'M0001' THEN C.sales ELSE 0 END) DESC;
```

14. 상품을 구매한 전체 고객의 거주지와 전용상품을 구매한 고객의 거주지를 각각 비교해보고 상품을 구매한 전체 고객의 직업과 전용 상품을 구매한 고객의 직업을 각각 비교하시오

```
SELECT B.address_detail 주소, B.zip_code, COUNT(B.address_detail) 카운팅
FROM (
SELECT DISTINCT A.customer_id, A.zip_code
FROM customer A, reservation B, order_info C
WHERE A.customer_id = B.customer_id
AND B.reserv_no = C.reserv_no
AND B.cancel = 'N'
-- AND C.item_id = 'M0001'
) A, address B
WHERE A.zip_code = B.zip_code
GROUP BY B.address_detail, B.zip_code
ORDER BY COUNT(B.address_detail) DESC;
```

```
SELECT NVL(B.job, '정보없음') 직업, COUNT(NVL(B.job,1)) 카운팅
FROM (
SELECT DISTINCT A.customer_id, A.zip_code
FROM customer A, reservation B, order_info C
WHERE A.customer_id = B.customer_id
AND B.reserv_no = C.reserv_no
AND B.cancel = 'N'
-- AND C.item_id = 'M0001'
) A, customer B
WHERE A.customer_id = B.customer_id
GROUP BY NVL(B.job, '정보없음')
ORDER BY COUNT(NVL(B.job,1)) DESC;
```

```
SELECT
B.address_detail 주소,
B.zip_code,
COUNT(B.address_detail) 카운팅
FROM (
SELECT DISTINCT A.customer_id, A.zip_code
FROM customer A
JOIN reservation B ON A.customer_id = B.customer_id
JOIN order_info C ON B.reserv_no = C.reserv_no
WHERE B.cancel = 'N'
-- AND C.item_id = 'M0001'
) A
JOIN address B ON A.zip_code = B.zip_code
GROUP BY B.address_detail, B.zip_code
ORDER BY 카운팅 DESC;
```

```
SELECT
COALESCE(B.job, '정보없음') 직업,
COUNT(COALESCE(B.job, 1)) 카운팅
FROM (
SELECT DISTINCT A.customer_id, A.zip_code
FROM customer A
JOIN reservation B ON A.customer_id = B.customer_id
JOIN order_info C ON B.reserv_no = C.reserv_no
WHERE B.cancel = 'N'
-- AND C.item_id = 'M0001'
) A
JOIN customer B ON A.customer_id = B.customer_id
GROUP BY COALESCE(B.job, '정보없음')
ORDER BY 카운팅 DESC;
```

15. 전용 상품 매출 기준 상위 10위 고객을 확인하시오


```

SELECT B.address_detail 주소, B.zip_code, COUNT(B.address_detail) 카운팅
FROM (
SELECT DISTINCT A.customer_id, A.zip_code
FROM customer A, reservation B, order_info C
WHERE A.customer_id = B.customer_id
AND B.reserv_no = C.reserv_no
AND B.cancel = 'N'
-- AND C.item_id = 'M0001'
) A, address B
WHERE A.zip_code = B.zip_code
GROUP BY B.address_detail, B.zip_code
ORDER BY COUNT(B.address_detail) DESC;

```

```

SELECT
B.address_detail 주소,
B.zip_code,
COUNT(B.address_detail) 카운팅
FROM (
SELECT DISTINCT A.customer_id, A.zip_code
FROM customer A
JOIN reservation B ON A.customer_id = B.customer_id
JOIN order_info C ON B.reserv_no = C.reserv_no
WHERE B.cancel = 'N'
-- AND C.item_id = 'M0001'
) A
JOIN address B ON A.zip_code = B.zip_code
GROUP BY B.address_detail, B.zip_code
ORDER BY 카운팅 DESC;

```

16. 전용 상품 매출 상위 10위 이상 고객이 두번째로 선호하는 상품 확인하시오

```

SELECT *
FROM (
SELECT A.고객아이디,
A.고객이름,
D.product_name 상품명,
SUM(C.sales) 상품매출,
RANK() OVER(PARTITION BY A.고객아이디 ORDER BY SUM(C.sales) DESC) 선호도순위
FROM
(
SELECT A.customer_id 고객아이디,
A.customer_name 고객이름,
SUM(C.sales) 전용상품_매출
FROM customer A, reservation B, order_info C
WHERE A.customer_id = B.customer_id
AND B.reserv_no = C.reserv_no
AND B.cancel = 'N'
AND C.item_id = 'M0001'
GROUP BY A.customer_id, A.customer_name
HAVING SUM(C.sales) >= 216000
) A, reservation B, order_info C, item D
WHERE A.고객아이디 = B.customer_id
AND B.reserv_no = C.reserv_no
AND C.item_id = D.item_id
AND D.item_id <> 'M0001'
AND B.cancel = 'N'
GROUP BY A.고객아이디, A.고객이름, D.product_name
) A
WHERE A.선호도순위 = 1;

```

```

SELECT *
FROM (
SELECT A.고객아이디,
A.고객이름,
D.product_name 상품명,
SUM(C.sales) 상품매출,
RANK() OVER (PARTITION BY A.고객아이디 ORDER BY SUM(C.sales) DESC) 선호도순위
FROM (
SELECT A.customer_id 고객아이디,

```

```

A.customer_name 고객이름,
SUM(C.sales) 전용상품_매출
FROM customer A
JOIN reservation B ON A.customer_id = B.customer_id
JOIN order_info C ON B.reserv_no = C.reserv_no
WHERE B.cancel = 'N'
AND C.item_id = 'M0001'
GROUP BY A.customer_id, A.customer_name
HAVING SUM(C.sales) >= 216000
) A
JOIN reservation B ON A.고객아이디 = B.customer_id
JOIN order_info C ON B.reserv_no = C.reserv_no
JOIN item D ON C.item_id = D.item_id
WHERE D.item_id <> 'M0001'
AND B.cancel = 'N'
GROUP BY A.고객아이디, A.고객이름, D.product_name
) A
WHERE A.선호도순위 = 1;

```