

## 8-2. 종합실습 1

홍형경

[chariehong@gmail.com](mailto:chariehong@gmail.com)

2020.01

# 1. 종합 실습 – 로또 데이터 분석

lotto\_master 테이블

컬럼	데이터형	Null	PK	설명
SEQ_NO	NUMBER	N	Y	로또회차
DRAW_DATE	DATE	Y		추첨일
NUM1	NUMBER	Y		당첨번호1
NUM2	NUMBER	Y		당첨번호2
NUM3	NUMBER	Y		당첨번호3
NUM4	NUMBER	Y		당첨번호4
NUM5	NUMBER	Y		당첨번호5
NUM6	NUMBER	Y		당첨번호6
BONUS	NUMBER	Y		보너스번호

lotto\_detail 테이블

컬럼	데이터형	Null	PK	설명
SEQ_NO	NUMBER	N	Y	로또회차
RANK_NO	DATE	N	Y	등수
WIN_PERSON_NO	NUMBER	Y		당첨자수
WIN_MONEY	NUMBER	Y		1인당 당첨금액

# 1. 종합 실습 – 로또 데이터 분석 (827회까지)

· lotto\_master 테이블 생성

```
CREATE TABLE lotto_master (  
  seq_no    NUMBER NOT NULL, -- 로또회차  
  draw_date DATE,           -- 추첨일  
  num1      NUMBER,         -- 당첨번호1  
  num2      NUMBER,         -- 당첨번호2  
  num3      NUMBER,         -- 당첨번호3  
  num4      NUMBER,         -- 당첨번호4  
  num5      NUMBER,         -- 당첨번호5  
  num6      NUMBER,         -- 당첨번호6  
  bonus     NUMBER          -- 보너스번호  
);
```

```
ALTER TABLE lotto_master  
ADD CONSTRAINTS lotto_master_pk PRIMARY  
KEY (seq_no);
```

# 1. 종합 실습 – 로또 데이터 분석 (827회까지)

- lotto\_detail 테이블 생성

```
CREATE TABLE lotto_detail (  
    seq_no      NUMBER NOT NULL, -- 로또회차  
    rank_no     NUMBER NOT NULL, -- 등수  
    win_person_no NUMBER,        -- 당첨자수  
    win_money   NUMBER          -- 1인당 당첨금액  
);
```

```
ALTER TABLE lotto_detail  
ADD CONSTRAINTS lotto_detail_pk PRIMARY KEY (seq_no, rank_no);
```

# 1. 종합 실습 – 로또 데이터 분석 (827회까지)

- lotto\_master, lotto\_detail 테이블 데이터 생성

**03.lotto\_data\_insert.sql** 파일 실행

# 1. 종합 실습 – 로또 데이터 분석

```
SELECT *  
    FROM lotto_master  
    ORDER BY 1;
```

	SEQ_NO	DRAW_DATE	NUM1	NUM2	NUM3	NUM4	NUM5	NUM6	BONUS
1	1	2002-12-07 00:00:00	10	23	29	33	37	40	16
2	2	2002-12-14 00:00:00	9	13	21	25	32	42	2
3	3	2002-12-21 00:00:00	11	16	19	21	27	31	30
4	4	2002-12-28 00:00:00	14	27	30	31	40	42	2
5	5	2003-01-04 00:00:00	16	24	29	40	41	42	3
6	6	2003-01-11 00:00:00	14	15	26	27	40	42	34
7	7	2003-01-18 00:00:00	2	9	16	25	26	40	42
8	8	2003-01-25 00:00:00	8	19	25	34	37	39	9
9	9	2003-02-01 00:00:00	2	4	16	17	36	39	14

```
SELECT *  
    FROM lotto_detail  
    ORDER BY 1, 2;
```

	SEQ_NO	RANK_NO	WIN_PERSON_NO	WIN_MONEY
1	1	1	0	0
2	1	2	1	143934100
3	1	3	28	5140500
4	1	4	2537	50000
5	1	5	40155	5000
6	2	1	1	2002006800
7	2	2	2	94866800
8	2	3	103	1842000
9	2	4	3763	50000

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (1) 중복 번호 조회

- num1 컬럼에 대해 1회 ~ 827회까지 중복 번호는?

- 집계쿼리 사용

```
SELECT num1, COUNT(*)  
FROM lotto_master  
GROUP BY num1  
ORDER BY 1;
```

	NUM1	COUNT(*)
1	1	119
2	2	90
3	3	79
4	4	76
5	5	71
6	6	64
7	7	51
8	8	41
9	9	25
10	10	34
11	11	31
12	12	28
13	13	21
14	14	23
15	15	11
16	16	17
17	17	15
18	18	5
19	19	8
20	20	3
21	21	3
22	22	3
23	23	2
24	24	2
25	25	1

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (1) 중복 번호 조회

- num1 ~ num6 컬럼에 대해 1회 ~ 827회까지 중복 번호는?

```
SELECT num1 ,num2 ,num3 ,num4 ,num5 ,num6 , COUNT(*)  
FROM lotto_master  
GROUP BY num1, num2, num3, num4, num5, num6  
ORDER BY 1, 2, 3, 4, 5, 6;
```

NUM1	NUM2	NUM3	NUM4	NUM5	NUM6	COUNT(*)
1	2	3	9	12	23	1
1	2	3	14	27	42	1
1	2	3	15	20	25	1
1	2	4	8	19	38	1
1	2	4	23	31	34	1
1	2	5	11	18	36	1
1	2	6	9	25	28	1
1	2	6	16	19	42	1
1	2	6	16	20	33	1
1	2	7	9	10	38	1
1	2	8	17	26	37	1
1	2	8	18	29	38	1
1	2	9	17	19	42	1
1	2	10	13	18	19	1
1	2	10	25	26	44	1
1	2	15	19	24	36	1
1	2	15	28	34	45	1
1	2	23	25	38	40	1
1	3	4	6	14	41	1
1	3	7	8	24	42	1
-	-	-	-	-	-	-



# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (1) 중복 번호 조회

- num1 ~ num6 컬럼에 대해 1회 ~ 827회까지 중복 번호는?

```
SELECT num1 ,num2 ,num3 ,num4 ,num5 ,num6 , COUNT(*)  
FROM lotto_master  
GROUP BY num1, num2, num3, num4, num5, num6  
HAVING COUNT(*) > 1  
ORDER BY 1, 2, 3, 4, 5, 6;
```

NUM1		NUM2	NUM3	NUM4	NUM5	NUM6	COUNT(*)
------	--	------	------	------	------	------	----------

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (2) 가장 많이 당첨된 번호 조회

- num1 컬럼에 대해 1회 ~ 827회까지 가장 많이 당첨된 번호는?

```
SELECT num1 lotto_num, COUNT(*) CNT  
FROM lotto_master  
GROUP BY num1  
ORDER BY 2 DESC;
```

	↕ LOTTO_NUM	↕ CNT
1	1	119
2	2	90
3	3	79
4	4	76
5	5	71
6	6	64
7	7	51
8	8	41
9	10	34
10	11	31
11	12	28
12	9	25

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (2) 가장 많이 당첨된 번호 조회

- num1과 num2 컬럼 통틀어 가장 많이 당첨된 번호는?

```
SELECT num1 lotto_num, COUNT(*) CNT
FROM lotto_master
GROUP BY num1
UNION ALL
SELECT num2 lotto_num, COUNT(*) CNT
FROM lotto_master
GROUP BY num2
ORDER BY 1;
```

	LOTTO_NUM	CNT
1	1	119
2	2	90
3	2	18
4	3	27
5	3	79
6	4	76
7	4	32
8	5	37
9	5	71
10	6	64
11	6	27
12	7	51
13	7	51
14	8	54
15	8	41
..	..	..

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (2) 가장 많이 당첨된 번호 조회

- num1과 num2 컬럼 통틀어 가장 많이 당첨된 번호는?

```
SELECT lotto_num, SUM(CNT) AS CNT
FROM ( SELECT num1 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num1
        UNION ALL
        SELECT num2 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num2
      )
GROUP BY lotto_num
ORDER BY 2 DESC;
```

	LOTTO_NUM	CNT
1	1	119
2	4	108
3	5	108
4	2	108
5	3	106
6	7	102
7	8	95
8	6	91
9	10	90
10	12	80
11	11	79
12	9	66
13	14	61
14	13	59
15	15	52
16	17	50

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (2) 가장 많이 당첨된 번호 조회

- num1 ~ num6 컬럼 통틀어 가장 많이 당첨된 번호는?

```
SELECT lotto_num, SUM(CNT) AS CNT
FROM ( SELECT num1 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num1
        UNION ALL
        SELECT num2 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num2
        UNION ALL
        SELECT num3 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num3
```

```
        UNION ALL
        SELECT num4 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num4
        UNION ALL
        SELECT num5 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num5
        UNION ALL
        SELECT num6 lotto_num, COUNT(*) CNT
        FROM lotto_master
        GROUP BY num6
    )
GROUP BY lotto_num
ORDER BY 2 DESC;
```

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (2) 가장 많이 당첨된 번호 조회

- num1 ~ num6 컬럼 통틀어 가장 많이 당첨된 번호는?

	LOTTO_NUM	CNT
1	34	131
2	27	125
3	40	123
4	20	122
5	12	120
6	17	119
7	1	119
8	10	118
9	43	116
10	37	116
11	14	116
12	18	116
13	45	115
14	11	115
15	13	115
16	8	114
17	5	114
..	..	..

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

## (3) 가장 많은 당첨금이 나온 회차와 번호, 금액 조회

- 가장 많은 당첨금? → 1등
- 1등 → lotto\_detail 테이블의 rank\_no = 1
- 1인당 당첨 금액 → lotto\_detail 테이블의 win\_money
- 해당 로또 번호 → lotto\_master 테이블
- lotto\_master와 lotto\_detail 테이블 조인

➔ 두 테이블 조인하고, 1등을 구한 다음 당첨금액 순으로 내림차순 정렬

# 1. 종합 실습 – 로또 데이터 분석 (06.lotto\_sql.sql 파일 참조)

(3) 가장 많은 당첨금이 나온 회차와 번호, 금액 조회

```
SELECT a.seq_no
       ,a.draw_date
       ,b.win_person_no
       ,b.win_money
       ,a.num1 ,a.num2 ,a.num3
       ,a.num4 ,a.num5 ,a.num6 ,a.bonus
FROM lotto_master a
     ,lotto_detail b
WHERE a.seq_no = b.seq_no
     AND b.rank_no = 1
ORDER BY b.win_money DESC;
```

	SEQ_NO	DRAW_DATE	WIN_PERSON_NO	WIN_MONEY	NUM1	NUM2	NUM3	NUM4	NUM5	NUM6	BONUS
1	19	2003-04-12 00:00:00	1	40722959400	6	30	38	39	40	43	26
2	25	2003-05-24 00:00:00	2	24227745300	2	4	21	26	43	44	16
3	20	2003-04-19 00:00:00	1	19352212800	10	14	18	20	23	30	41
4	43	2003-09-27 00:00:00	1	17749630800	6	31	35	38	39	44	1
5	15	2003-03-15 00:00:00	1	17014245000	3	4	16	30	31	37	13
6	36	2003-08-09 00:00:00	1	16014475800	1	10	23	26	28	40	31
7	62	2004-02-07 00:00:00	1	15817286400	3	8	15	27	29	35	21
8	33	2003-07-19 00:00:00	1	14903517600	4	7	32	33	40	41	9
9	82	2004-06-26 00:00:00	1	14562494400	1	2	3	14	27	42	39
10	86	2004-07-24 00:00:00	1	14252186400	2	12	37	39	41	45	33



## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

traffic\_accident 테이블

컬럼	데이터형	Null	PK	설명
YEAR	NUMBER	N	Y	년도
TRANS_TYPE	VARCHAR2(30)	N	Y	교통수단
TOTAL_ACCT_NUM	NUMBER	Y		사고발생건수
DEATH_PERSON_NUM	NUMBER	Y		사망자수

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

- traffic\_accident 테이블 생성

```
CREATE TABLE traffic_accident (  
    year          NUMBER    NOT NULL, -- 연도  
    trans_type     VARCHAR2(30) NOT NULL, -- 교통수단  
    total_acct_num NUMBER,      -- 사고발생건수  
    death_person_num NUMBER      -- 사망자수  
);
```

```
ALTER TABLE traffic_accident  
ADD CONSTRAINTS traffic_accident_pk PRIMARY KEY (year, trans_type);
```

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

- traffic\_accident 테이블 데이터 입력

04.traffic\_accident\_insert.sql 파일

```
SELECT *  
FROM traffic_accident;
```

	YEAR	TRANS_TYPE	TOTAL_ACCT_NUM	DEATH_PERSON_NUM
1	1980	자동차	120182	5608
2	1981	자동차	123373	5804
3	1982	자동차	141128	6110
4	1983	자동차	170026	6834
5	1984	자동차	134335	7468
6	1985	자동차	146836	7522
7	1986	자동차	153777	7702
8	1987	자동차	175661	7206
9	1988	자동차	225062	11563
10	1989	자동차	255787	12603
11	1990	자동차	255303	12325
12	1991	자동차	265964	13429
13	1992	자동차	257194	11640
14	1993	자동차	260921	10402
15	1994	자동차	266107	10087
16	1995	자동차	248865	10323
17	1996	자동차	265052	12653

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (1) 연도, 교통수단별 총 사고건수 조회

- 연도(year), 교통수단(trans\_type), 사고건수(total\_acct\_num)
- 집계쿼리 → 연도, 교통수단 별로 사고건수 SUM

```
SELECT YEAR
      ,trans_type
      ,SUM(total_acct_num) AS 사고건수
      ,SUM(death_person_num) AS 사망자수
FROM traffic_accident
WHERE 1=1
GROUP BY YEAR, trans_type
ORDER BY 1, 2;
```

	YEAR	TRANS_TYPE	사고건수	사망자수
1	1980	선박	255	147
2	1980	자동차	120182	5608
3	1980	지하철	7	4
4	1980	철도	2136	778
5	1980	항공기	4	17
6	1981	선박	336	123
7	1981	자동차	123373	5804
8	1981	지하철	15	7
9	1981	철도	1894	858
10	1981	항공기	2	0
11	1982	선박	301	42
12	1982	자동차	141128	6110
13	1982	지하철	8	4
14	1982	철도	1849	755
15	1982	항공기	0	0
16	1983	선박	300	74
17	1983	자동차	170026	6834

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (1) 연도, 교통수단별 총 사고건수 조회

	YEAR	TRANS_TYPE	사고건수	사망자수
1	1980	선박	255	147
2	1980	자동차	120182	5608
3	1980	지하철	7	4
4	1980	철도	2136	778
5	1980	항공기	4	17
6	1981	선박	336	123
7	1981	자동차	123373	5804
8	1981	지하철	15	7
9	1981	철도	1894	858
10	1981	항공기	2	0
11	1982	선박	301	42
12	1982	자동차	141128	6110
13	1982	지하철	8	4
14	1982	철도	1849	755
15	1982	항공기	0	0
16	1983	선박	300	74
17	1983	자동차	170026	6834

- 원하는 결과는 얻었으나 결과 로우가 너무 많아 한 눈에 파악하기 힘들
- 개별 연도 별이 아닌 년대(예, 1980년대) 별로 다시 집계해보자.

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (1) 연도, 교통수단별 총 사고건수 조회

```
SELECT CASE WHEN year BETWEEN 1980 AND 1989 THEN '1980년대'
           WHEN year BETWEEN 1990 AND 1999 THEN '1990년대'
           WHEN year BETWEEN 2000 AND 2009 THEN '2000년대'
           WHEN year BETWEEN 2010 AND 2019 THEN '2010년대'
           END AS YEARS
       ,trans_type
       ,SUM(total_acct_num) AS 사고건수
FROM traffic_accident
WHERE 1=1
GROUP BY CASE WHEN year BETWEEN 1980 AND 1989 THEN '1980년대'
           WHEN year BETWEEN 1990 AND 1999 THEN '1990년대'
           WHEN year BETWEEN 2000 AND 2009 THEN '2000년대'
           WHEN year BETWEEN 2010 AND 2019 THEN '2010년대'
           END, trans_type
ORDER BY 1, 2;
```

	YEARS	TRANS_TYPE	사고건수
1	1980년대	선박	3965
2	1980년대	자동차	1646167
3	1980년대	지하철	208
4	1980년대	철도	18681
5	1980년대	항공기	20
6	1990년대	선박	6605
7	1990년대	자동차	2581517
8	1990년대	지하철	311
9	1990년대	철도	13415
10	1990년대	항공기	24
11	2000년대	선박	7780
12	2000년대	자동차	2331063
13	2000년대	지하철	732
14	2000년대	철도	5083
15	2000년대	항공기	63
16	2010년대	선박	11840
17	2010년대	자동차	1564103
18	2010년대	지하철	17
19	2010년대	철도	1546
20	2010년대	항공기	77

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (2) 연대별 교통사고 추이 분석

- 교통사고 유형을 로우로, 연대를 컬럼으로 두고 교통사고 건수 조회
- 1980년대에 발생한 교통사고 총 건수를 컬럼으로 만들려면?

```
CASE WHEN year  
        BETWEEN 1980 AND 1989  
        THEN total_acct_num ELSE 0 END
```

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (2) 연대별 교통사고 추이 분석

```
SELECT trans_type
      ,CASE WHEN year BETWEEN 1980 AND 1989 THEN total_acct_num ELSE 0 END "1980년대"
      ,CASE WHEN year BETWEEN 1990 AND 1999 THEN total_acct_num ELSE 0 END "1990년대"
      ,CASE WHEN year BETWEEN 2000 AND 2009 THEN total_acct_num ELSE 0 END "2000년대"
      ,CASE WHEN year BETWEEN 2010 AND 2019 THEN total_acct_num ELSE 0 END "2010년대"
FROM traffic_accident
WHERE 1=1
ORDER BY trans_type;
```



## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (2) 연대별 교통사고 추이 분석

	TRANS_TYPE	1980년대	1990년대	2000년대	2010년대
1	선택	255	0	0	0
2	선택	0	0	0	2307
3	선택	301	0	0	0
4	선택	300	0	0	0
5	선택	486	0	0	0
6	선택	408	0	0	0
7	선택	429	0	0	0
8	선택	533	0	0	0
9	선택	438	0	0	0
10	선택	479	0	0	0
11	선택	0	515	0	0
12	선택	0	555	0	0
13	선택	0	476	0	0
14	선택	0	529	0	0
15	선택	0	699	0	0
16	선택	0	709	0	0
17	선택	0	661	0	0
18	선택	0	840	0	0
19	선택	0	772	0	0
20	선택	0	849	0	0
21	선택	0	0	634	0
22	선택	0	0	610	0
23	선택	0	0	557	0
24	선택	0	0	531	0
25	선택	0	0	804	0
26	선택	0	0	658	0
27	선택	0	0	657	0
28	선택	0	0	555	0

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (2) 연대별 교통사고 추이 분석

```
SELECT trans_type
      ,SUM(CASE WHEN year BETWEEN 1980 AND 1989 THEN total_acct_num ELSE 0 END) "1980년대"
      ,SUM(CASE WHEN year BETWEEN 1990 AND 1999 THEN total_acct_num ELSE 0 END) "1990년대"
      ,SUM(CASE WHEN year BETWEEN 2000 AND 2009 THEN total_acct_num ELSE 0 END) "2000년대"
      ,SUM(CASE WHEN year BETWEEN 2010 AND 2019 THEN total_acct_num ELSE 0 END) "2010년대"
FROM traffic_accident
WHERE 1=1
GROUP BY trans_type
ORDER BY trans_type;
```

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (2) 연대별 교통사고 추이 분석

TRANS_TYPE	1980년대	1990년대	2000년대	2010년대
1 선박	3965	6605	7780	11840
2 자동차	1646167	2581517	2331063	1564103
3 지하철	208	311	732	17
4 철도	18681	13415	5083	1546
5 항공기	20	24	63	77

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

(3) 교통수단별 가장 많은 사망자 수가 발생한 연도 구하기

- 사망자 수 → death\_person\_num 컬럼
- 교통수단 별 **가장 많은** 사망자 수  
→ trans\_type을 group by, **MAX**(사망자수)

```
SELECT trans_type
      ,MAX(death_person_num) death_per
FROM traffic_accident
GROUP BY trans_type;
```

	TRANS_TYPE	DEATH_PER
1	지하철	256
2	철도	858
3	항공기	276
4	선박	477
5	자동차	13429

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

### (3) 교통수단별 가장 많은 사망자 수가 발생한 연도 구하기

- 교통수단별 가장 많은 사망자 수까지 구했으니, 연도만 구하면 됨
- 교통수단 별 가장 많은 사망자 수를 교통수단별 사망자 수가 같은 건을 구하면?  
➔ 가장 많은 사망자 수가 발생한 연도를 구할 수 있음
- 앞에서 구한 쿼리 결과와 traffic\_accident 테이블을 조인해 연도를 구함

## 2. 종합 실습 – 교통사고 분석 (07.traffic\_accident\_query.sql 참조)

(3) 교통수단별 가장 많은 사망자 수가 발생한 연도 구하기

```
SELECT a.*
FROM traffic_accident a
      ,( SELECT trans_type
          ,MAX(death_person_num) death_per
        FROM traffic_accident
        GROUP BY trans_type
      ) b
WHERE a.trans_type      = b.trans_type
      AND a.death_person_num = b.death_per;
```

	YEAR	TRANS_TYPE	TOTAL_ACCT_NUM	DEATH_PERSON_NUM
1	1991	자동차	265964	13429
2	1981	철도	1894	858
3	2003	지하철	103	256
4	1987	선박	533	477
5	1983	항공기	5	276

# 학습정리

- 조건절에서 자주 사용되는 컬럼을 인덱스로 만들면 조회 성능을 향상시킬 수 있다.
- 시노ним은 객체에 대한 또 다른 이름을 부여한 데이터베이스 객체이다.
- 시퀀스는 일련번호를 생성하는 데이터베이스 객체이다.