2023년 1학기 강의 ‘데이터베이스(03)’

과제1 REPORT

1971060 사이버보안전공

김서현

목 차

1. 데이터베이스 스키마 설명
2. 데이터베이스 생성
3. 데이터베이스 샘플 데이터
   1. Sample data for relation Constructors
   2. Sample data for relation Drivers
   3. Sample data for relation Races
   4. Sample data for relation Results
4. 문제 1번 쿼리 및 실행 결과
   1. Table Constructors 관련 쿼리 및 실행 결과
   2. Table Drivers 관련 쿼리 및 실행 결과
   3. Table Races 관련 쿼리 및 실행 결과
   4. Table Results 관련 쿼리 및 실행 결과
5. 문제 2번 쿼리 및 실행 결과
   1. Table Constructors 관련 쿼리 및 실행 결과
   2. Table Drivers 관련 쿼리 및 실행 결과
   3. Table Races 관련 쿼리 및 실행 결과
   4. Table Results 관련 쿼리 및 실행 결과
6. 문제 3번 쿼리 및 실행 결과
   1. 쿼리
   2. 실행 결과
7. 문제 4번 쿼리 및 실행 전후 비교
   1. a번 관련 쿼리 및 실행 결과 전후 비교
   2. b번 관련 쿼리 및 실행 결과 전후 비교
   3. c번 관련 쿼리 및 실행 결과 전후 비교
   4. d번 관련 쿼리 및 실행 결과 전후 비교
8. 문제 5번 쿼리 및 실행 결과
   1. a번 관련 쿼리 및 실행 결과
   2. b번 관련 쿼리 및 실행 결과
9. 데이터베이스 스키마 설명

다음의 데이터베이스 스키마는 FIA 포뮬러 원 월드 챔피언십 (FIA Formula One World Championship)에 관한 것이다. 포뮬라 원은 운전석 하나에 바퀴가 겉으로 드러난 오픈 휠 형식의 포뮬러 자동차 경주 중 가장 급이 높은 자동차 경주 대회이다. F-1 데이터베이스는 다음의 릴레이션을 포함한다. 각 속성(attribute)에 대한 적절한 데이터 타입을 사용한다.

* Constructors (constructor, engine, country, races\_entered, height, width)
  + constructor의 이름(name), 사용하는 엔진 종류(engine), constructor가 소속된 나라(country), 참가한 횟수(races\_entered), 차체의 크기(height, width)를 기록
* Drivers (name, birthday, country, constructor)
  + 드라이버의 이름(driver), 생일(birthday), 국적(country), 소속팀(constructor) 기록
  + birthday 속성(attribute)은 DATE 타입
* Races (name, beginDate, area)
  + 경기 그랑프리의 이름(name)과 개막날짜(beginDate)와 지역(area)을 저장
  + beginDate 속성(attribute)은 DATE 타입
* Results (race, driver, race\_rank)
  + 각 그랑프리의 선수들의 성적을 저장(각 선수는 같은 그랑프리에 한 번만 참가)
  + race\_rank 속성(attribute)은 각 선수가 참가한 그랑프리의 순위를 저장

1. 데이터베이스 생성 및 설정

본 과제에서 사용될 데이터베이스를 생성하기 위해 우선 MySQL에 접속합니다. 본인의 경우, ‘C:\Program Files\MySQL\MySQL Server 8.0\bin’ 디렉토리에서 작업을 진행했습니다. 해당 디렉토리를 CMD에서 접근한 다음 명령어 ‘mysql -u root -p’를 이용하여 MySQL 서버에 접속합니다. 그러면 이후 ‘mysql>’이라는 문구가 뜨고 해당 위치에 과제 수행을 위한 쿼리문을 작성합니다.

다음, 테이블들을 생성하기 위한 데이터베이스를 생성해야 합니다. 아래와 같은 명령어로 해당 작업을 수행할 수 있으며, 본인은 데이터베이스를 과제 안내에 따라 ‘db1971060’으로 설정했습니다. ‘USE’ 명령어로 현재 사용 중인 데이터베이스를 생성한 데이터베이스로 전환하면 과제 수행을 위한 모든 설정이 완료됩니다.

|  |
| --- |
| CREATE DATABASE db1971060;  USE db1971060; |

1. 데이터베이스 샘플 데이터
   1. Sample data for relation Constructors

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **constructor** | **country** | **engine** | **races\_entered** | **height** | **width** |
| McLaren | British | Mercedes | 884 | 95 | 180 |
| BMW Sauber | German | BMW | 70 | 95 | 180 |
| Renault | French | Renault | 660 | 91 | 175 |
| Ferrari | Italian | Ferrari | 1010 | 93 | 180 |
| Toyota | Japanese | Toyota | 1140 | 91 | 175 |
| Red Bull | Austrian | Honda | 304 | 91 | 175 |
| Honda | Japanese | Honda | 88 | 91 | 175 |
| Toro Rosso | Italian | Ferrari | 268 | 93 | 180 |
| Jaguar | British | Cosworth | 85 | 93 | 178 |

* 1. Sample data for relation Drivers

|  |  |  |  |
| --- | --- | --- | --- |
| **name** | **birthday** | **country** | **constructor** |
| hamilton | 1985-01-07 | British | McLaren |
| Heidfeld | 1977-05-10 | German | BMW Sauber |
| alonso | 1981-07-29 | Spanish | Renault |
| raikkonen | 1979-10-17 | Finnish | Ferrari |
| kubica | 1984-12-07 | Polish | BMW Sauber |
| massa | 1981-04-25 | Brazilian | Ferrari |
| trulli | 1974-07-13 | Italian | Toyota |
| webber | 1976-08-27 | Australian | Red Bull |
| button | 1980-01-19 | British | Honda |
| vettel | 1987-07-03 | German | Toro Rosso |
| barrichello | 1972-05-23 | Brazilian | Honda |
| Irvine | 1965-11-10 | British | Jaguar |
| Schmacher | 1969-01-03 | German | Ferrari |

* 1. Sample data for relation Races

|  |  |  |
| --- | --- | --- |
| **name** | **beginDate** | **area** |
| Australian Grand Prix | 2009-03-29 | Australia |
| British Grand Prix | 2009-06-21 | Europe |
| German Grand Prix | 2009-07-12 | Europe |
| European Grand Prix | 2009-08-23 | Europe |
| Abu Dhabi Grand Prix | 2009-11-01 | Middle East |
| Malaysian Grand Prix | 2008-03-23 | Asia |
| Spanish Grand Prix | 2008-04-27 | Europe |
| Singapore Grand Prix | 2008-09-28 | Asia |
| Brazilian Grand Prix | 2008-11-02 | South America |
| Monaco Grand Prix | 2007-05-27 | Africa |
| Canadian Grand Prix | 2007-06-10 | North America |
| United States Grand Prix | 2007-06-17 | North America |
| French Grand Prix | 2007-07-01 | Europe |
| Italian Grand Prix | 2007-09-09 | Europe |
| San Marino Grand Prix | 2004-04-25 | Europe |

* 1. Sample data for relation Results

|  |  |  |
| --- | --- | --- |
| **race** | **driver** | **race\_rank** |
| Australian Grand Prix | trulli | first place |
| Australian Grand Prix | barrichello | second place |
| Australian Grand Prix | button | third place |
| British Grand Prix | button | first place |
| British Grand Prix | barrichello | second place |
| British Grand Prix | vettel | third place |
| German Grand Prix | button | first place |
| German Grand Prix | vettel | second place |
| German Grand Prix | webber | third place |
| European Grand Prix | button | first place |
| European Grand Prix | barrichello | second place |
| European Grand Prix | webber | third place |
| Abu Dhabi Grand Prix | button | first place |
| Abu Dhabi Grand Prix | vettel | second place |
| Abu Dhabi Grand Prix | barrichello | third place |
| Malaysian Grand Prix | hamilton | first place |
| Malaysian Grand Prix | heidfeld | second place |
| Malaysian Grand Prix | raikkonen | third place |
| Spanish Grand Prix | hamilton | first place |
| Singapore Grand Prix | hamilton | second place |
| Singapore Grand Prix | massa | third place |
| Singapore Grand Prix | kubica | first place |
| Brazilian Grand Prix | hamilton | second place |
| Brazilian Grand Prix | massa | third place |
| Monaco Grand Prix | alonso | first place |
| Monaco Grand Prix | hamilton | second place |
| Monaco Grand Prix | massa | third place |
| Canadian Grand Prix | hamilton | first place |
| Canadian Grand Prix | alonso | second place |
| Canadian Grand Prix | massa | third place |
| United States Grand Prix | hamilton | first place |
| United States Grand Prix | alonso | second place |
| United States Grand Prix | massa | third place |
| French Grand Prix | hamilton | first place |
| French Grand Prix | alonso | second place |
| French Grand Prix | massa | third place |
| Italian Grand Prix | hamilton | first place |
| Italian Grand Prix | alonso | second place |
| San Marino Grand Prix | Schmacher | first place |
| San Marino Grand Prix | barrichello | second place |

1. 문제 1번 쿼리 및 실행 결과

과제에서 제시된 악식 스키마와 샘플 데이터를 기반으로 네 객의 테이블 만드는 선언(table creation declaration)을 각각 SQL로 작성했습니다. 이후, 생성된 테이블 결과를 보여주기 위해 다음의 두 명령어를 실행하여 생성 결과를 확인했습니다.

|  |
| --- |
| DESCRIBE [*table name*];  SELECT \* FROM INFORMATION\_SCHEMA.TABLE\_CONSTRAINTS  WHERE CONSTRAINT\_SCHEMA = '[*DB name*]' and TABLE\_NAME = '[*table name*]'; |

1. Table Constructors 관련 쿼리 및 실행 결과

* 작성한 테이블 생성 SQL 쿼리문

|  |
| --- |
| CREATE TABLE Constructors (  constructor VARCHAR(50) PRIMARY KEY,  engine VARCHAR(50),  country VARCHAR(50),  races\_entered INT,  height INT,  width INT  ); |

* 실행 결과

텍스트, 스크린샷, 모니터, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

1. Table Drivers 관련 쿼리 및 실행 결과

* 작성한 테이블 생성 SQL 쿼리문

|  |
| --- |
| CREATE TABLE Drivers (  name VARCHAR(50) PRIMARY KEY,  birthday DATE,  country VARCHAR(50),  constructor VARCHAR(50),  FOREIGN KEY (constructor) REFERENCES Constructors(constructor)  ); |

* 실행 결과

텍스트이(가) 표시된 사진

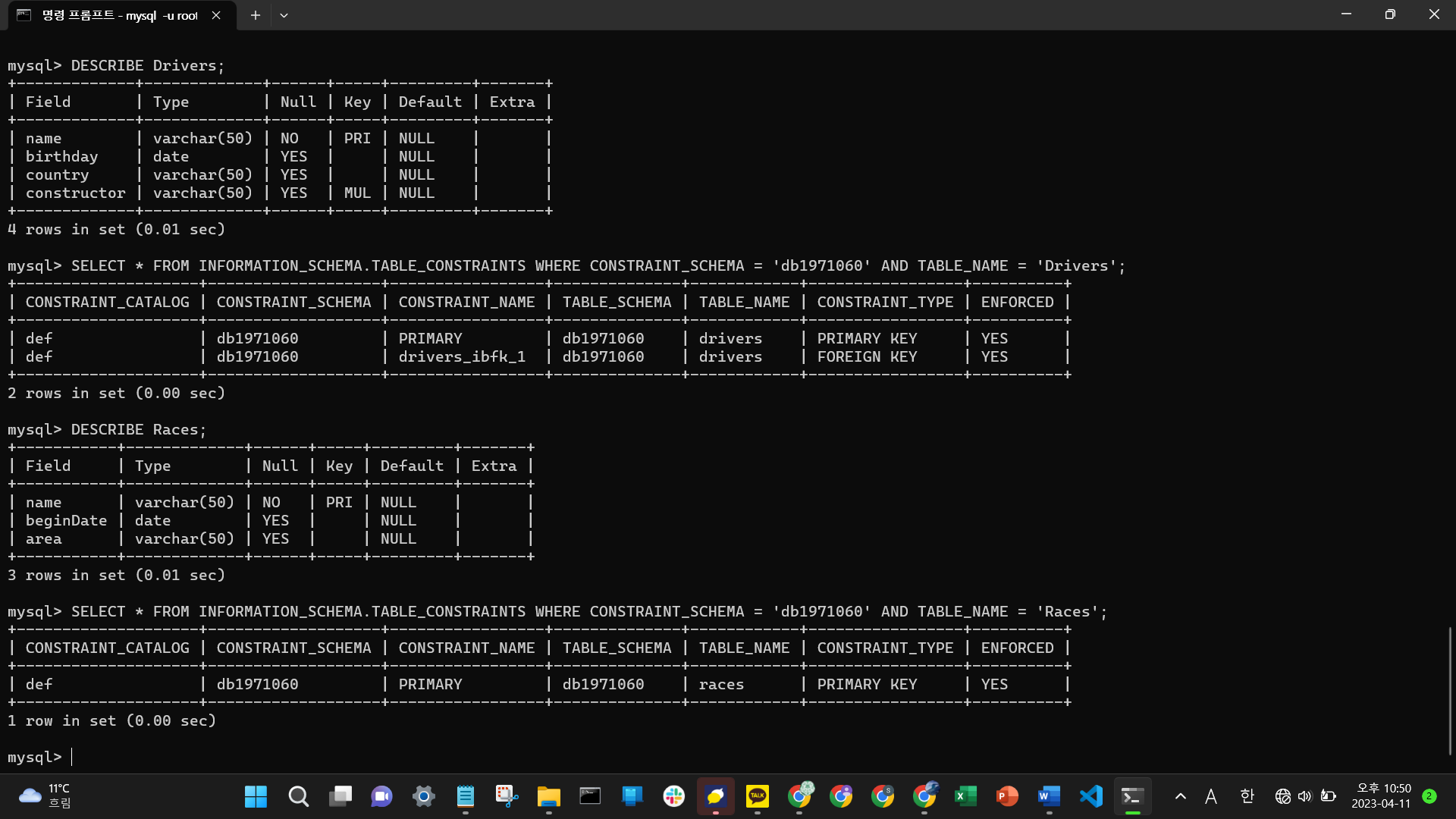
자동 생성된 설명

1. Table Races 관련 쿼리 및 실행 결과

* 작성한 테이블 생성 SQL 쿼리문

|  |
| --- |
| CREATE TABLE Races (  name VARCHAR(50) PRIMARY KEY,  beginDate DATE,  area VARCHAR(50)  ); |

* 실행 결과



1. Table Results 관련 쿼리 및 실행 결과

* 작성한 테이블 생성 SQL 쿼리문

|  |
| --- |
| CREATE TABLE Results (  race VARCHAR(50),  driver VARCHAR(50),  race\_rank VARCHAR(255),  PRIMARY KEY (race, driver),  FOREIGN KEY (race) REFERENCES Races(name),  FOREIGN KEY (driver) REFERENCES Drivers(name)  ); |

* 실행 결과

텍스트이(가) 표시된 사진

자동 생성된 설명

1. 문제 2번 쿼리 및 실행 결과

위의 샘플들에서 보여준 컨텐츠(contents)를 각 릴레이션의 컨텐츠로 생성하기 위한 각각의 insert SQL 쿼리를 작성했습니다. 이때, 외래키(foreign key) 관계를 바탕으로 Constructors, Drivers, Races, Results 테이블 순서로 데이터를 삽입했습니다. Results 테이블이 Races과 Drivers 테이블을 참조하는 외래키가, Drivers 테이블이 Constructor 테이블을 참조하는 외래키가 있기 때문입니다. 또한, 아래의 명령어를 실행하여 생성 결과를 확인했습니다.

|  |
| --- |
| SELECT \* FROM [*table name*]; |

1. Table Constructors 관련 쿼리 및 실행 결과
   * 작성한 컨텐츠 생성 SQL 쿼리

|  |
| --- |
| INSERT INTO Constructors (constructor, country, engine, races\_entered, height, width) VALUES  ('McLaren', 'British', 'Mercedes', '884', '95', '180'),  ('BMW Sauber', 'German', 'BMW', '70', '95', '180'),  ('Renault', 'French', 'Renault', '660', '91', '175'),  ('Ferrari', 'Italian', 'Ferrari', '1010', '93', '180'),  ('Toyota', 'Japanese', 'Toyota', '1140', '91', '175’),  ('Red Bull', 'Austrian', 'Honda', '304', '91', '175'),  ('Honda', 'Japanese', 'Honda', '88', '91', '175'),  ('Toro Rosso', 'Italian', 'Ferrari', '268', '93', '180'),  ('Jaguar', 'British', 'Cosworth', '85', '93', '178'); |

* + 실행 결과

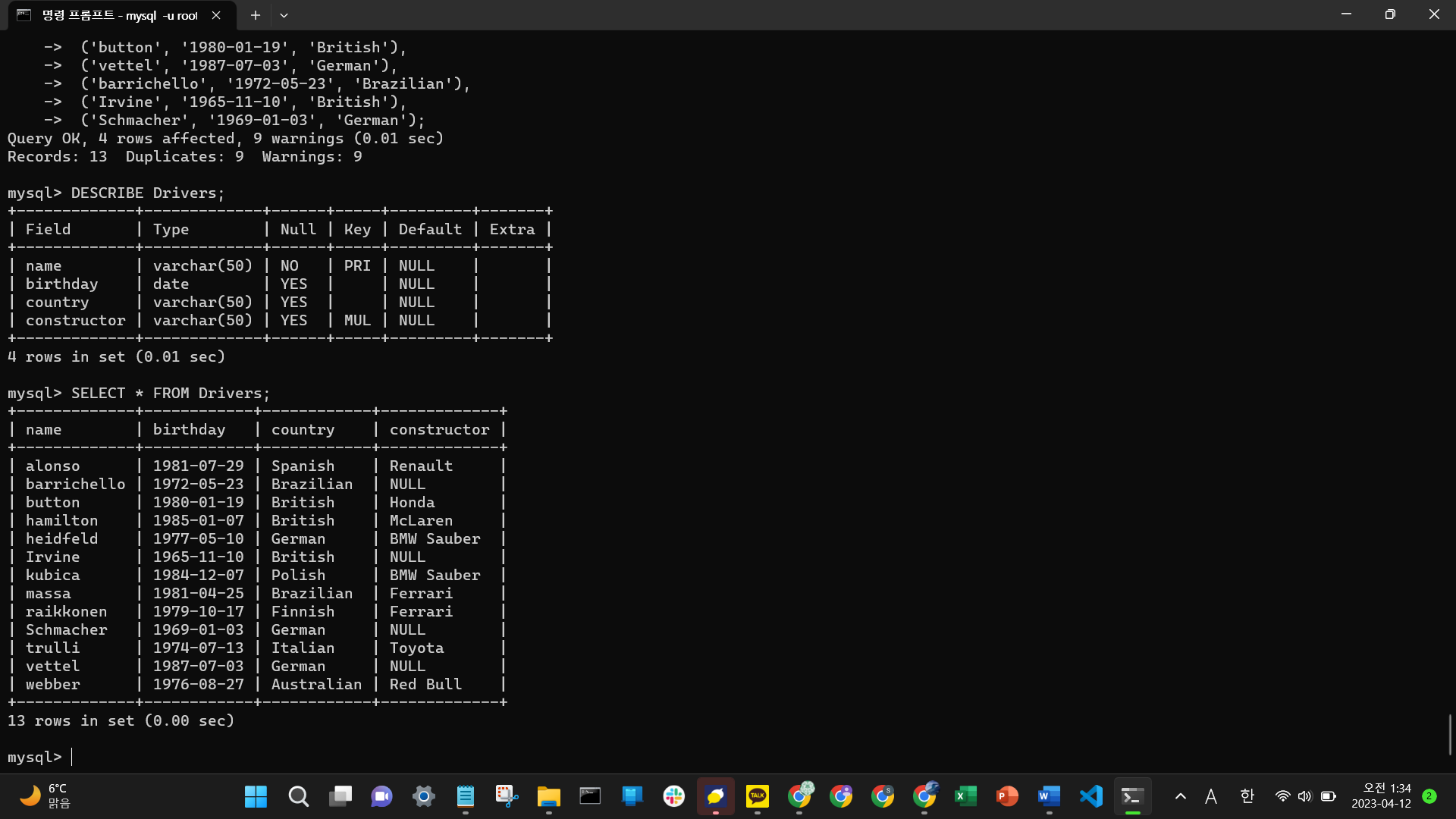
텍스트, 스크린샷, 전자제품, 컴퓨터이(가) 표시된 사진

자동 생성된 설명

1. Table Drivers 관련 쿼리 및 실행 결과
   * 작성한 컨텐츠 생성 SQL 쿼리

|  |
| --- |
| INSERT INTO Drivers (name, birthday, country, constructor) VALUES  ('hamilton', '1985-01-07', 'British', 'McLaren'),  ('heidfeld', '1977-05-10', 'German', 'BMW Sauber'),  ('alonso', '1981-07-29', 'Spanish', 'Renault'),  ('raikkonen', '1979-10-17', 'Finnish', 'Ferrari'),  ('kubica', '1984-12-07', 'Polish', 'BMW Sauber'),  ('massa', '1981-04-25', 'Brazilian', 'Ferrari'),  ('trulli', '1974-07-13', 'Italian', 'Toyota'),  ('webber', '1976-08-27', 'Australian', 'Red Bull'),  ('button', '1980-01-19', 'British', 'Honda'),  ('vettel', '1987-07-03', 'German'),  ('barrichello', '1972-05-23', 'Brazilian'),  ('Irvine', '1965-11-10', 'British'),  ('Schmacher', '1969-01-03', 'German'); |

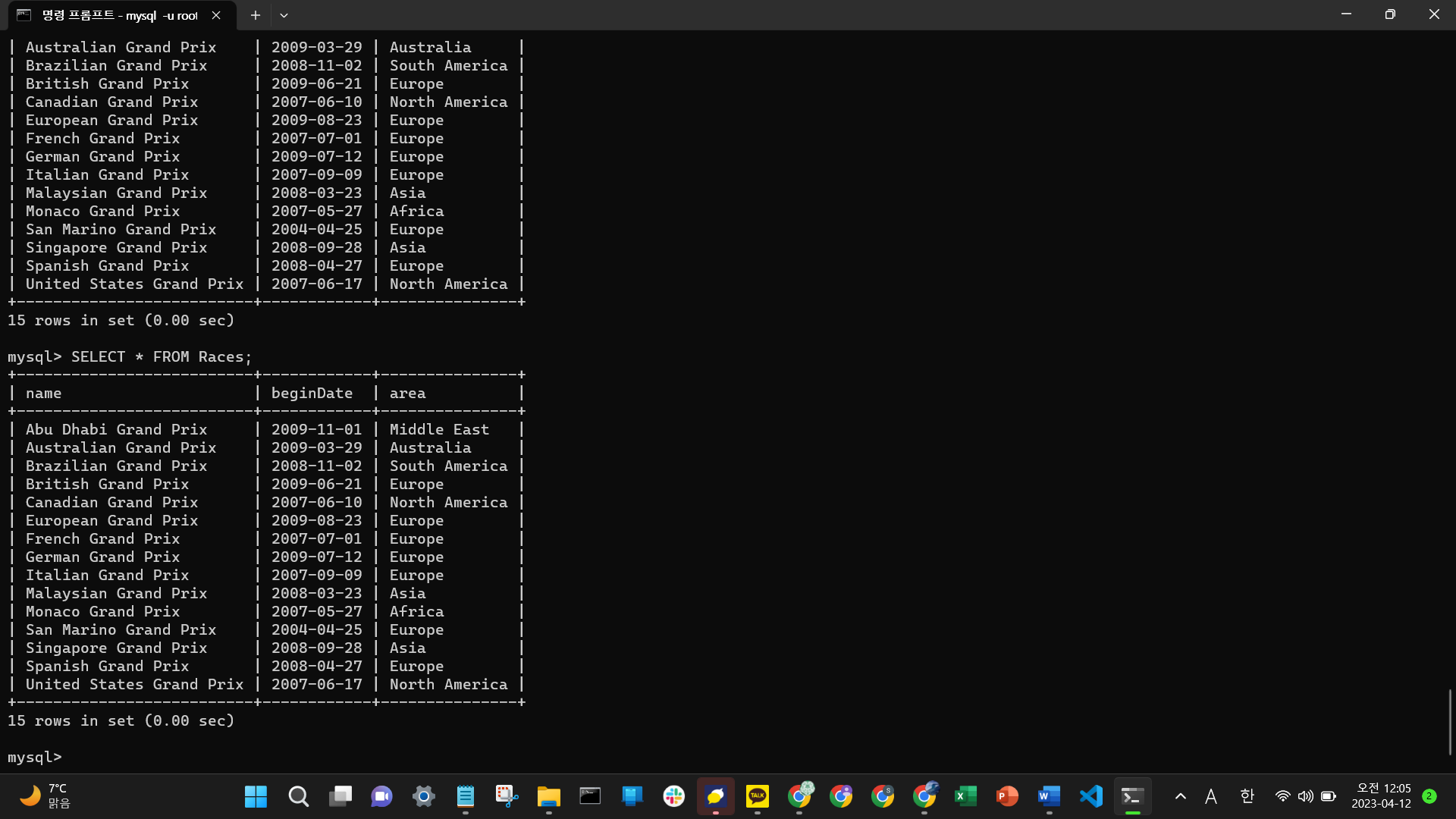
* + 실행 결과



1. Table Races 관련 쿼리 및 실행 결과
   * 작성한 컨텐츠 생성 SQL 쿼리

|  |
| --- |
| INSERT INTO Races (name, beginDate, area) VALUES  ('Australian Grand Prix', '2009-03-29', 'Australia'),  ('British Grand Prix', '2009-06-21', 'Europe'),  ('German Grand Prix', '2009-07-12', 'Europe'),  ('European Grand Prix', '2009-08-23', 'Europe'),  ('Abu Dhabi Grand Prix', '2009-11-01', 'Middle East'),  ('Malaysian Grand Prix', '2008-03-23', 'Asia'),  ('Spanish Grand Prix', '2008-04-27', 'Europe'),  ('Singapore Grand Prix', '2008-09-28', 'Asia'),  ('Brazilian Grand Prix', '2008-11-02', 'South America'),  ('Monaco Grand Prix', '2007-05-27', 'Africa'),  ('Canadian Grand Prix', '2007-06-10', 'North America'),  ('United States Grand Prix', '2007-06-17', 'North America'),  ('French Grand Prix', '2007-07-01', 'Europe'),  ('Italian Grand Prix', '2007-09-09', 'Europe'),  ('San Marino Grand Prix', '2004-04-25', 'Europe'); |

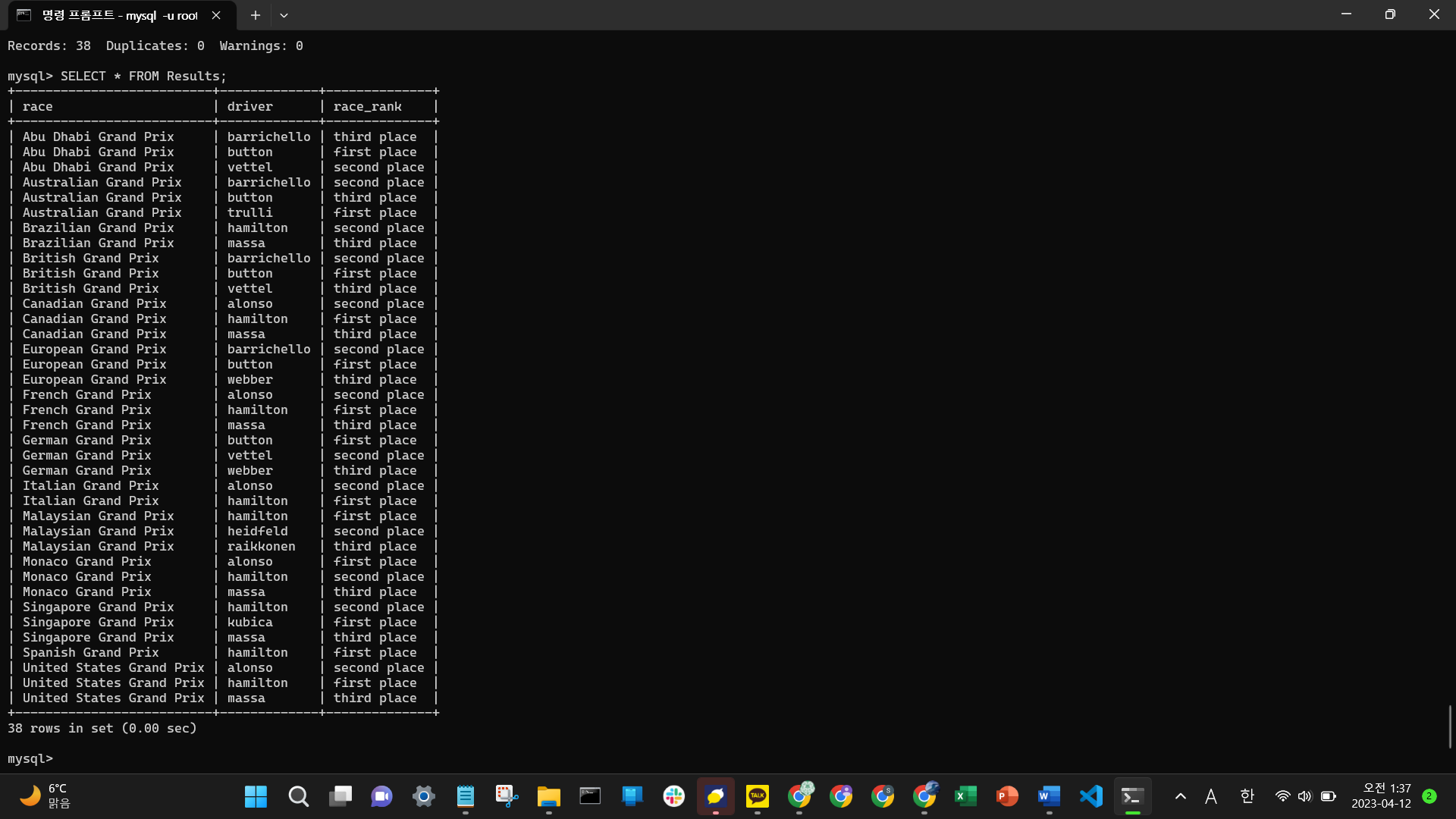
* + 실행 결과

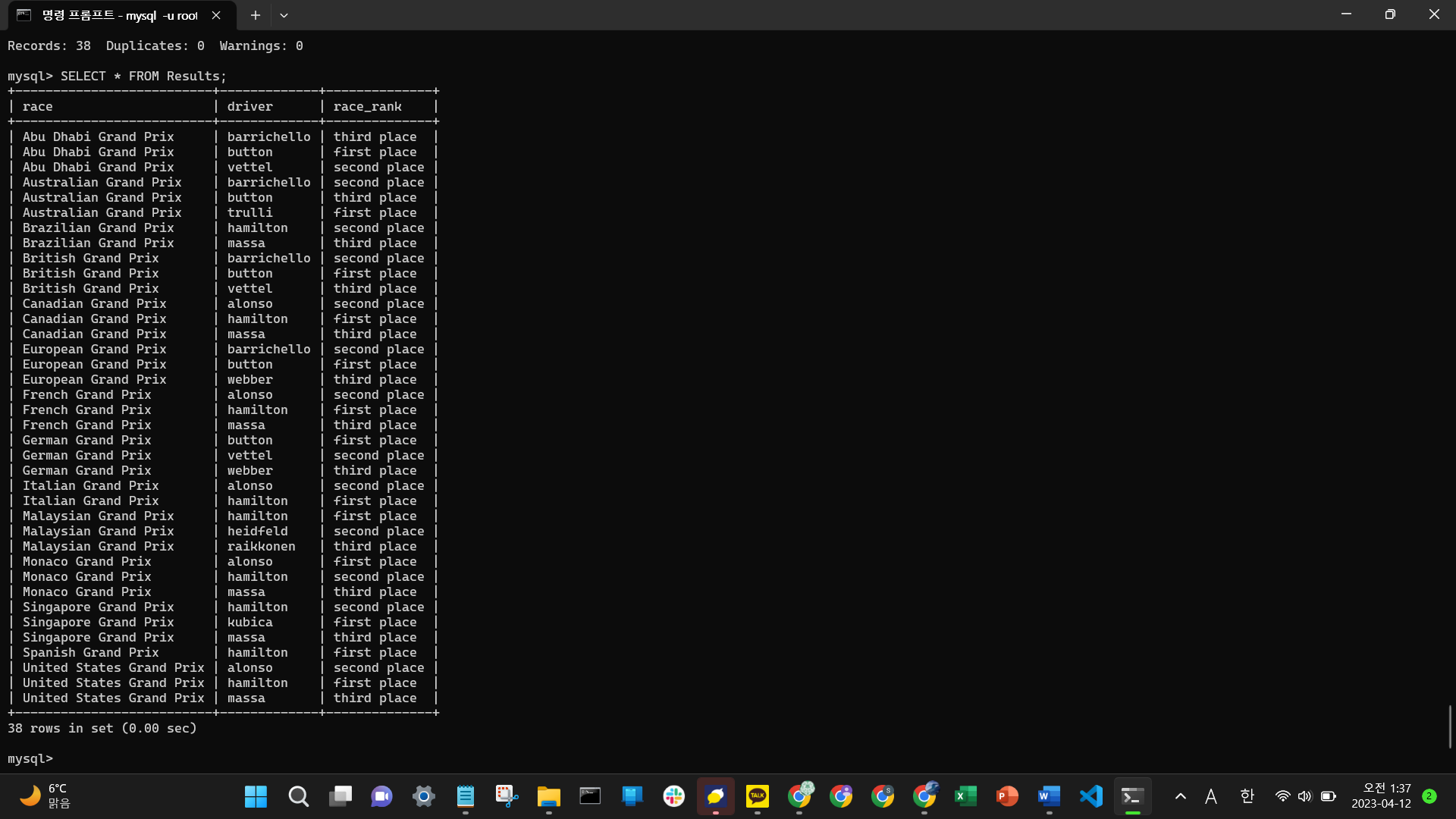


1. Table Results 관련 쿼리 및 실행 결과
   * 작성한 컨텐츠 생성 SQL 쿼리

|  |
| --- |
| INSERT INTO Results (race, driver, race\_rank) VALUES  ('Australian Grand Prix', 'trulli', 'first place'),  ('Australian Grand Prix', 'barrichello', 'second place'),  ('Australian Grand Prix', 'button', 'third place'),  ('British Grand Prix', 'button', 'first place'),  ('British Grand Prix', 'barrichello', 'second place'),  ('British Grand Prix', 'vettel', 'third place'),  ('German Grand Prix', 'button', 'first place'),  ('German Grand Prix', 'vettel', 'second place'),  ('German Grand Prix', 'webber', 'third place'),  ('European Grand Prix', 'button', 'first place'),  ('European Grand Prix', 'barrichello', 'second place'),  ('European Grand Prix', 'webber', 'third place'),  ('Abu Dhabi Grand Prix', 'button', 'first place'),  ('Abu Dhabi Grand Prix', 'vettel', 'second place'),  ('Abu Dhabi Grand Prix', 'barrichello', 'third place'),  ('Malaysian Grand Prix', 'hamilton', 'first place'),  ('Malaysian Grand Prix', 'heidfeld', 'second place'),  ('Malaysian Grand Prix', 'raikkonen', 'third place'),  ('Spanish Grand Prix', 'hamilton', 'first place'),  ('Singapore Grand Prix', 'hamilton', 'second place'),  ('Singapore Grand Prix', 'massa', 'third place'),  ('Singapore Grand Prix', 'kubica', 'first place'),  ('Brazilian Grand Prix', 'hamilton', 'second place'),  ('Brazilian Grand Prix', 'massa', 'third place'),  ('Monaco Grand Prix', 'alonso', 'first place'),  ('Monaco Grand Prix', 'hamilton', 'second place'),  ('Monaco Grand Prix', 'massa', 'third place'),  ('Canadian Grand Prix', 'hamilton', 'first place'),  ('Canadian Grand Prix', 'alonso', 'second place'),  ('Canadian Grand Prix', 'massa', 'third place'),  ('United States Grand Prix', 'hamilton', 'first place'),  ('United States Grand Prix', 'alonso', 'second place'),  ('United States Grand Prix', 'massa', 'third place'),  ('French Grand Prix', 'hamilton', 'first place'),  ('French Grand Prix', 'alonso', 'second place'),  ('French Grand Prix', 'massa', 'third place'),  ('Italian Grand Prix', 'hamilton', 'first place'),  ('Italian Grand Prix', 'alonso', 'second place'); |

* + 실행 결과





1. 문제 3번 쿼리 및 실행 결과

데이터베이스 스키마를 기반으로 하나의 relation에 관한 다음의 간단한 SQL 쿼리(simple SQL query)를 작성했습니다. a부터 h번까지 각각의 쿼리 및 실행 결과는 다음과 같습니다.

* 1. 쿼리

1. 그랑프리에 100번 이상 참가한 constructor의 이름과 constructor의 국가(country)

|  |
| --- |
| SELECT constructor, country  FROM Constructors  WHERE races\_entered > 100; |

1. 차체의 높이가 95센치가 아니고 차체의 너비가 180 센티보다 작은 차체를 사용하는 레이싱 팀의 이름, 엔진이름, 그랑프리 참가 횟수(결과는 첫번째는 엔진 이름으로, 두 번째는 그랑프리에 참가한 횟수를 기준으로 오름차순으로)

|  |
| --- |
| SELECT constructor, engine, races\_entered  FROM Constructors  WHERE height != 95 AND width < 180  ORDER BY engine ASC, races\_entered ASC; |

1. Ferrari 엔진을 사용하는 모든 constructor

|  |
| --- |
| SELECT \*  FROM Constructors  WHERE engine = 'Ferrari'; |

1. 1980년도 이후에 태어난 모든 드라이버의 이름(단, 결과 테이블의 컬럼은 새 이름으로 newBoys을 사용)

|  |
| --- |
| SELECT names AS newBoys  FROM Drivers  WHERE YEAR(birthday) > 1980; |

1. 각 그랑프리에서 우승한 드라이버의 이름과 그랑프리의 이름

|  |
| --- |
| SELECT race, driver  FROM Results  WHERE race\_rank = 'first place'; |

1. 유럽 지역에서 열린 모든 그랑프리(그랑프리 이름을 기준으로 오름차순 정렬)

|  |
| --- |
| SELECT name  FROM Races  WHERE area = 'Europe'  ORDER BY name ASC; |

1. 문자 ‘H’로 시작하는 이름을 가진 모든 드라이버

|  |
| --- |
| SELECT name  FROM Drivers  WHERE name LIKE 'H%'; |

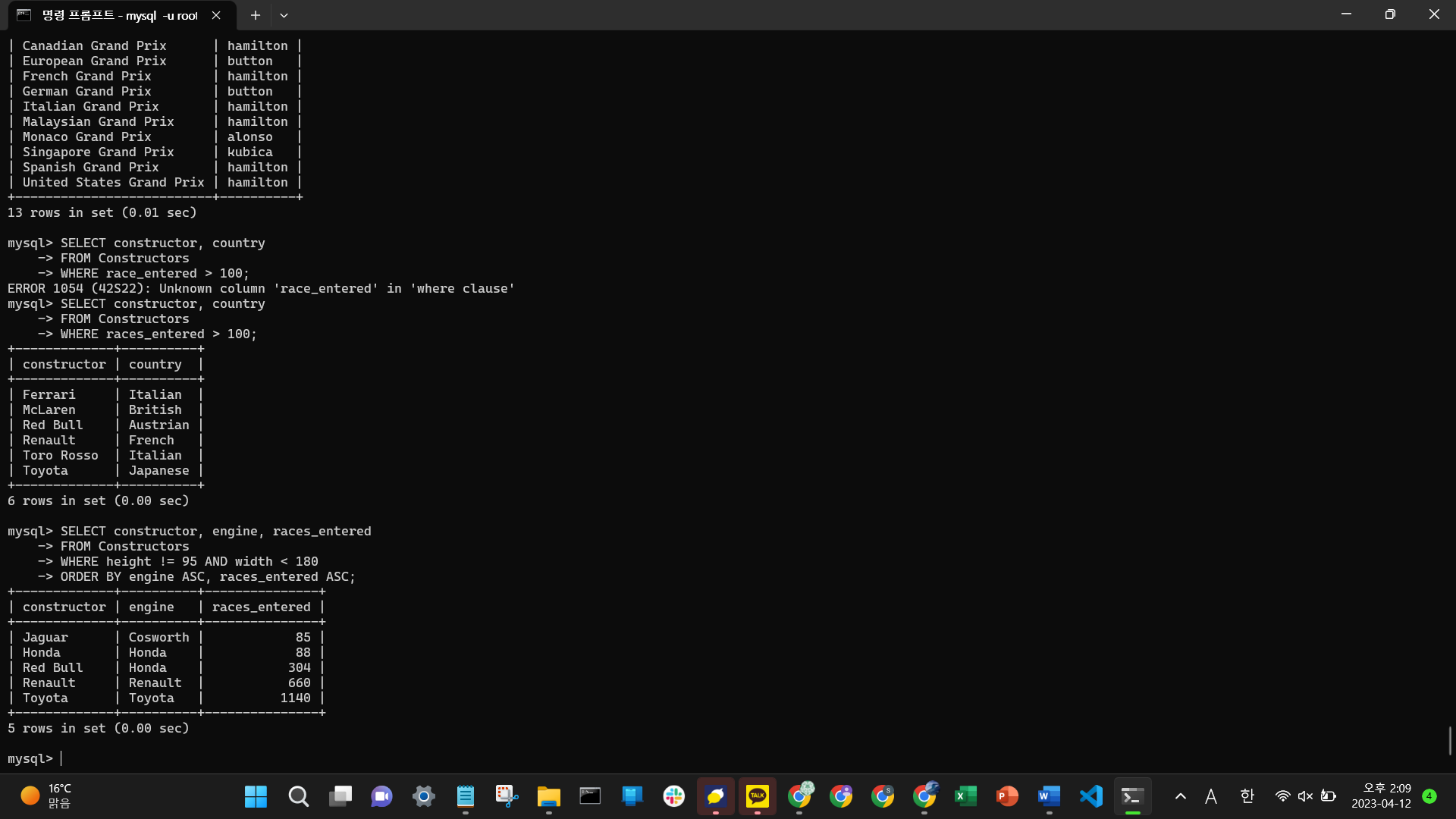
1. 두 개 이상의 단어로 이루어진 이름을 가지는 모든 constructor들의 이름

|  |
| --- |
| SELECT constructor  FROM Constructors  WHERE constructor LIKE '% %'; |

* 1. 실행 결과
  2. 그랑프리에 100번 이상 참가한 constructor의 이름과 constructor의 국가(country)



* 1. 차체의 높이가 95센치가 아니고 차체의 너비가 180 센티보다 작은 차체를 사용하는 레이싱 팀의 이름, 엔진이름, 그랑프리 참가 횟수(결과는 첫번째는 엔진 이름으로, 두 번째는 그랑프리에 참가한 횟수를 기준으로 오름차순으로)

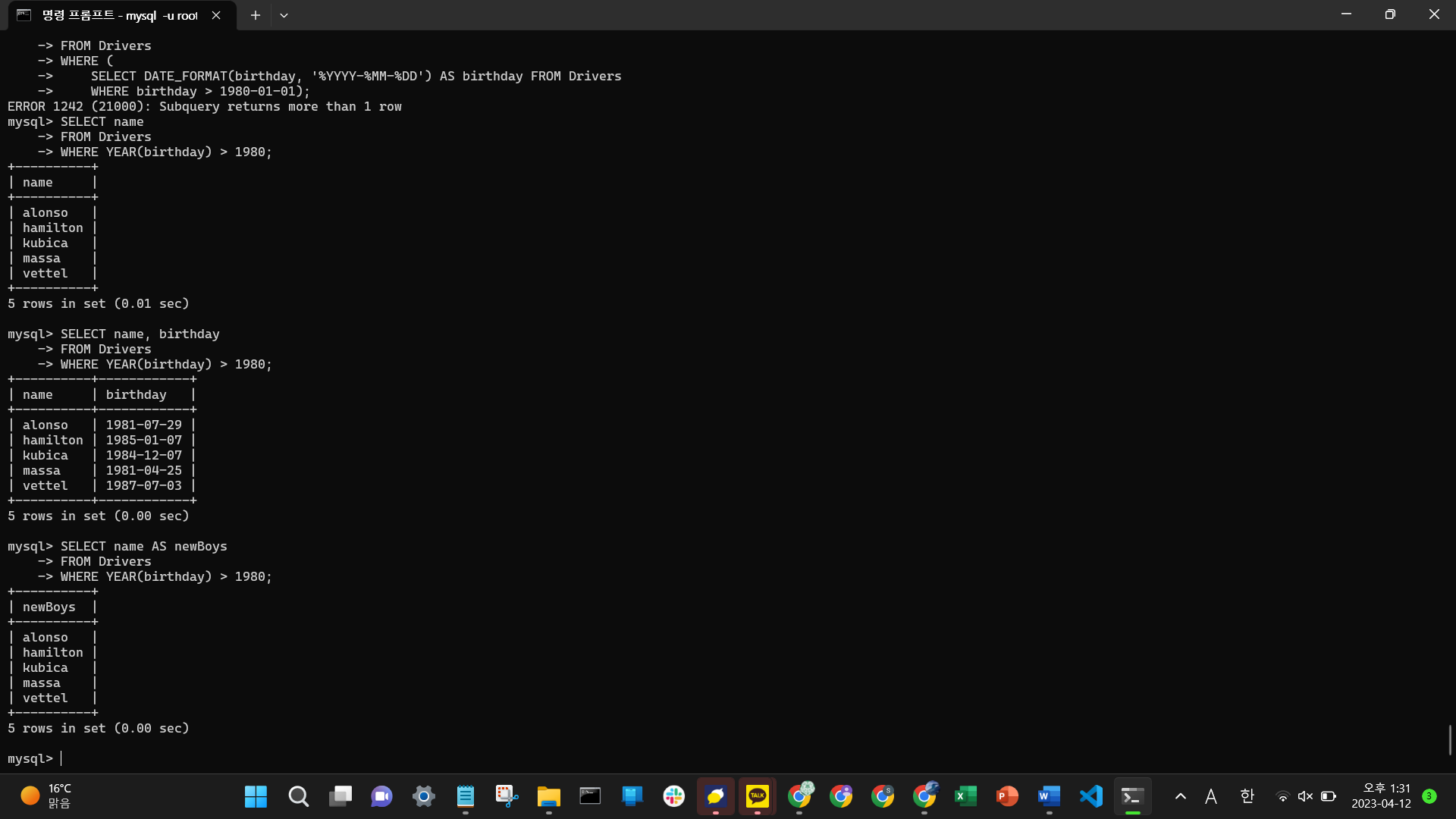


* 1. Ferrari 엔진을 사용하는 모든 constructor

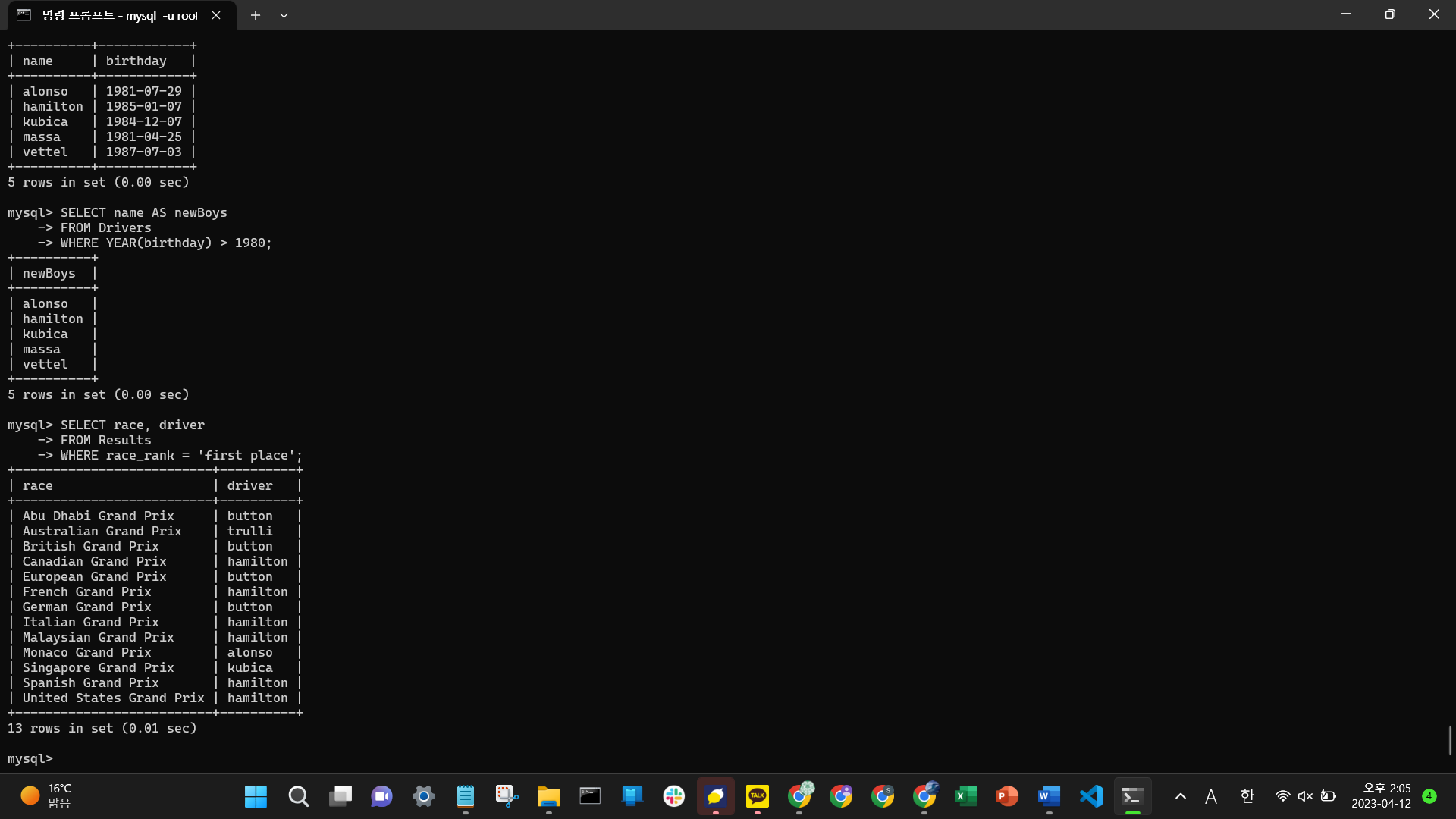
텍스트, 전자제품, 스크린샷, 디스플레이이(가) 표시된 사진

자동 생성된 설명

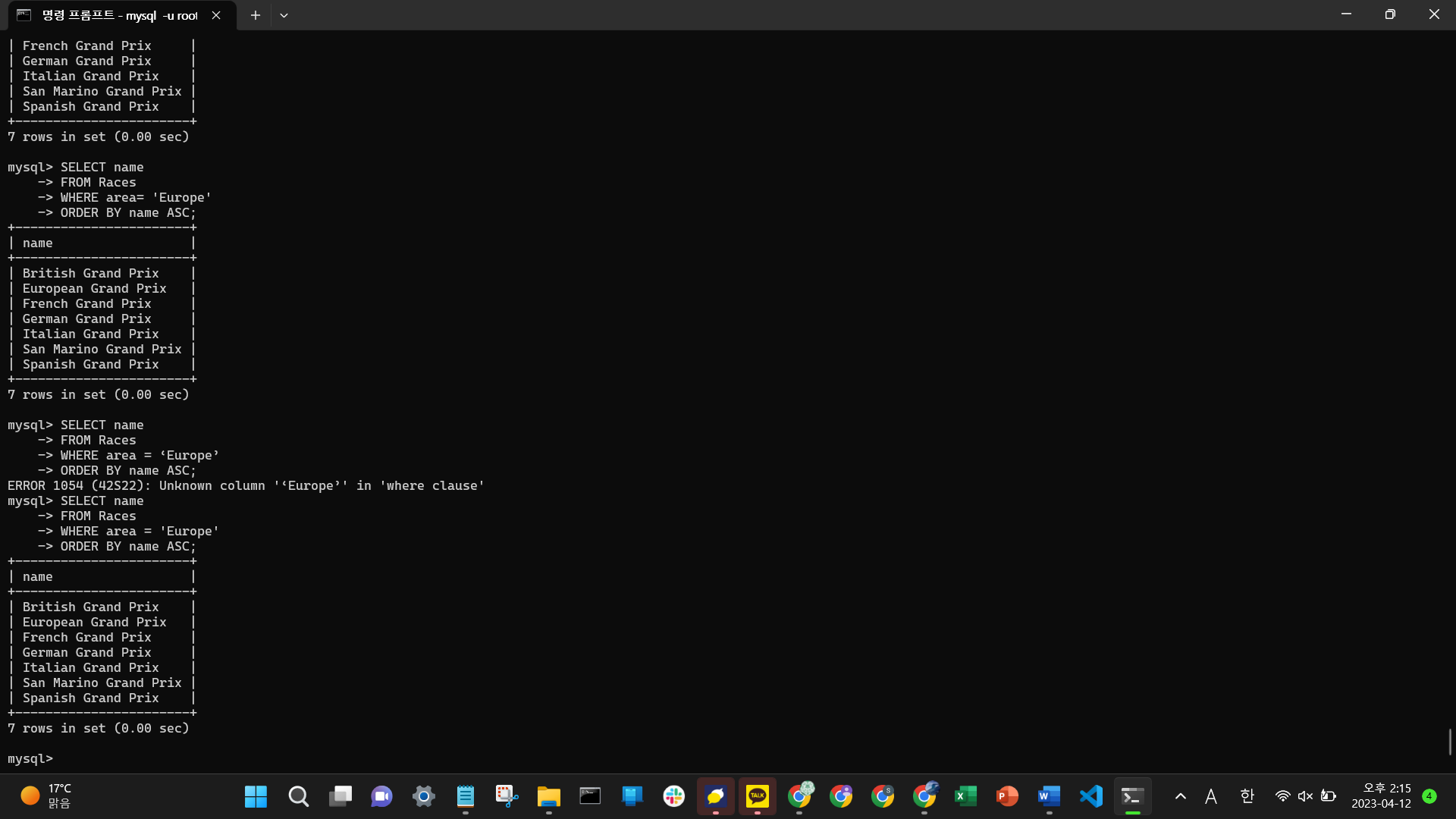
* 1. 1980년도 이후에 태어난 모든 드라이버의 이름(단, 결과 테이블의 컬럼은 새 이름으로 newBoys을 사용)



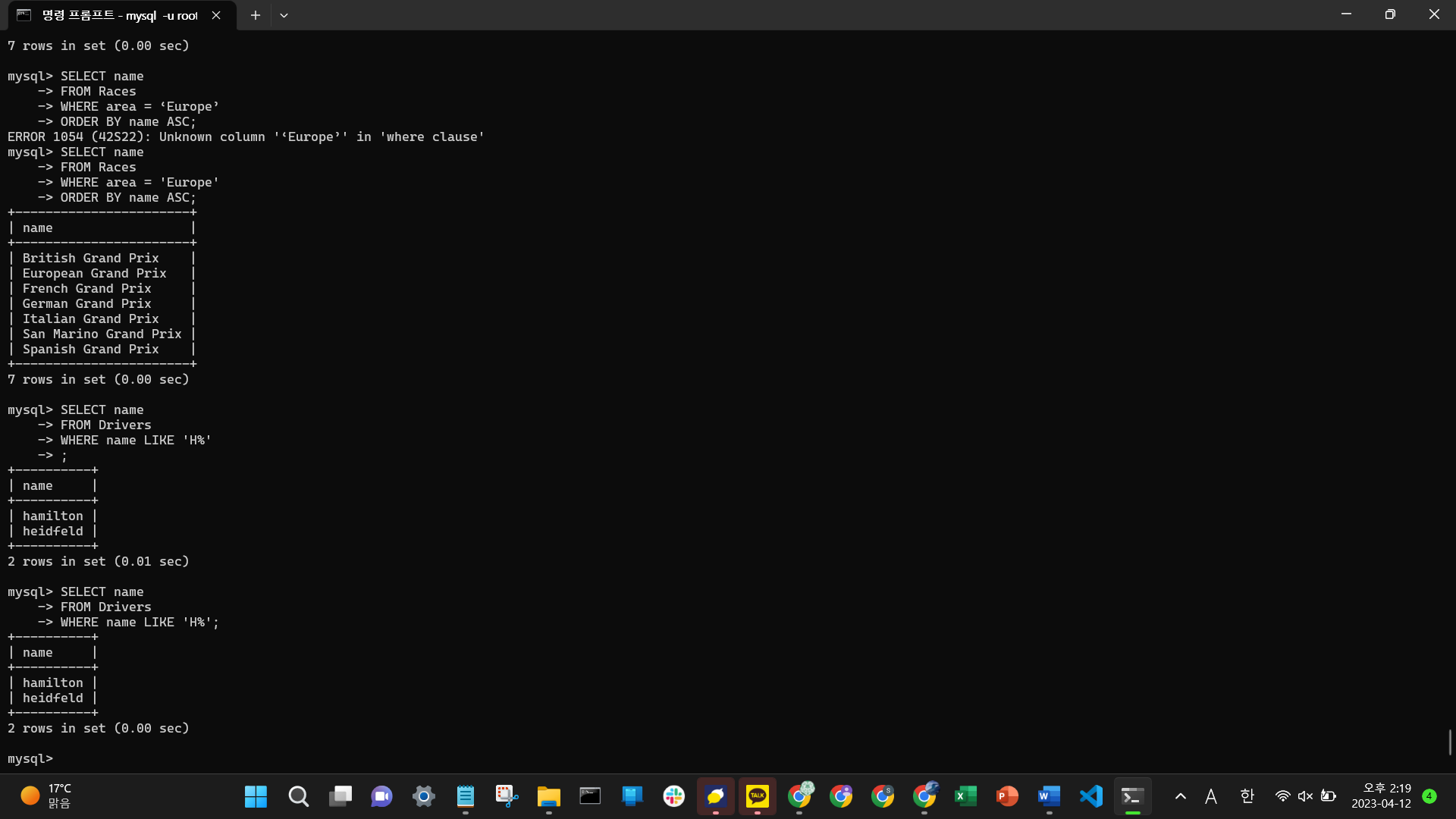
* 1. 각 그랑프리에서 우승한 드라이버의 이름과 그랑프리의 이름



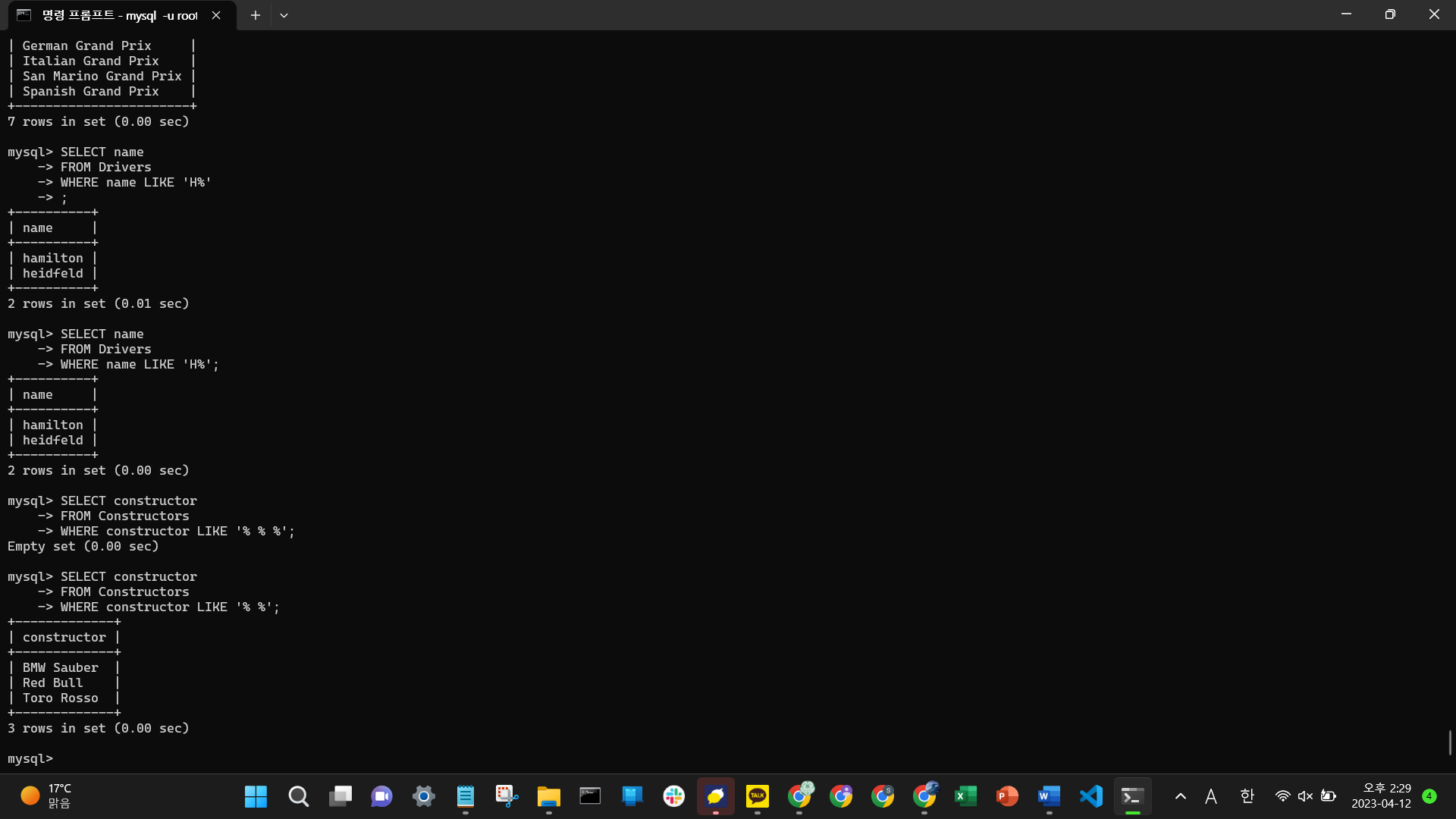
* 1. 유럽 지역에서 열린 모든 그랑프리(그랑프리 이름을 기준으로 오름차순 정렬)



* 1. 문자 ‘H’로 시작하는 이름을 가진 모든 드라이버



* 1. 두 개 이상의 단어로 이루어진 이름을 가지는 모든 constructor들의 이름



1. 문제 4번 쿼리 및 실행 결과

데이터베이스 스키마를 기반으로 수정(modification)을 포함하는 SQL쿼리를 작성했습니다. 스크립트에는 릴레이션 수행 전후의 결과를 모두 보여주기 위해 ‘select \*’를 실행했습니다. 각 문제에 대한 작성된 쿼리 및 실행 결과는 다음과 같습니다.

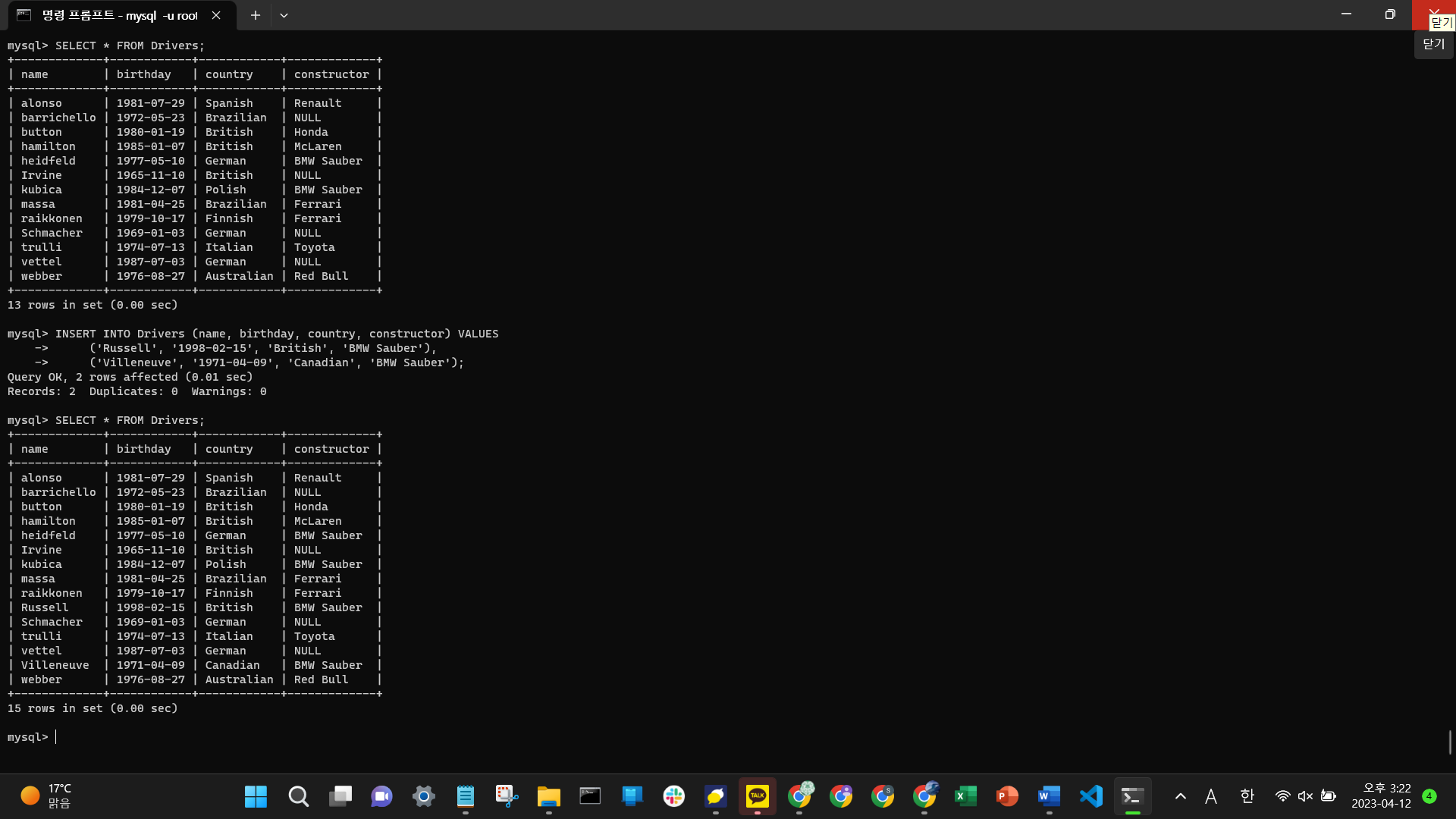
* 1. a번 관련 쿼리 및 실행 결과 전후 비교

BMW Sauber 레이싱 팀에 두 명의 드라이버 Russell와 Villeneuve를 영입하였다. 영국 출신(British)의 Russell의 생년월일은 15/02/1998이다. 캐나다 출신(Canadian) Villeneuve의 생년월일은 09/04/1971이다. 이러한 사실을 데이터베이스에 삽입하라.

* + 작성한 컨텐츠 수정 SQL 쿼리

|  |
| --- |
| SELECT \* FROM Drivers;  -- Insert new Drivers ‘Russell’ And ‘Villeneuve’  INSERT INTO Drivers (name, birthday, country, constructor) VALUES  ('Russell', '1998-02-15', 'British', 'BMW Sauber'),  ('Villeneuve', '1971-04-09', 'Canadian', 'BMW Sauber');  -- View Changes  SELECT \* FROM Drivers; |

* + 실행 결과



* 1. b번 관련 쿼리 및 실행 결과 전후 비교

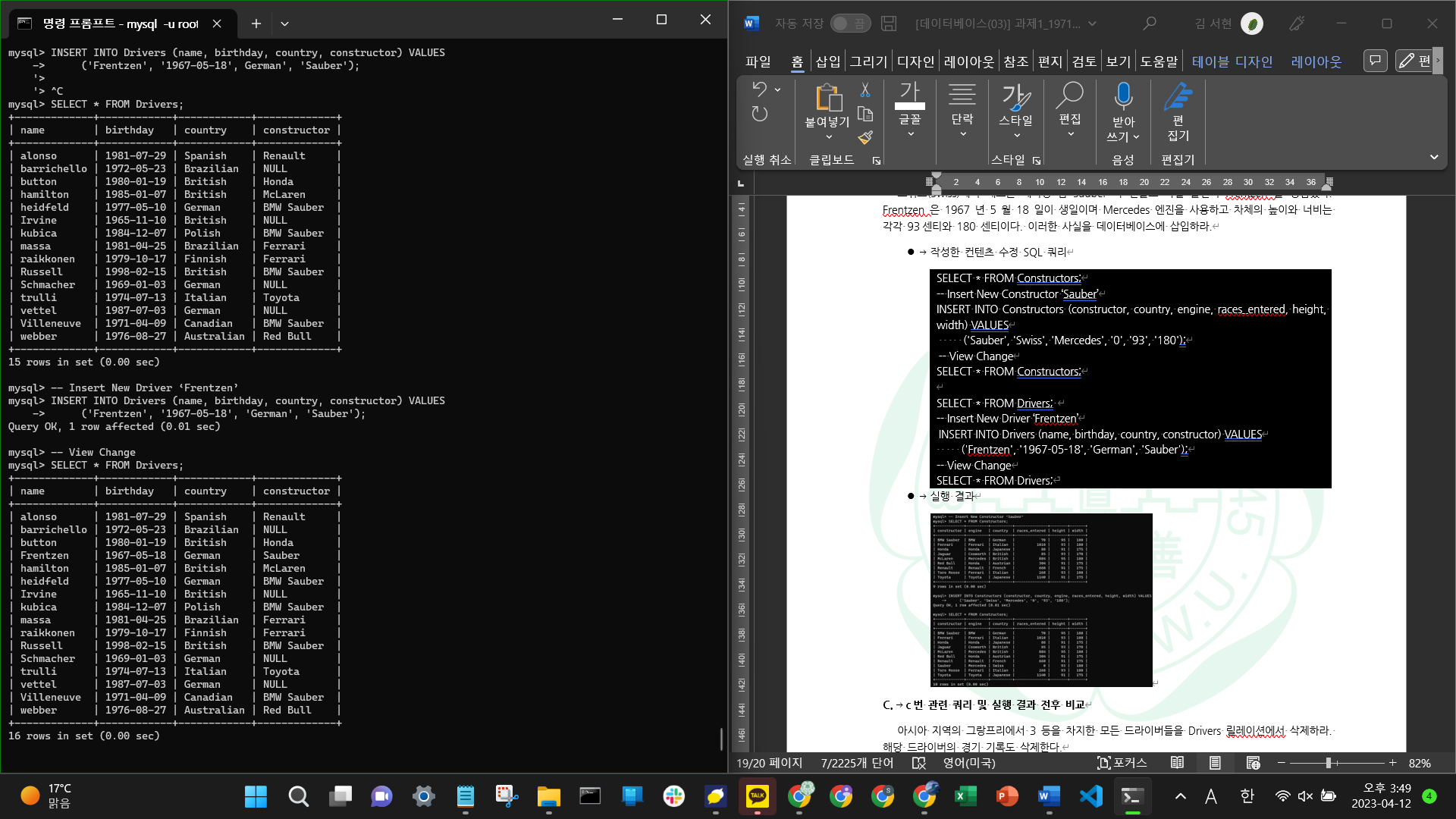
스위스(Swiss)에서 새로운 레이싱 팀 Sauber가 만들고 독일 출신의 Frentzen를 영입했다. Frentzen은 1967년 5월 18일이 생일이며 Mercedes 엔진을 사용하고 차체의 높이와 너비는 각각 93센티와 180 센티이다. 이러한 사실을 데이터베이스에 삽입하라.

* + 작성한 컨텐츠 수정 SQL 쿼리

|  |
| --- |
| SELECT \* FROM Constructors;  -- Insert New Constructor ‘Sauber’  INSERT INTO Constructors (constructor, country, engine, races\_entered, height, width) VALUES  ('Sauber', 'Swiss', 'Mercedes', '0', '93', '180');  -- View Change  SELECT \* FROM Constructors;  SELECT \* FROM Drivers;  -- Insert New Driver ‘Frentzen’  INSERT INTO Drivers (name, birthday, country, constructor) VALUES  ('Frentzen', '1967-05-18', 'German', 'Sauber');  -- View Change  SELECT \* FROM Drivers; |

* + 실행 결과





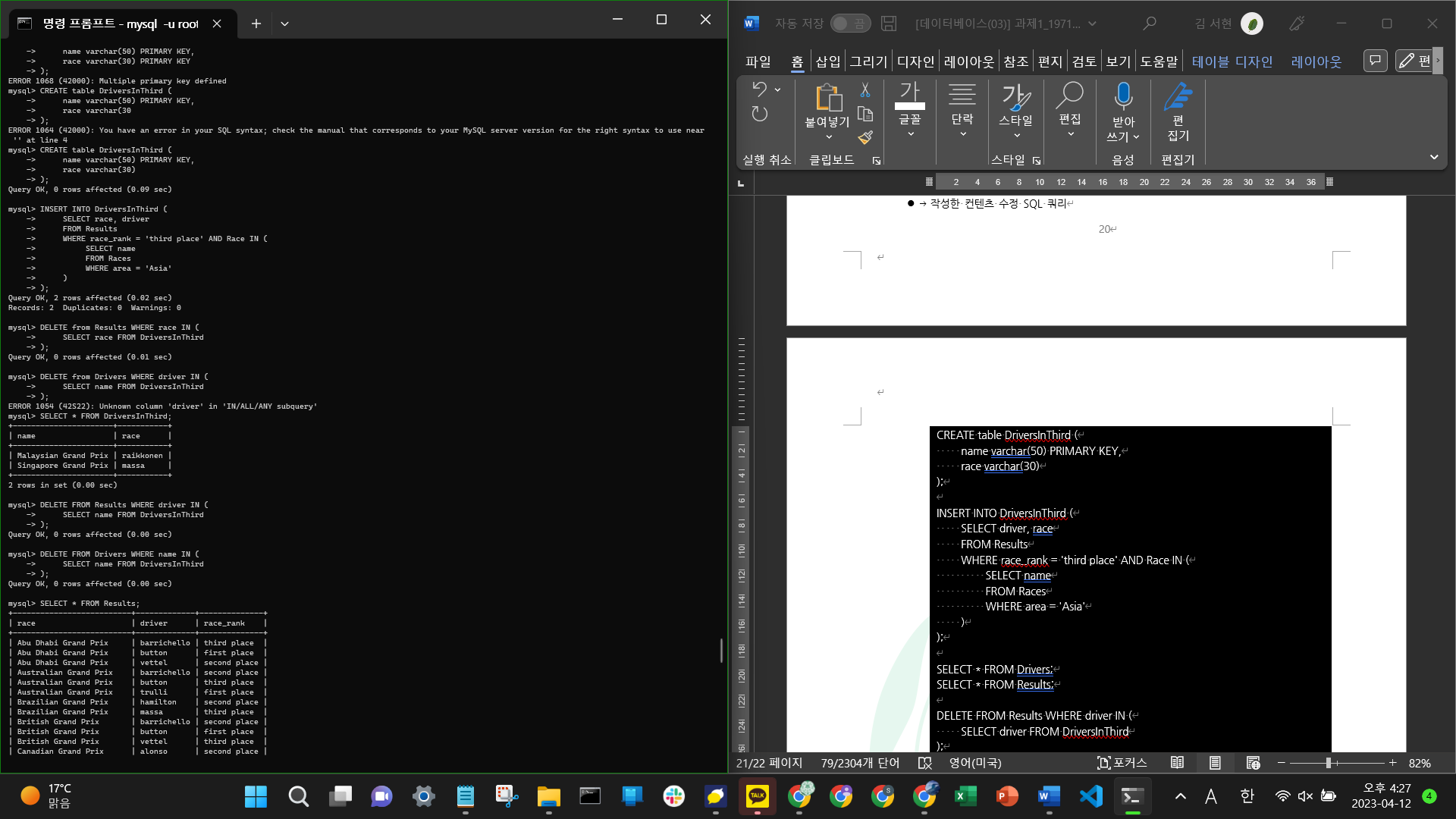
* 1. c번 관련 쿼리 및 실행 결과 전후 비교

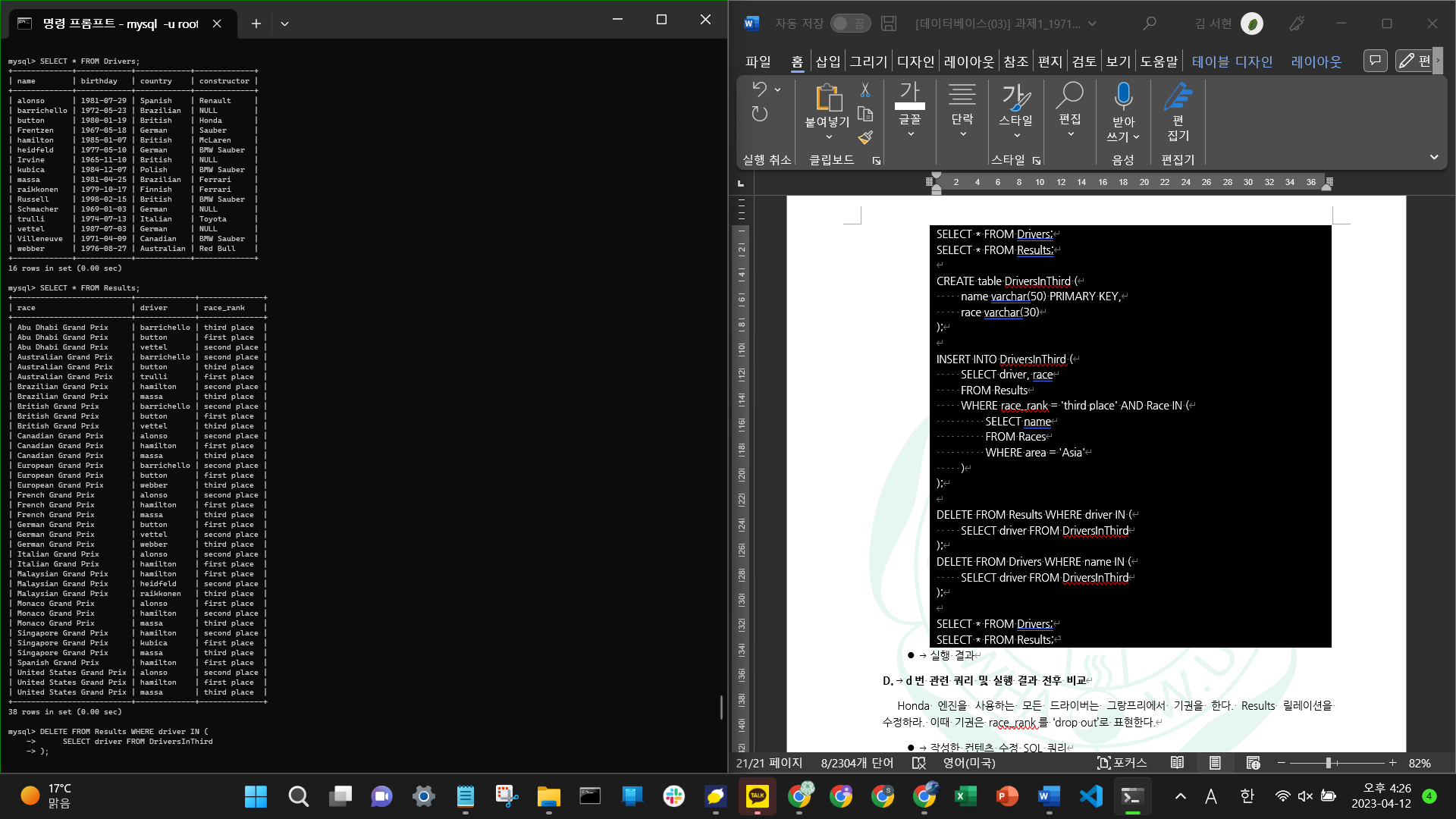
아시아 지역의 그랑프리에서 3등을 차지한 모든 드라이버들을 Drivers 릴레이션에서 삭제하라. 해당 드라이버의 경기 기록도 삭제한다.

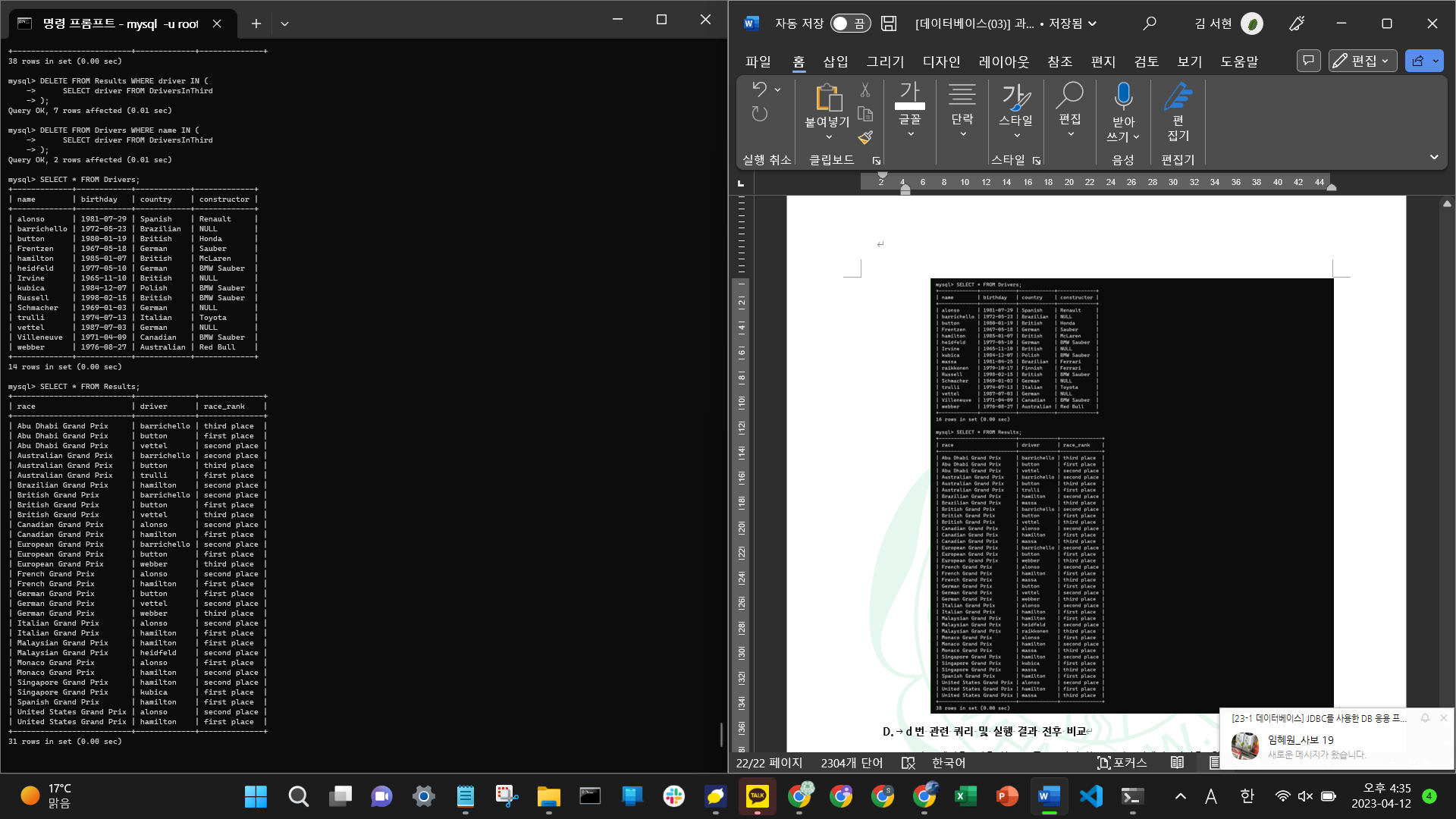
* + 작성한 컨텐츠 수정 SQL 쿼리

|  |
| --- |
| -- Create Table to Store Drivers Who Won Third Places  CREATE TABLE DriversInThird (  name varchar(50) PRIMARY KEY,  race varchar(30)  );  -- Insert Drivers Who Won Third Places  INSERT INTO DriversInThird (  SELECT driver, race  FROM Results  WHERE race\_rank = 'third place' AND Race IN (  SELECT name  FROM Races  WHERE area = 'Asia'  )  );  -- View Current Data  SELECT \* FROM Drivers;  SELECT \* FROM Results;  -- Delete Data  DELETE FROM Results WHERE driver IN (  SELECT driver FROM DriversInThird  );  DELETE FROM Drivers WHERE name IN (  SELECT driver FROM DriversInThird  );  -- View Changes  SELECT \* FROM Drivers;  SELECT \* FROM Results; |

* + 실행 결과







* 1. d번 관련 쿼리 및 실행 결과 전후 비교

Honda 엔진을 사용하는 모든 드라이버는 그랑프리에서 기권을 한다. Results 릴레이션을 수정하라. 이때 기권은 race\_rank를 ‘drop out’로 표현한다.

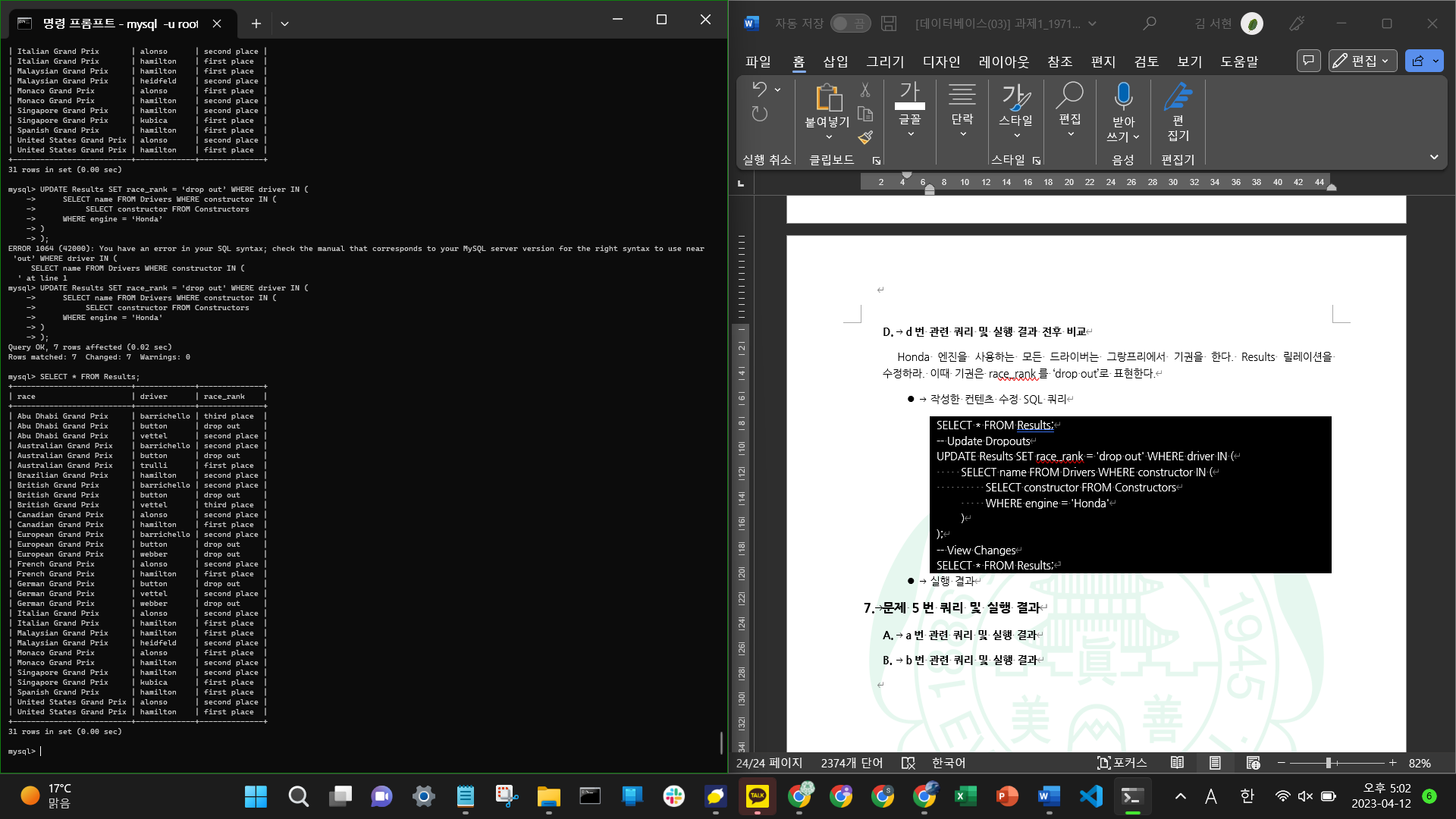
* + 작성한 컨텐츠 수정 SQL 쿼리

|  |
| --- |
| SELECT \* FROM Results;  -- Update Dropouts  UPDATE Results SET race\_rank = 'drop out' WHERE driver IN (  SELECT name FROM Drivers WHERE constructor IN (  SELECT constructor FROM Constructors  WHERE engine = 'Honda'  )  );  -- View Changes  SELECT \* FROM Results; |

* + 실행 결과

텍스트, 전자제품, 스크린샷, 디스플레이이(가) 표시된 사진

자동 생성된 설명



1. 문제 5번 쿼리 및 실행 결과

위의 데이터베이스 스키마를 기반으로 스키마 수정(modification)을 관련된 다음의 SQL 쿼리를 작성했습니다. 각각 작성한 쿼리 및 실행과는 다음과 같습니다.

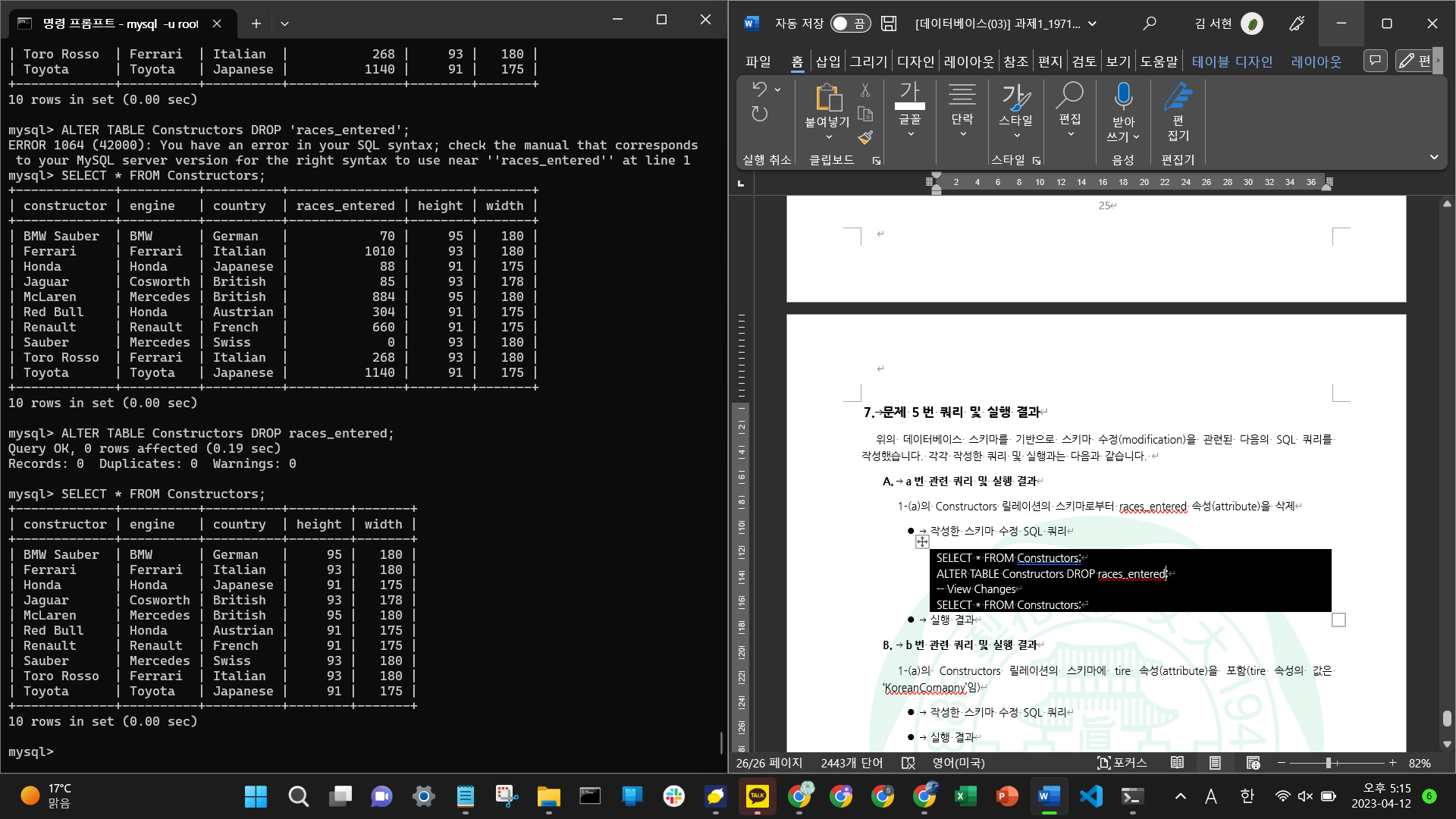
* 1. a번 관련 쿼리 및 실행 결과

1-(a)의 Constructors 릴레이션의 스키마로부터 races\_entered 속성(attribute)을 삭제

* + - 작성한 스키마 수정 SQL 쿼리

|  |
| --- |
| SELECT \* FROM Constructors;  ALTER TABLE Constructors DROP races\_entered;  -- View Changes  SELECT \* FROM Constructors; |

* + - 실행 결과



* 1. b번 관련 쿼리 및 실행 결과

1-(a)의 Constructors 릴레이션의 스키마에 tire 속성(attribute)을 포함(tire 속성의 값은 ‘KoreanComapny’임)

* + - 작성한 스키마 수정 SQL 쿼리

|  |
| --- |
| SELECT \* FROM Constructors;  ALTER TABLE Constructors ADD tire varchar(30);  UPDATE Constructors SET tire = 'Korean company';  -- View Changes  SELECT \* FROM Constructors; |

* + - 실행 결과

