Database Management System I
CPS 542 – Fall 2023

To: Dr. Eman Elrifaei

From: Group 16
Student: 101792276, Hava, Saivedant - Team leader
Student: 1016351520, Sheldon, Daniel
Student: 1017922060, Joshi, Ketan
Student: 1017888630, Bojja, Spoorthi
Student: 1017891080, Chanchad, Anant

Project Title: Formula 1 Race Management Database

Content	Page Number
1) Project Description	2
2) Assumptions	2
3) Entities, Attributes and Keys	3
4) ER-Diagram	5
5) Relational Schema	6
6) Functional Dependencies and Normalisation	7
7) Queries	10
8) Observation	29
9) Conclusion	30

Project Description :

The Formula 1 database is designed to manage comprehensive data related to Formula 1 racing. Its entity set includes: *Races*, *Drivers*, *Teams*, *Results*, *Seasons*, and the weak entity, *RaceSession*. Races hold information about race events, their dates, and locations. Drivers have attributes pertaining to their nationalities and respective teams that are participating in the races. Teams include the team's name and principal details about the team and its participation in various races. Results capture race-specific data, including driver's positions and their attained points. Seasons organize races by year. The RaceSession entity tracks the duration of the race and the changed duration This database serves as a centralized repository for Formula 1-related information and can be used as an aid to facilitate analysis, for historical record keeping, and for reporting to users in the Formula 1 community.

Assumptions:

1. Total Duration of the race is in minutes(120).

2. Circuit length is in miles.

3. This data is for season year 2022.

4. The determination of the individual season champion will be based solely on the individual driver's performance throughout the season, without consideration of team performance.

5. In the race results, the drivers will have their points listed based on their top 2 finishes, where winner will get 25 points and runner up gets 18 points.

6. Points will be awarded to teams based on the performance of their drivers who achieve top - 2 finishes in the race.

7. Team Winner is the maximum Team_Score. The individual winner is the maximum Total_ind_Score.

Entities, Attributes And Keys:

Race: (Race_Name(Primary Key), Date(Year, Month, Day), location(State, Country),Circuit(Circuit_Name, Circuit_length), Year(Foreign Key))

Relationships:

One Race can have many Drivers (1:M)

Many Races in a Season(M:1)

Once Race has a Race Session(1:1)

Drivers: (DriverID(Primary Key), Nationality, Last_Name, First_Name, Total_ind_score, Year(Foreign Key), Team_Name(Foreign Key))

Relationships :

Many Drivers can participate in many races (M: M)

Many Drivers belong to one team (M: 1)

Each Driver has a Result(1:1)

Many Drivers Register for one Season(M:1)

Team: (Team_Name(Primary Key), Principal(P_First_Name, P_Last_Name), Team_Score, Year(Foreign Key))

Relationship:

Many Teams Compete in One Season(M:1)

One Team has Many Drivers(1:M)

One Team has many Results

Results: (Result_ID (Primary Key), Position, Points, Driver_ID(Foreign Key), Team_Name(Foreign Key), Year(Foreign Key))

Relationships:

Many Results correspond to one Team (1:1)

Every Result has a Driver(1:1)(Full Participation)

Season(Year(Primary Key), Team_Winner, Individual_Winner)

Relationships:

One season hosts many races (1:M)

One season has many teams Compete(1:M)

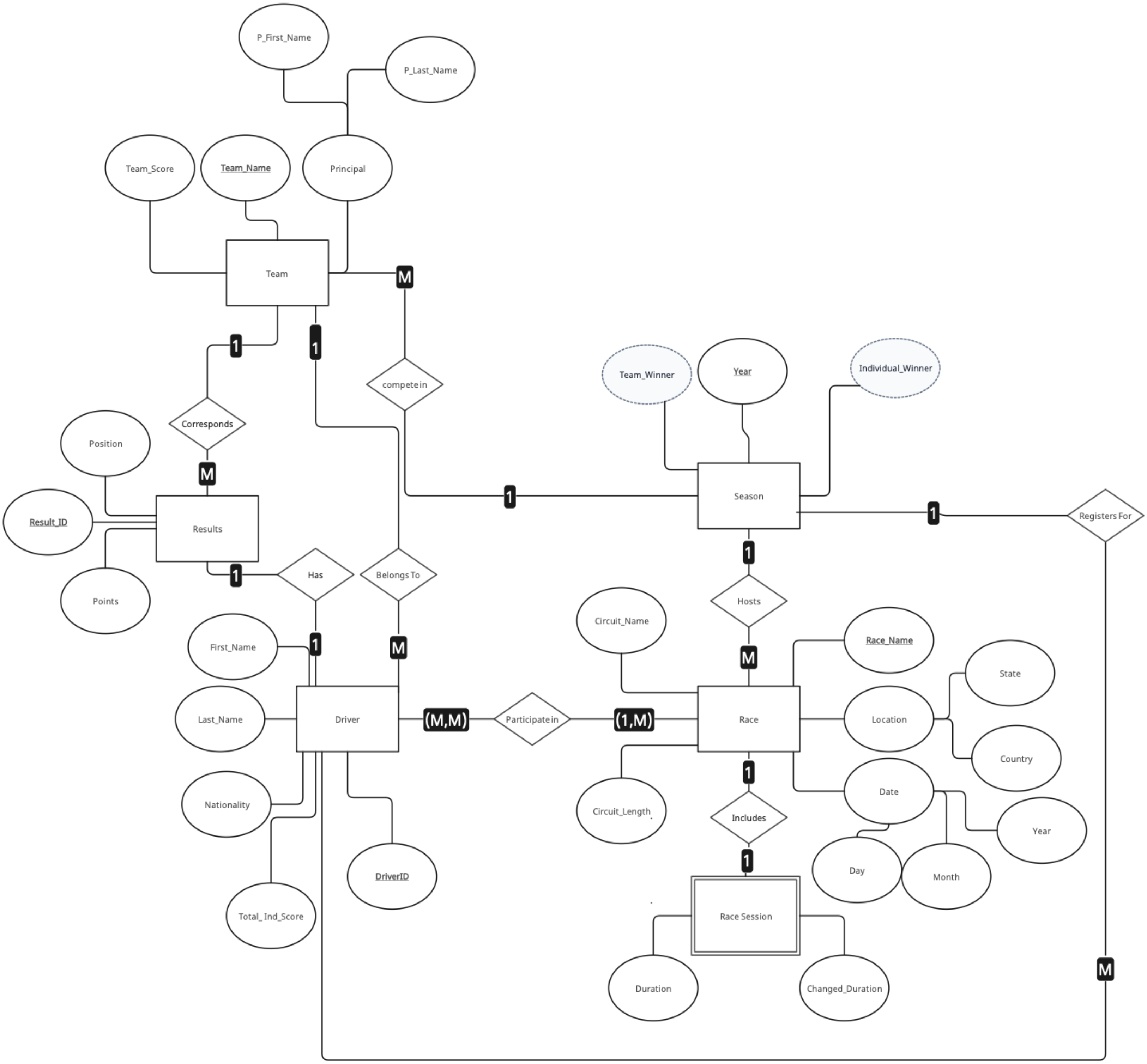
One Season has many Drivers Register(1:M)

RaceSession (Weak Entity)(Race_SessionID(Primary Key), Race_Name(Discriminator), Duration, Changed_Duration)

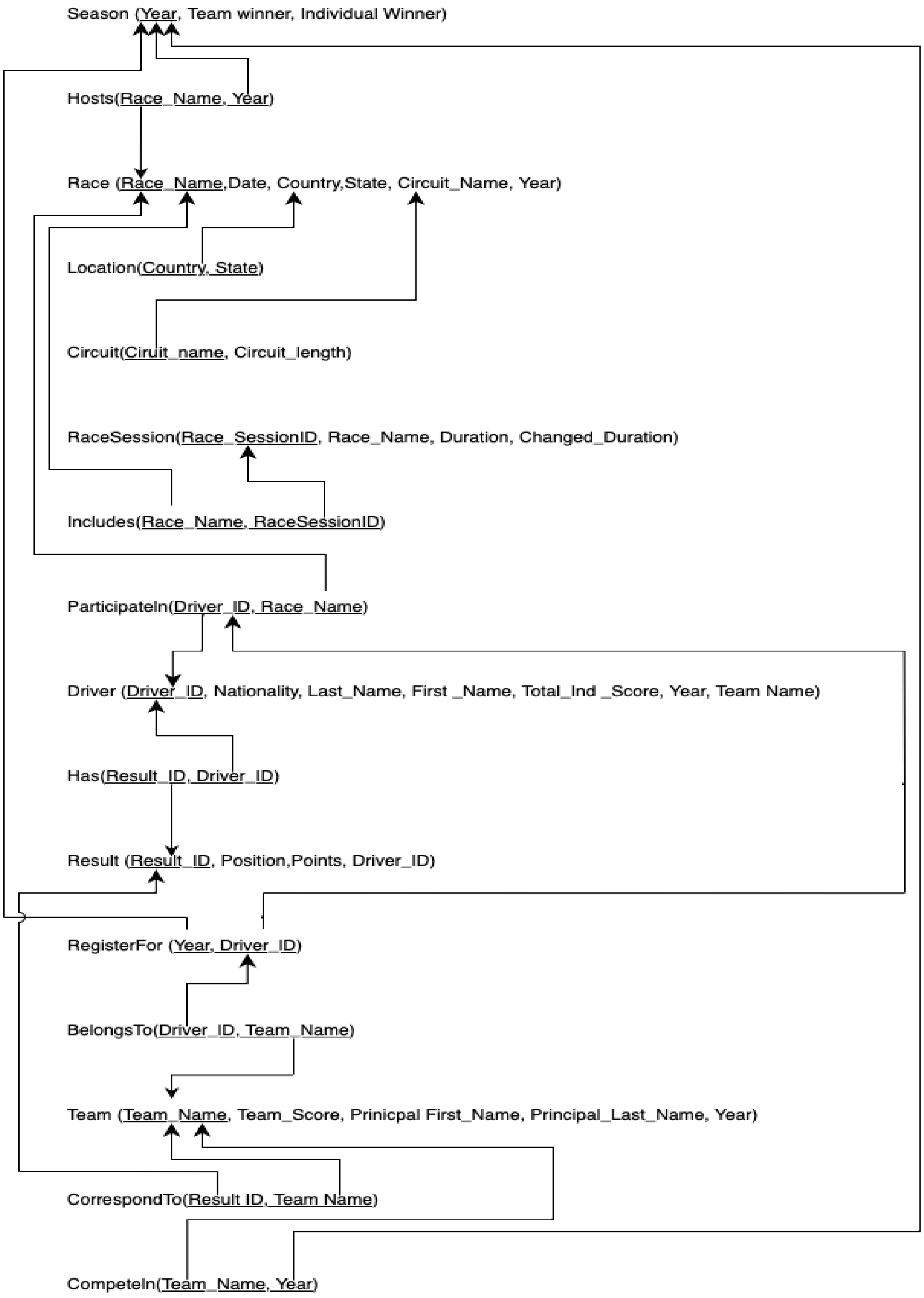
Relationships:

Each RaceSession is included in one Race (1:1)

Entity - Relationship Diagram:



Relational Schema Diagram:



Season (Year, Team winner, Individual Winner)

Hosts(Race_Name, Year)

Race (Race_Name,Date, Country,State, Circuit_Name, Year)

Location(Country, State)

Circuit(Ciruit_name, Circuit_length)

RaceSession(Race_SessionID, Race_Name, Duration, Changed_Duration)

Includes(Race_Name, RaceSessionID)

ParticipateIn(Driver_ID, Race_Name)

Driver (Driver_ID, Nationality, Last_Name, First _Name, Total_Ind _Score, Year, Team Name)

Has(Result_ID, Driver_ID)

Result (Result_ID, Position,Points, Driver_ID)

RegisterFor (Year, Driver_ID)

BelongsTo(Driver_ID, Team_Name)

Team (Team_Name, Team_Score, Prinicpal First_Name, Principal_Last_Name, Year)

CorrespondTo(Result ID, Team Name)

CompeteIn(Team_Name, Year)

Functional Dependencies and Normalization:

Our database preserves all attributes and dependencies, ensuring that those mentioned in our ER diagram remain present after normalization. We initially examined our ER diagram to confirm that our relations were in the first and second normal forms, with each attribute allowed only one atomic value. After this verification, we addressed any transitive dependencies to place our database in the third normal form, resolving the following:

- {Race_Name} → {Race_Name, Date, State, Country, Circuit_Name, Year}
- {Location} → {Country, State }
- {Circuit} → {Circuit_Name, Circuit_Length }

These transitive dependencies were resolved by creating new relations. If X->Y, Y was removed from its original relation, and a new relation was formed where X is the primary key, remaining in the original relation, and Y is solely found in the new relation. With this approach, the third normal form is satisfied, ensuring that for each X->A, either X is a super key or A is a candidate key. This dependency preservation is always achievable in the third normal form.

1. Race: (Race_Name, Date, Location(State, Country), Circuit(Circuit_Name, Circuit_length), Year)

- Functional Dependency

  {Location} → {Country, State }

  {Circuit} → {Circuit_Name, Circuit_Length }

- Normalization

  New table and attributes

  {Race_Name} → {Race_Name, Date, State, Country, Circuit_Name, Year}

  {Location} → {Country, State }

  {Circuit} → {Circuit_Name, Circuit_Length }

2. Drivers: (Driver_ID, Nationality, Last_Name, First_Name, Total_ind_score, Year, Team_Name)

- Functional Dependency

{Driver_ID} → {Driver_ID, Nationality, Last_Name, First_Name, Total_ind_score, Year, Team_Name}

- Normalization

The above entity is already in Boyce Codd Normal form as all the non-key attributes depend on the Driver_ID, which is the primary key.

3. Team: (Team_Name, Principal(P_First_Name, P_Last_Name), Team_Score, Year)

- Functional Dependency

{Team_Name} → {Team_Name, Principal(P_First_Name, P_Last_Name), Team_Score, Year}

- Normalization

The above entity is in Boyce Codd Normal form as all the non-key attributes depend on the Team_Name, which is a primary key.

4. Results: (Result_ID, Position, Points, Driver_ID, Team_Name, Year)

- Functional Dependency

{Result_ID} → {Result_ID, Position, Points, Driver_ID, Team_Name, Year}

- Normalization

The above entity is in Boyce Codd Normal form as all the non-key attributes depend on the Result_ID, which is a primary key.

5. Season(Year(Primary Key), Team_Winner, Individual_Winner)

- Functional Dependency

{Year} → {Year, Team_Winner, Individual_Winner}

- Normalization

The above entity is in Boyce Codd Normal form as all the non-key attributes depend on the Year, which is the primary key.

6. RaceSession (Race_SessionID, Race_Name, Duration, Changed_Duration)

- Functional Dependency

{Race_SessionID} → {(Race_SessionID, Race_Name, Duration, Changed_Duration)}

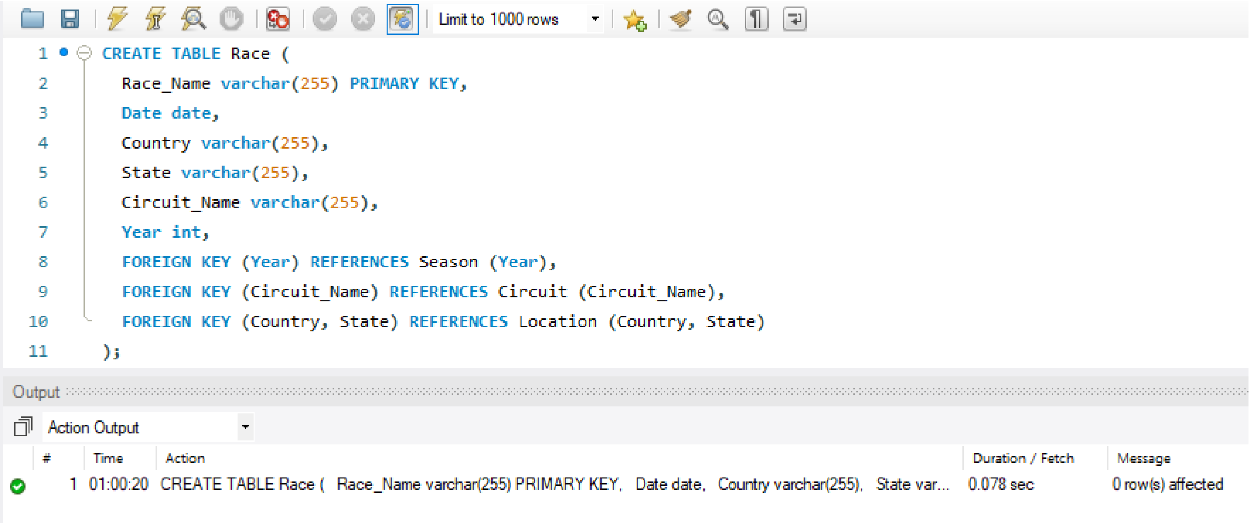
- Normalization

The above entity is in Boyce Codd Normal form as all the non-key attributes depend on the Race_SessionID, which is the primary key.

Queries:

- **Queries to create table**

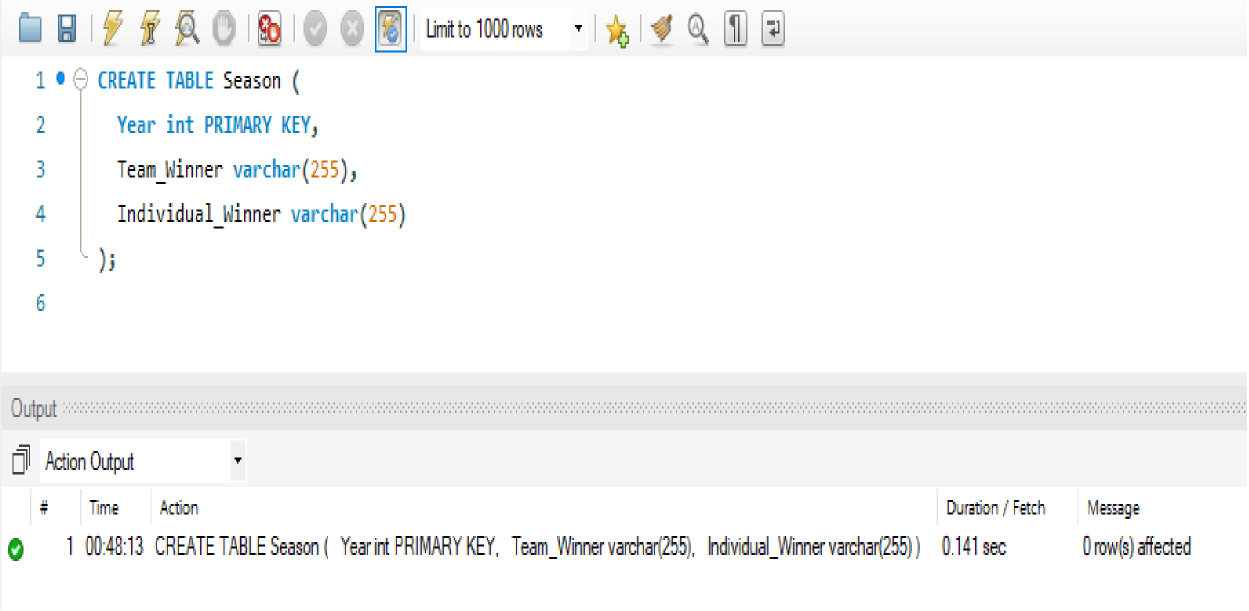
1) Table: Race



```
CREATE TABLE Race (
    Race_Name varchar(255) PRIMARY KEY,
    Date date,
    Country varchar(255),
    State varchar(255),
    Circuit_Name varchar(255),
    Year int,
    FOREIGN KEY (Year) REFERENCES Season (Year),
    FOREIGN KEY (Circuit_Name) REFERENCES Circuit (Circuit_Name),
    FOREIGN KEY (Country, State) REFERENCES Location (Country, State)
);
```

#	Time	Action	Duration / Fetch	Message
1	01:00:20	CREATE TABLE Race (Race_Name varchar(255) PRIMARY KEY, Date date, Country varchar(255), State var...	0.078 sec	0 row(s) affected

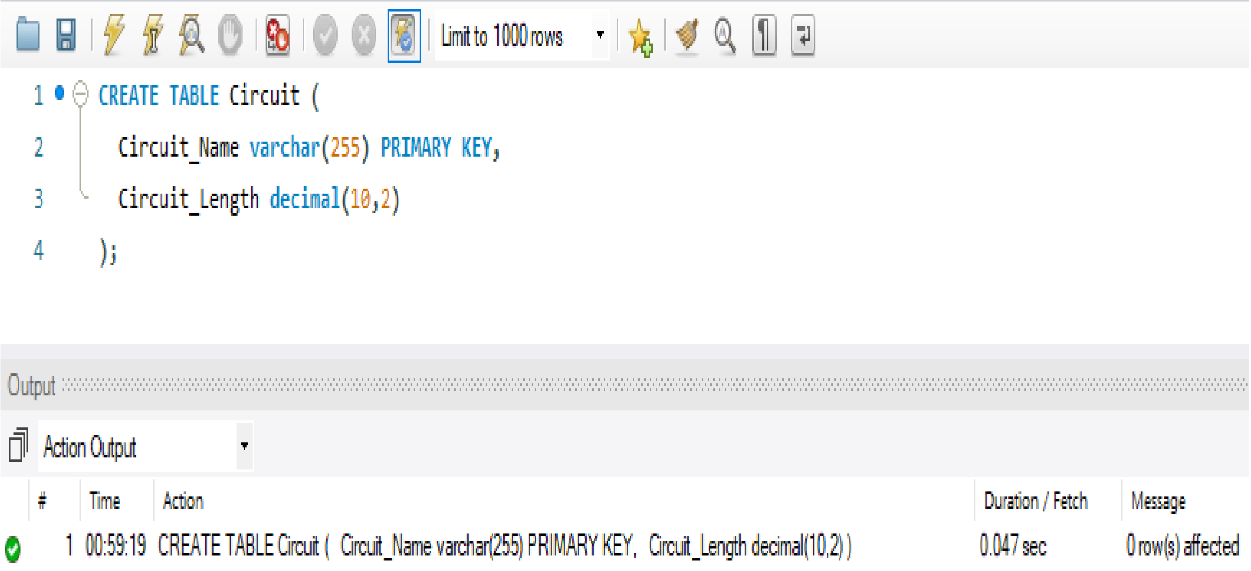
2) Table: Season



```
CREATE TABLE Season (
    Year int PRIMARY KEY,
    Team_Winner varchar(255),
    Individual_Winner varchar(255)
);
```

#	Time	Action	Duration / Fetch	Message
1	00:48:13	CREATE TABLE Season (Year int PRIMARY KEY, Team_Winner varchar(255), Individual_Winner varchar(255))	0.141 sec	0 row(s) affected

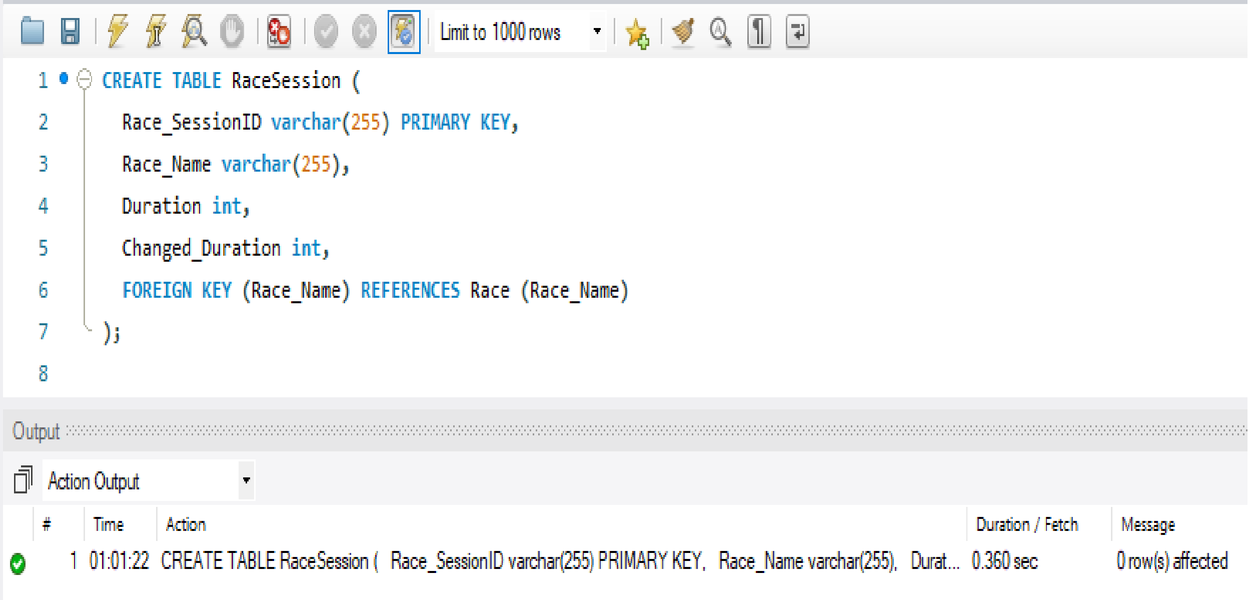
3) Table: Circuit



```
CREATE TABLE Circuit (
    Circuit_Name varchar(255) PRIMARY KEY,
    Circuit_Length decimal(10,2)
);
```

```
#  Time      Action                                                                                  Duration / Fetch  Message
1  00:59:19  CREATE TABLE Circuit (  Circuit_Name varchar(255) PRIMARY KEY,  Circuit_Length decimal(10,2) )  0.047 sec         0 row(s) affected
```

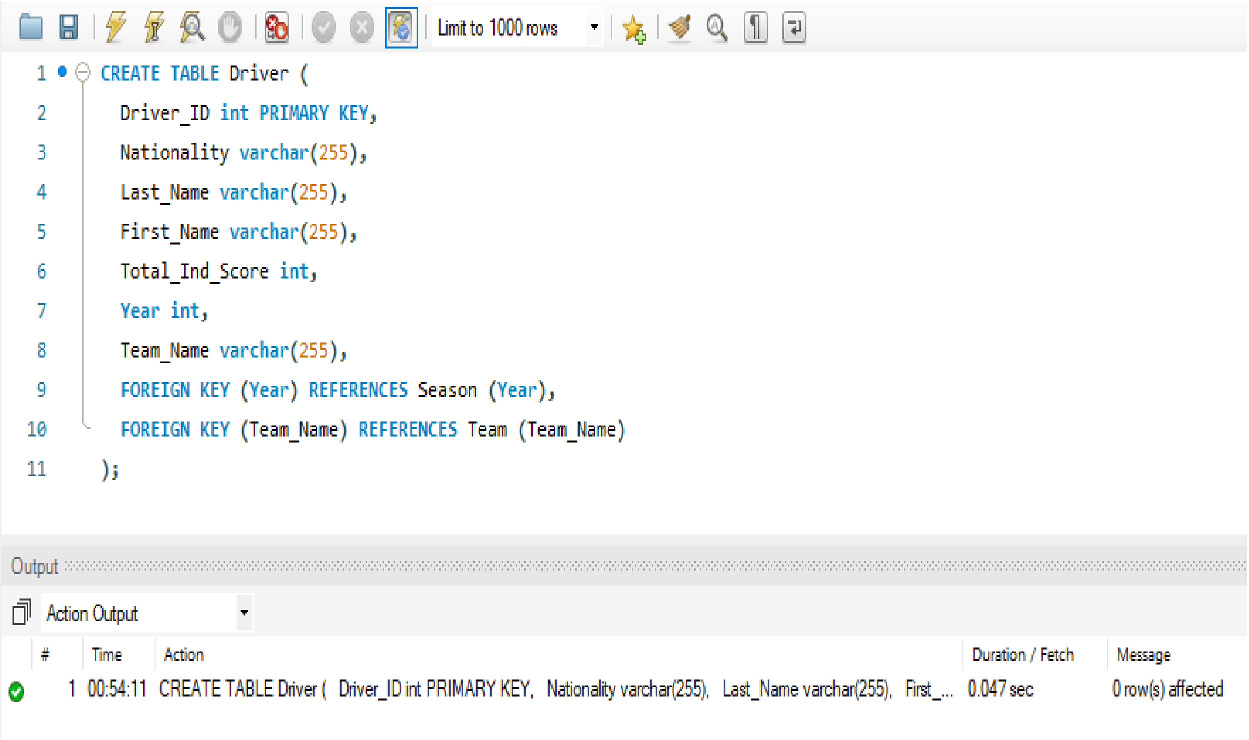
4) Table: Race Session



```
CREATE TABLE RaceSession (
    Race_SessionID varchar(255) PRIMARY KEY,
    Race_Name varchar(255),
    Duration int,
    Changed_Duration int,
    FOREIGN KEY (Race_Name) REFERENCES Race (Race_Name)
);
```

```
#  Time      Action                                                                                        Duration / Fetch  Message
1  01:01:22  CREATE TABLE RaceSession (  Race_SessionID varchar(255) PRIMARY KEY,  Race_Name varchar(255),  Durat...  0.360 sec         0 row(s) affected
```

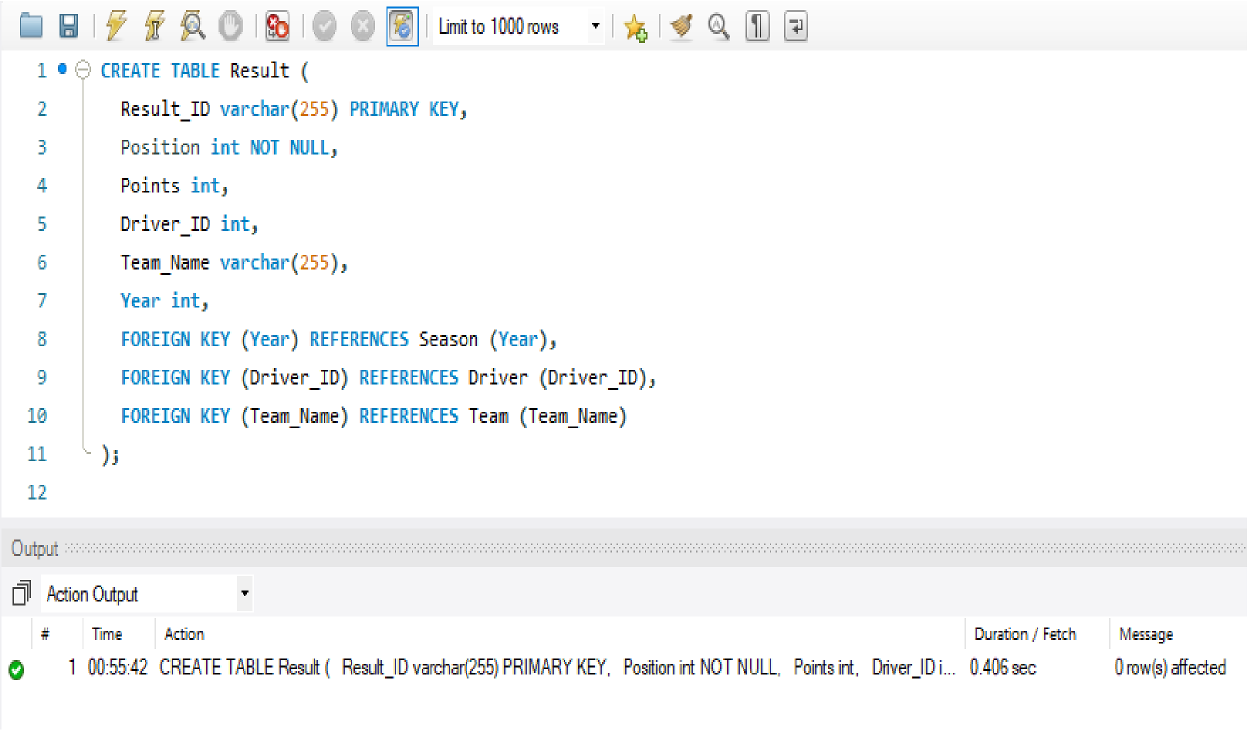
5) Table: Driver



```sql
CREATE TABLE Driver (
  Driver_ID int PRIMARY KEY,
  Nationality varchar(255),
  Last_Name varchar(255),
  First_Name varchar(255),
  Total_Ind_Score int,
  Year int,
  Team_Name varchar(255),
  FOREIGN KEY (Year) REFERENCES Season (Year),
  FOREIGN KEY (Team_Name) REFERENCES Team (Team_Name)
);
```

```
   #  Time      Action                                                                                                    Duration / Fetch   Message
✓  1  00:54:11  CREATE TABLE Driver (  Driver_ID int PRIMARY KEY,  Nationality varchar(255),  Last_Name varchar(255),  First_...  0.047 sec   0 row(s) affected
```

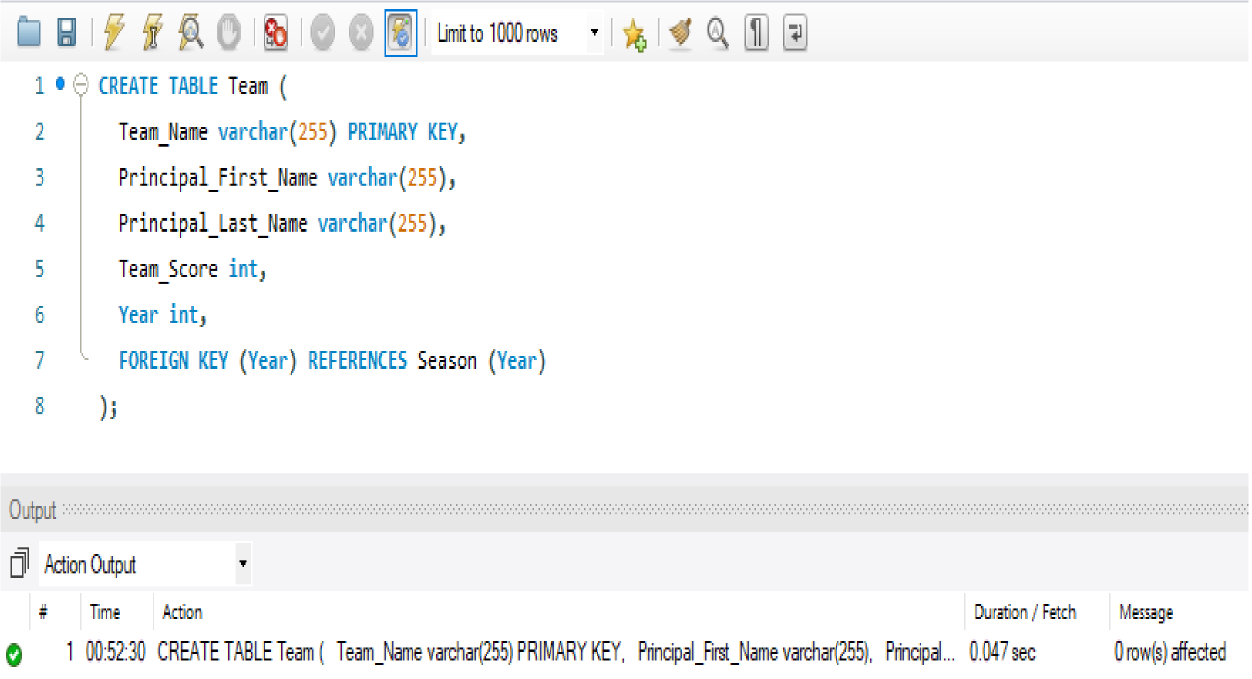
6) Result: Result



```sql
CREATE TABLE Result (
  Result_ID varchar(255) PRIMARY KEY,
  Position int NOT NULL,
  Points int,
  Driver_ID int,
  Team_Name varchar(255),
  Year int,
  FOREIGN KEY (Year) REFERENCES Season (Year),
  FOREIGN KEY (Driver_ID) REFERENCES Driver (Driver_ID),
  FOREIGN KEY (Team_Name) REFERENCES Team (Team_Name)
);
```

```
   #  Time      Action                                                                                               Duration / Fetch   Message
✓  1  00:55:42  CREATE TABLE Result (  Result_ID varchar(255) PRIMARY KEY,  Position int NOT NULL,  Points int,  Driver_ID i...  0.406 sec   0 row(s) affected
```

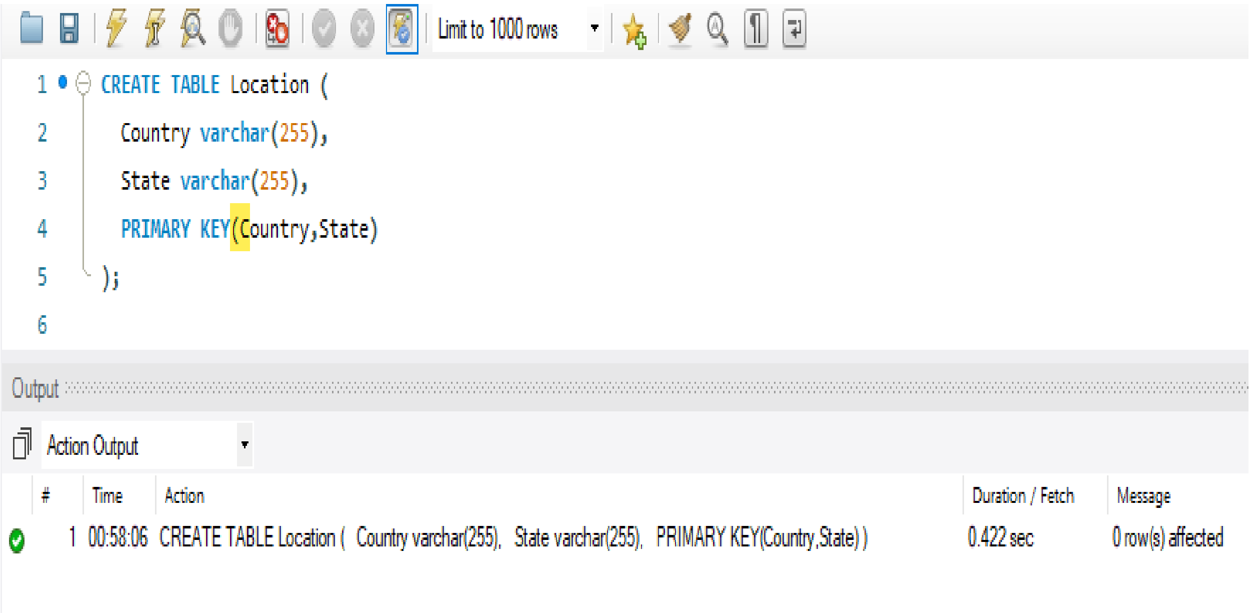
7) Table: Team



```
CREATE TABLE Team (
    Team_Name varchar(255) PRIMARY KEY,
    Principal_First_Name varchar(255),
    Principal_Last_Name varchar(255),
    Team_Score int,
    Year int,
    FOREIGN KEY (Year) REFERENCES Season (Year)
);
```

	#	Time	Action	Duration / Fetch	Message
✓	1	00:52:30	CREATE TABLE Team (Team_Name varchar(255) PRIMARY KEY, Principal_First_Name varchar(255), Principal...	0.047 sec	0 row(s) affected

8) Table: Location



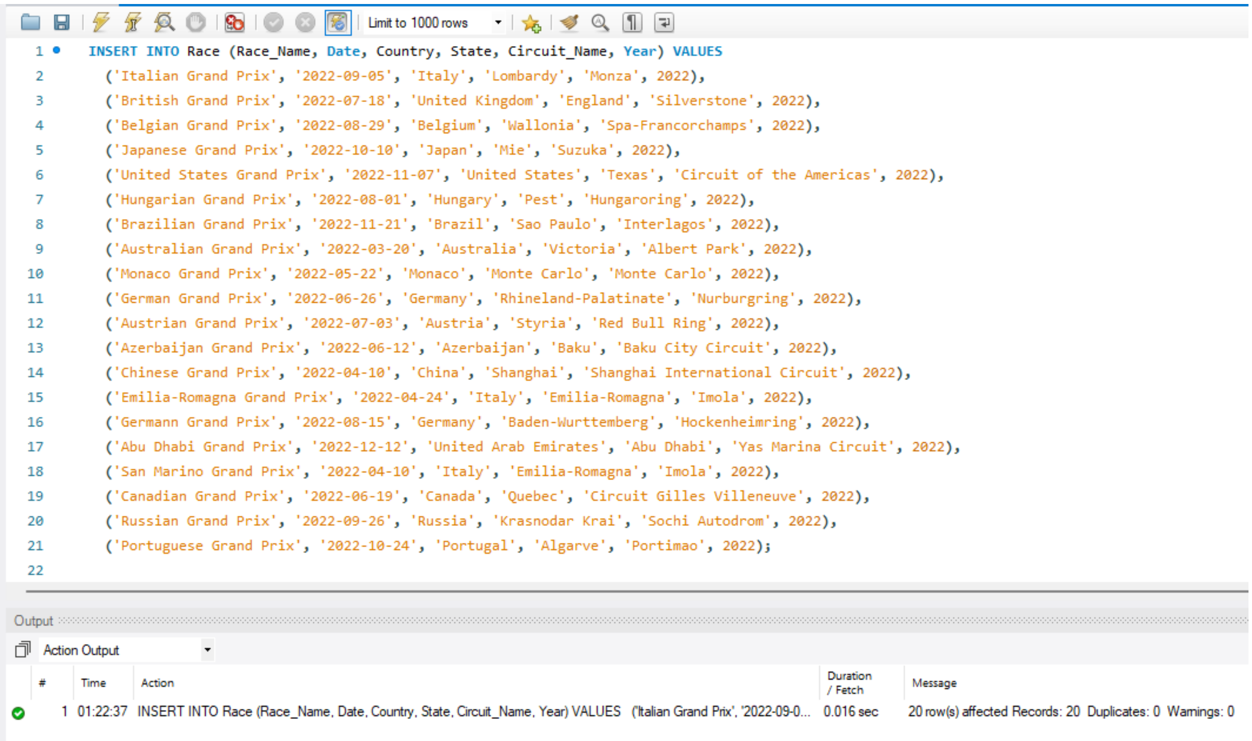
```
CREATE TABLE Location (
    Country varchar(255),
    State varchar(255),
    PRIMARY KEY(Country,State)
);

```

	#	Time	Action	Duration / Fetch	Message
✓	1	00:58:06	CREATE TABLE Location (Country varchar(255), State varchar(255), PRIMARY KEY(Country,State))	0.422 sec	0 row(s) affected

- **Queries to INSERT data**

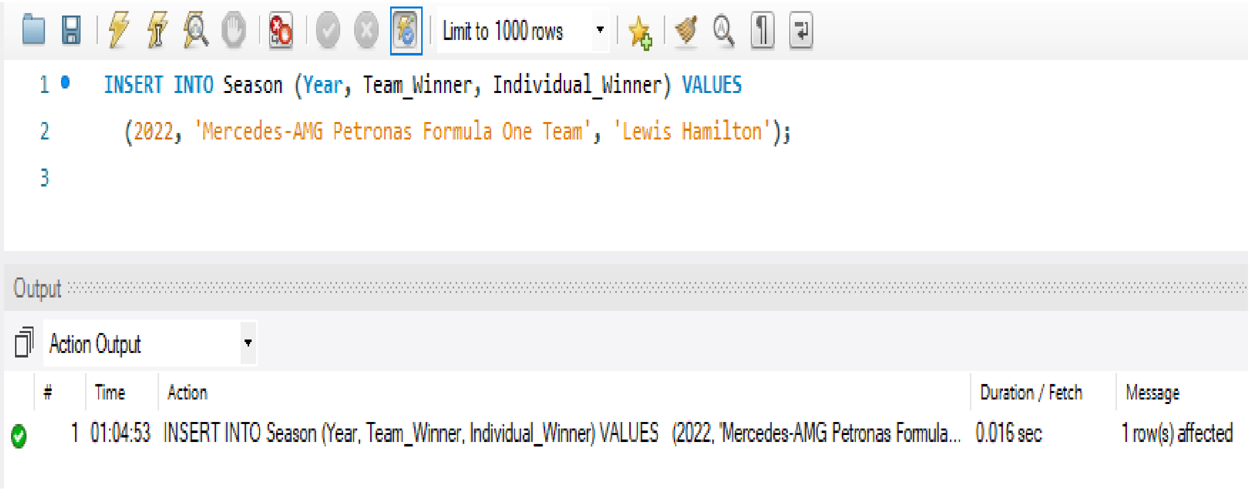
1) INSERT into Race table:



```sql
INSERT INTO Race (Race_Name, Date, Country, State, Circuit_Name, Year) VALUES
  ('Italian Grand Prix', '2022-09-05', 'Italy', 'Lombardy', 'Monza', 2022),
  ('British Grand Prix', '2022-07-18', 'United Kingdom', 'England', 'Silverstone', 2022),
  ('Belgian Grand Prix', '2022-08-29', 'Belgium', 'Wallonia', 'Spa-Francorchamps', 2022),
  ('Japanese Grand Prix', '2022-10-10', 'Japan', 'Mie', 'Suzuka', 2022),
  ('United States Grand Prix', '2022-11-07', 'United States', 'Texas', 'Circuit of the Americas', 2022),
  ('Hungarian Grand Prix', '2022-08-01', 'Hungary', 'Pest', 'Hungaroring', 2022),
  ('Brazilian Grand Prix', '2022-11-21', 'Brazil', 'Sao Paulo', 'Interlagos', 2022),
  ('Australian Grand Prix', '2022-03-20', 'Australia', 'Victoria', 'Albert Park', 2022),
  ('Monaco Grand Prix', '2022-05-22', 'Monaco', 'Monte Carlo', 'Monte Carlo', 2022),
  ('German Grand Prix', '2022-06-26', 'Germany', 'Rhineland-Palatinate', 'Nurburgring', 2022),
  ('Austrian Grand Prix', '2022-07-03', 'Austria', 'Styria', 'Red Bull Ring', 2022),
  ('Azerbaijan Grand Prix', '2022-06-12', 'Azerbaijan', 'Baku', 'Baku City Circuit', 2022),
  ('Chinese Grand Prix', '2022-04-10', 'China', 'Shanghai', 'Shanghai International Circuit', 2022),
  ('Emilia-Romagna Grand Prix', '2022-04-24', 'Italy', 'Emilia-Romagna', 'Imola', 2022),
  ('Germann Grand Prix', '2022-08-15', 'Germany', 'Baden-Wurttemberg', 'Hockenheimring', 2022),
  ('Abu Dhabi Grand Prix', '2022-12-12', 'United Arab Emirates', 'Abu Dhabi', 'Yas Marina Circuit', 2022),
  ('San Marino Grand Prix', '2022-04-10', 'Italy', 'Emilia-Romagna', 'Imola', 2022),
  ('Canadian Grand Prix', '2022-06-19', 'Canada', 'Quebec', 'Circuit Gilles Villeneuve', 2022),
  ('Russian Grand Prix', '2022-09-26', 'Russia', 'Krasnodar Krai', 'Sochi Autodrom', 2022),
  ('Portuguese Grand Prix', '2022-10-24', 'Portugal', 'Algarve', 'Portimao', 2022);
```

	#	Time	Action	Duration / Fetch	Message
✓	1	01:22:37	INSERT INTO Race (Race_Name, Date, Country, State, Circuit_Name, Year) VALUES ('Italian Grand Prix', '2022-09-0...	0.016 sec	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0

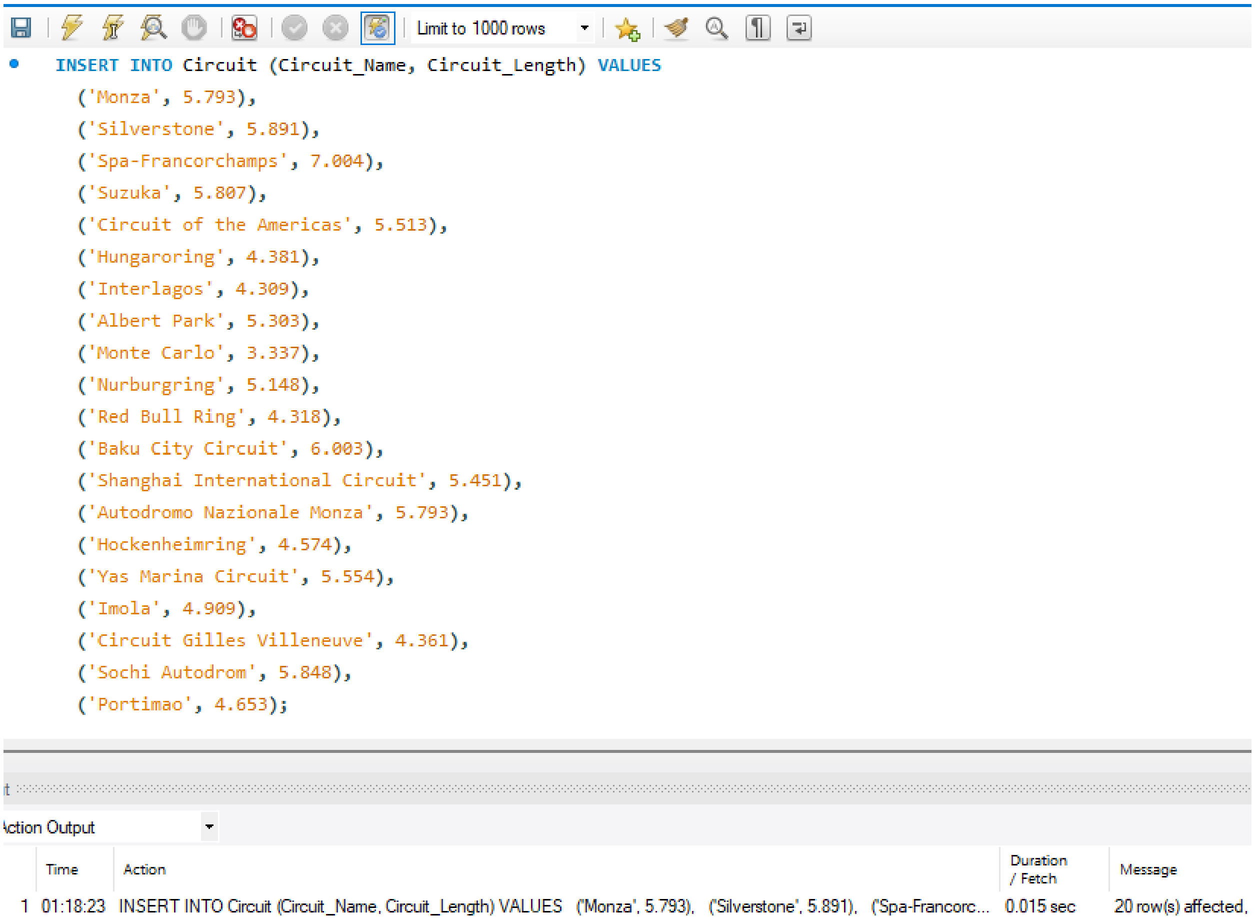
2) INSERT into Season table (Including only one season in the database is a result of the challenge of simultaneously filling numerous entries at the moment):



```sql
INSERT INTO Season (Year, Team_Winner, Individual_Winner) VALUES
  (2022, 'Mercedes-AMG Petronas Formula One Team', 'Lewis Hamilton');
```

	#	Time	Action	Duration / Fetch	Message
✓	1	01:04:53	INSERT INTO Season (Year, Team_Winner, Individual_Winner) VALUES (2022, 'Mercedes-AMG Petronas Formula...	0.016 sec	1 row(s) affected

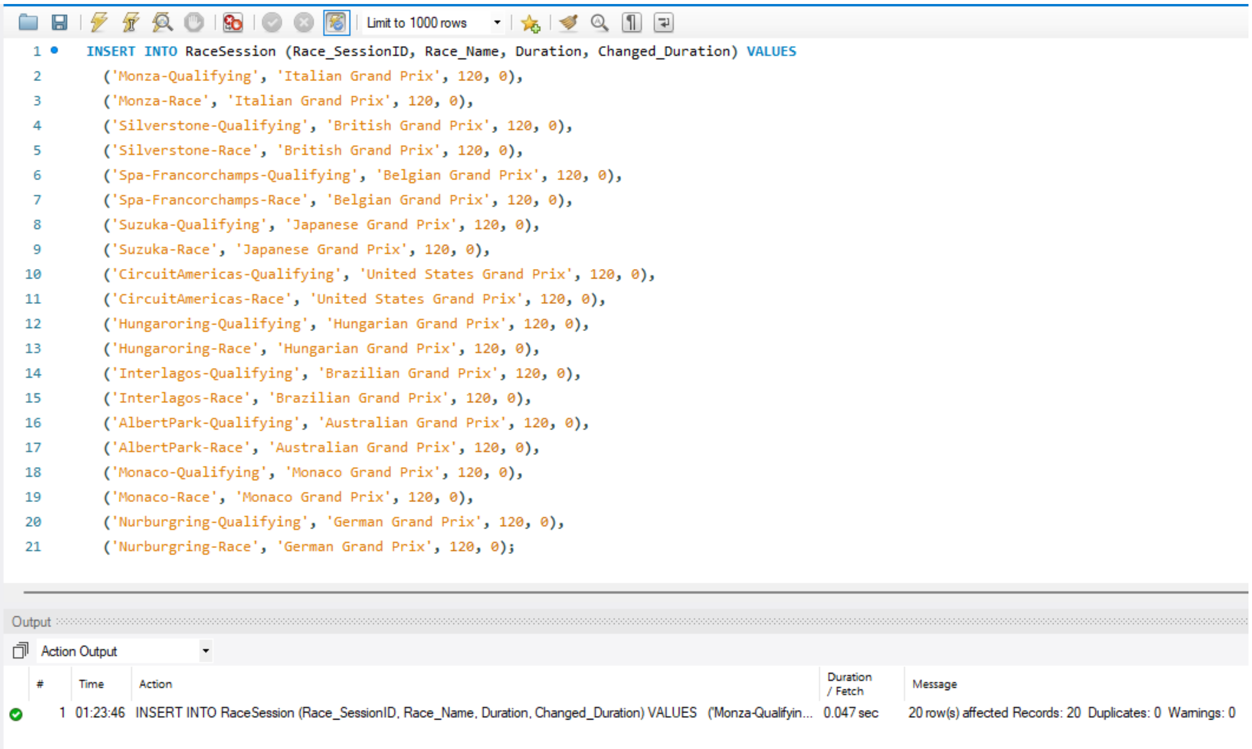
3) INSERT into table Circuit:



```
INSERT INTO Circuit (Circuit_Name, Circuit_Length) VALUES
  ('Monza', 5.793),
  ('Silverstone', 5.891),
  ('Spa-Francorchamps', 7.004),
  ('Suzuka', 5.807),
  ('Circuit of the Americas', 5.513),
  ('Hungaroring', 4.381),
  ('Interlagos', 4.309),
  ('Albert Park', 5.303),
  ('Monte Carlo', 3.337),
  ('Nurburgring', 5.148),
  ('Red Bull Ring', 4.318),
  ('Baku City Circuit', 6.003),
  ('Shanghai International Circuit', 5.451),
  ('Autodromo Nazionale Monza', 5.793),
  ('Hockenheimring', 4.574),
  ('Yas Marina Circuit', 5.554),
  ('Imola', 4.909),
  ('Circuit Gilles Villeneuve', 4.361),
  ('Sochi Autodrom', 5.848),
  ('Portimao', 4.653);
```

	Time	Action	Duration / Fetch	Message
1	01:18:23	INSERT INTO Circuit (Circuit_Name, Circuit_Length) VALUES ('Monza', 5.793), ('Silverstone', 5.891), ('Spa-Francorc...	0.015 sec	20 row(s) affected,

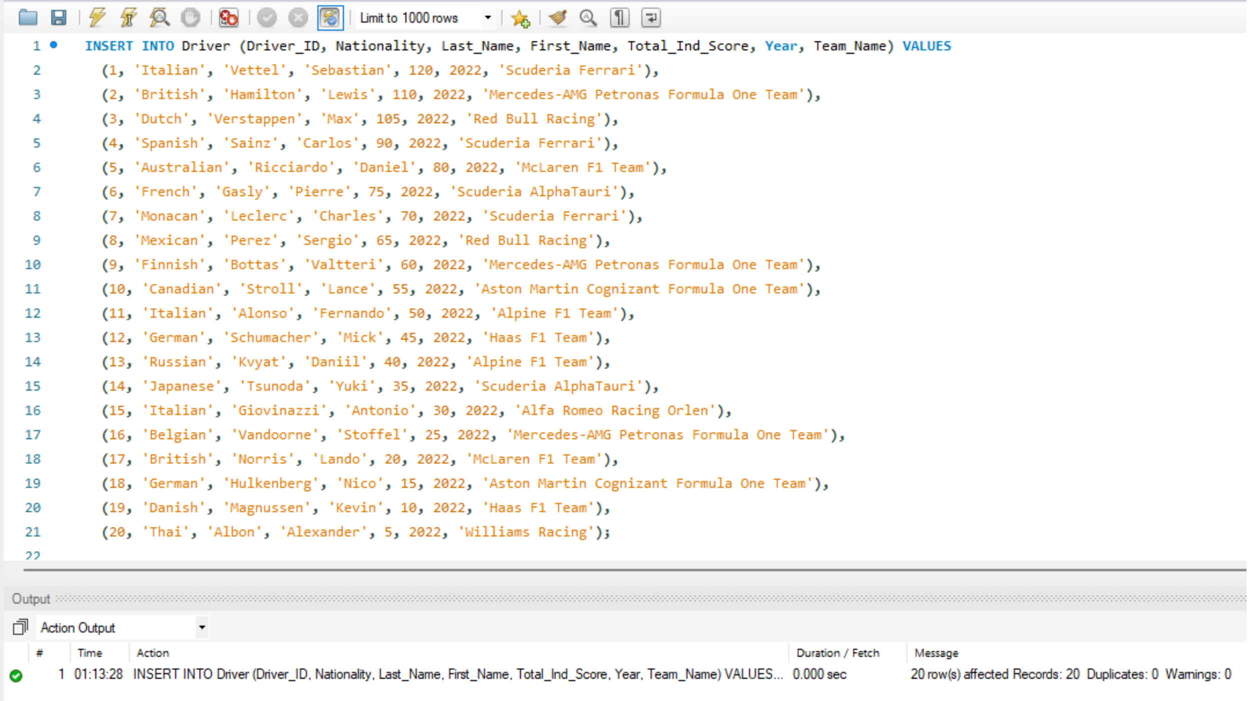
4) INSERT into table Race Session:



```sql
INSERT INTO RaceSession (Race_SessionID, Race_Name, Duration, Changed_Duration) VALUES
  ('Monza-Qualifying', 'Italian Grand Prix', 120, 0),
  ('Monza-Race', 'Italian Grand Prix', 120, 0),
  ('Silverstone-Qualifying', 'British Grand Prix', 120, 0),
  ('Silverstone-Race', 'British Grand Prix', 120, 0),
  ('Spa-Francorchamps-Qualifying', 'Belgian Grand Prix', 120, 0),
  ('Spa-Francorchamps-Race', 'Belgian Grand Prix', 120, 0),
  ('Suzuka-Qualifying', 'Japanese Grand Prix', 120, 0),
  ('Suzuka-Race', 'Japanese Grand Prix', 120, 0),
  ('CircuitAmericas-Qualifying', 'United States Grand Prix', 120, 0),
  ('CircuitAmericas-Race', 'United States Grand Prix', 120, 0),
  ('Hungaroring-Qualifying', 'Hungarian Grand Prix', 120, 0),
  ('Hungaroring-Race', 'Hungarian Grand Prix', 120, 0),
  ('Interlagos-Qualifying', 'Brazilian Grand Prix', 120, 0),
  ('Interlagos-Race', 'Brazilian Grand Prix', 120, 0),
  ('AlbertPark-Qualifying', 'Australian Grand Prix', 120, 0),
  ('AlbertPark-Race', 'Australian Grand Prix', 120, 0),
  ('Monaco-Qualifying', 'Monaco Grand Prix', 120, 0),
  ('Monaco-Race', 'Monaco Grand Prix', 120, 0),
  ('Nurburgring-Qualifying', 'German Grand Prix', 120, 0),
  ('Nurburgring-Race', 'German Grand Prix', 120, 0);
```

```
  #  Time      Action                                                                                                  Duration / Fetch  Message
✓ 1  01:23:46  INSERT INTO RaceSession (Race_SessionID, Race_Name, Duration, Changed_Duration) VALUES  ('Monza-Qualifyin...  0.047 sec  20 row(s) affected Records: 20  Duplicates: 0  Warnings: 0
```

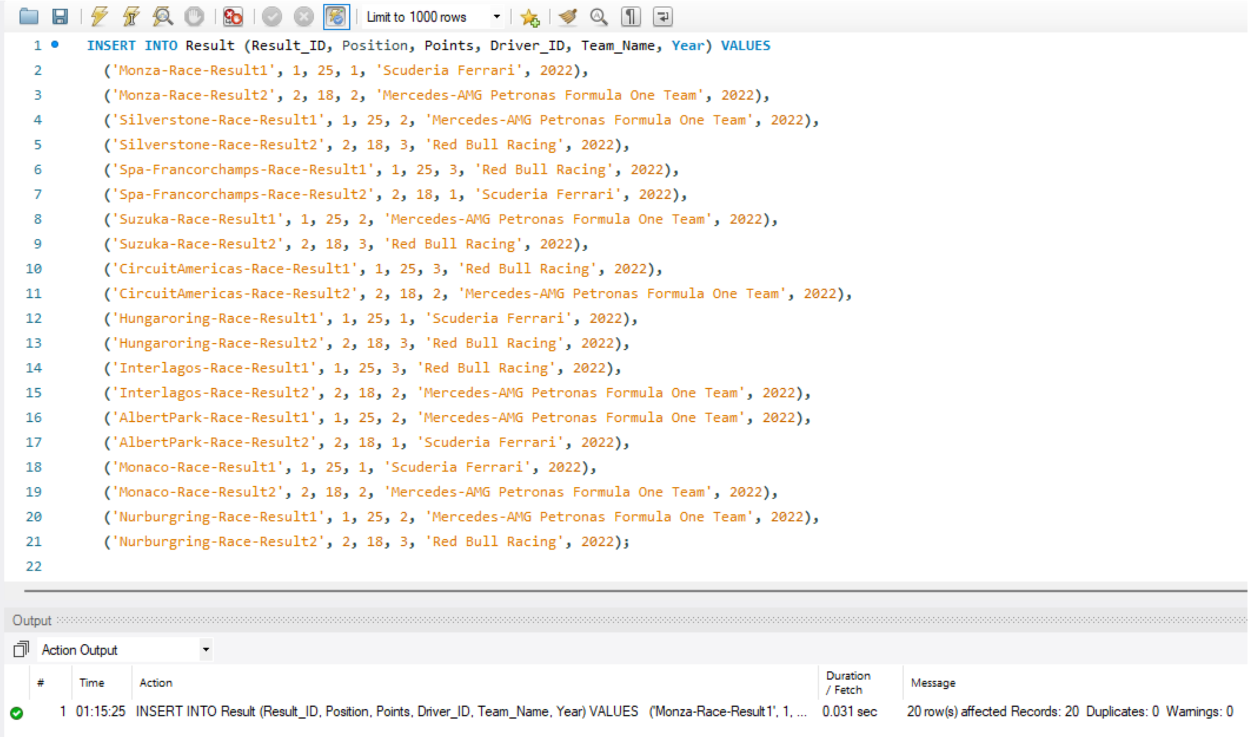
5) INSERT into table Driver:



```sql
INSERT INTO Driver (Driver_ID, Nationality, Last_Name, First_Name, Total_Ind_Score, Year, Team_Name) VALUES
  (1, 'Italian', 'Vettel', 'Sebastian', 120, 2022, 'Scuderia Ferrari'),
  (2, 'British', 'Hamilton', 'Lewis', 110, 2022, 'Mercedes-AMG Petronas Formula One Team'),
  (3, 'Dutch', 'Verstappen', 'Max', 105, 2022, 'Red Bull Racing'),
  (4, 'Spanish', 'Sainz', 'Carlos', 90, 2022, 'Scuderia Ferrari'),
  (5, 'Australian', 'Ricciardo', 'Daniel', 80, 2022, 'McLaren F1 Team'),
  (6, 'French', 'Gasly', 'Pierre', 75, 2022, 'Scuderia AlphaTauri'),
  (7, 'Monacan', 'Leclerc', 'Charles', 70, 2022, 'Scuderia Ferrari'),
  (8, 'Mexican', 'Perez', 'Sergio', 65, 2022, 'Red Bull Racing'),
  (9, 'Finnish', 'Bottas', 'Valtteri', 60, 2022, 'Mercedes-AMG Petronas Formula One Team'),
  (10, 'Canadian', 'Stroll', 'Lance', 55, 2022, 'Aston Martin Cognizant Formula One Team'),
  (11, 'Italian', 'Alonso', 'Fernando', 50, 2022, 'Alpine F1 Team'),
  (12, 'German', 'Schumacher', 'Mick', 45, 2022, 'Haas F1 Team'),
  (13, 'Russian', 'Kvyat', 'Daniil', 40, 2022, 'Alpine F1 Team'),
  (14, 'Japanese', 'Tsunoda', 'Yuki', 35, 2022, 'Scuderia AlphaTauri'),
  (15, 'Italian', 'Giovinazzi', 'Antonio', 30, 2022, 'Alfa Romeo Racing Orlen'),
  (16, 'Belgian', 'Vandoorne', 'Stoffel', 25, 2022, 'Mercedes-AMG Petronas Formula One Team'),
  (17, 'British', 'Norris', 'Lando', 20, 2022, 'McLaren F1 Team'),
  (18, 'German', 'Hulkenberg', 'Nico', 15, 2022, 'Aston Martin Cognizant Formula One Team'),
  (19, 'Danish', 'Magnussen', 'Kevin', 10, 2022, 'Haas F1 Team'),
  (20, 'Thai', 'Albon', 'Alexander', 5, 2022, 'Williams Racing');
```

```
  #  Time      Action                                                                                                   Duration / Fetch  Message
✓ 1  01:13:28  INSERT INTO Driver (Driver_ID, Nationality, Last_Name, First_Name, Total_Ind_Score, Year, Team_Name) VALUES...  0.000 sec  20 row(s) affected Records: 20  Duplicates: 0  Warnings: 0
```

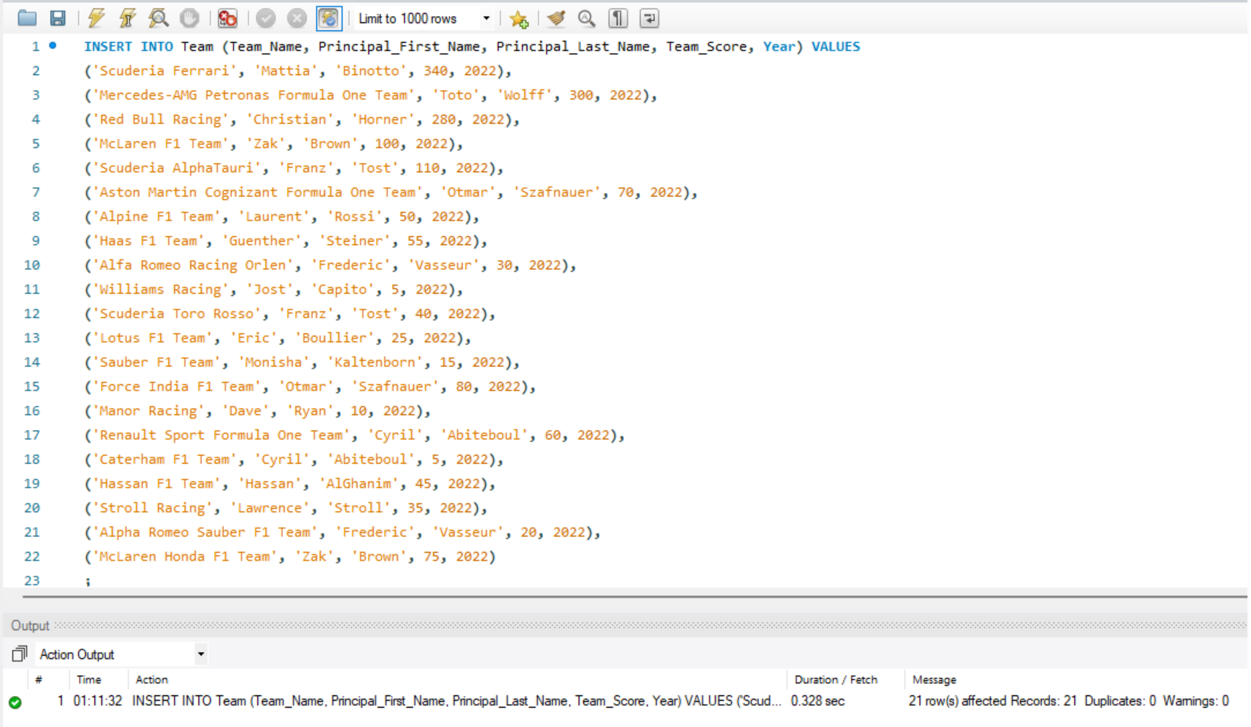
6) INSERT into table Result:



```sql
INSERT INTO Result (Result_ID, Position, Points, Driver_ID, Team_Name, Year) VALUES
  ('Monza-Race-Result1', 1, 25, 1, 'Scuderia Ferrari', 2022),
  ('Monza-Race-Result2', 2, 18, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('Silverstone-Race-Result1', 1, 25, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('Silverstone-Race-Result2', 2, 18, 3, 'Red Bull Racing', 2022),
  ('Spa-Francorchamps-Race-Result1', 1, 25, 3, 'Red Bull Racing', 2022),
  ('Spa-Francorchamps-Race-Result2', 2, 18, 1, 'Scuderia Ferrari', 2022),
  ('Suzuka-Race-Result1', 1, 25, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('Suzuka-Race-Result2', 2, 18, 3, 'Red Bull Racing', 2022),
  ('CircuitAmericas-Race-Result1', 1, 25, 3, 'Red Bull Racing', 2022),
  ('CircuitAmericas-Race-Result2', 2, 18, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('Hungaroring-Race-Result1', 1, 25, 1, 'Scuderia Ferrari', 2022),
  ('Hungaroring-Race-Result2', 2, 18, 3, 'Red Bull Racing', 2022),
  ('Interlagos-Race-Result1', 1, 25, 3, 'Red Bull Racing', 2022),
  ('Interlagos-Race-Result2', 2, 18, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('AlbertPark-Race-Result1', 1, 25, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('AlbertPark-Race-Result2', 2, 18, 1, 'Scuderia Ferrari', 2022),
  ('Monaco-Race-Result1', 1, 25, 1, 'Scuderia Ferrari', 2022),
  ('Monaco-Race-Result2', 2, 18, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('Nurburgring-Race-Result1', 1, 25, 2, 'Mercedes-AMG Petronas Formula One Team', 2022),
  ('Nurburgring-Race-Result2', 2, 18, 3, 'Red Bull Racing', 2022);
```

```
#  Time      Action                                                                                            Duration / Fetch  Message
1  01:15:25  INSERT INTO Result (Result_ID, Position, Points, Driver_ID, Team_Name, Year) VALUES  ('Monza-Race-Result1', 1, ...  0.031 sec  20 row(s) affected Records: 20  Duplicates: 0  Warnings: 0
```

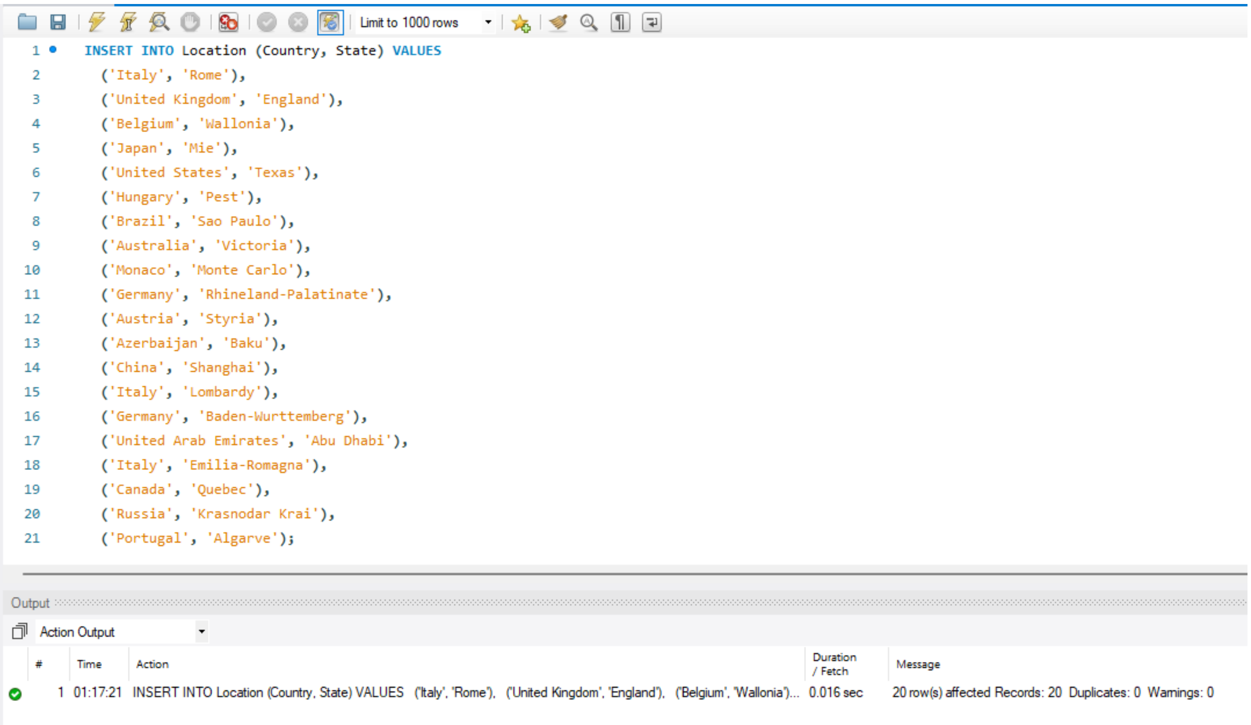
7) INSERT into table Team:



```sql
INSERT INTO Team (Team_Name, Principal_First_Name, Principal_Last_Name, Team_Score, Year) VALUES
('Scuderia Ferrari', 'Mattia', 'Binotto', 340, 2022),
('Mercedes-AMG Petronas Formula One Team', 'Toto', 'Wolff', 300, 2022),
('Red Bull Racing', 'Christian', 'Horner', 280, 2022),
('McLaren F1 Team', 'Zak', 'Brown', 100, 2022),
('Scuderia AlphaTauri', 'Franz', 'Tost', 110, 2022),
('Aston Martin Cognizant Formula One Team', 'Otmar', 'Szafnauer', 70, 2022),
('Alpine F1 Team', 'Laurent', 'Rossi', 50, 2022),
('Haas F1 Team', 'Guenther', 'Steiner', 55, 2022),
('Alfa Romeo Racing Orlen', 'Frederic', 'Vasseur', 30, 2022),
('Williams Racing', 'Jost', 'Capito', 5, 2022),
('Scuderia Toro Rosso', 'Franz', 'Tost', 40, 2022),
('Lotus F1 Team', 'Eric', 'Boullier', 25, 2022),
('Sauber F1 Team', 'Monisha', 'Kaltenborn', 15, 2022),
('Force India F1 Team', 'Otmar', 'Szafnauer', 80, 2022),
('Manor Racing', 'Dave', 'Ryan', 10, 2022),
('Renault Sport Formula One Team', 'Cyril', 'Abiteboul', 60, 2022),
('Caterham F1 Team', 'Cyril', 'Abiteboul', 5, 2022),
('Hassan F1 Team', 'Hassan', 'AlGhanim', 45, 2022),
('Stroll Racing', 'Lawrence', 'Stroll', 35, 2022),
('Alpha Romeo Sauber F1 Team', 'Frederic', 'Vasseur', 20, 2022),
('McLaren Honda F1 Team', 'Zak', 'Brown', 75, 2022)
;
```

```
#  Time      Action                                                                                          Duration / Fetch  Message
1  01:11:32  INSERT INTO Team (Team_Name, Principal_First_Name, Principal_Last_Name, Team_Score, Year) VALUES ('Scud...  0.328 sec  21 row(s) affected Records: 21  Duplicates: 0  Warnings: 0
```

8) INSERT into table Location:

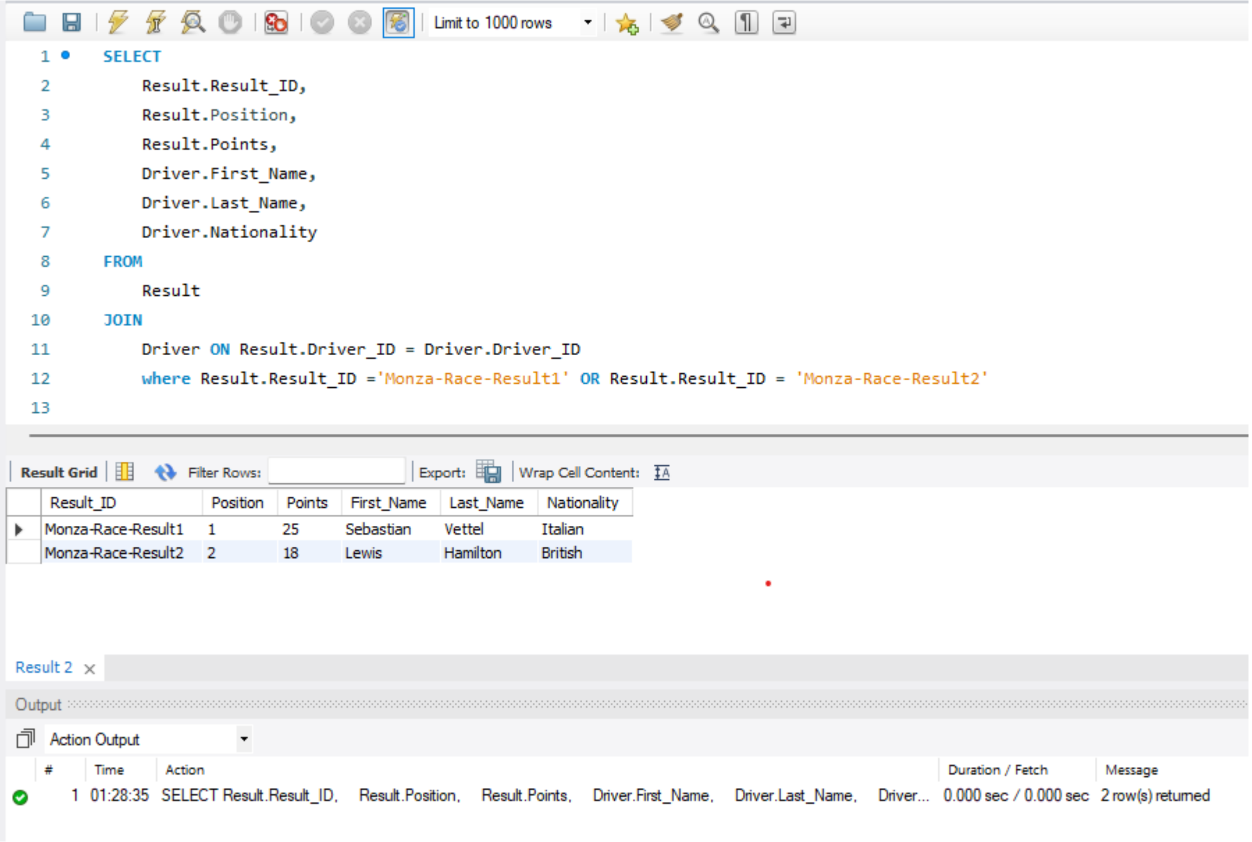


```sql
INSERT INTO Location (Country, State) VALUES
  ('Italy', 'Rome'),
  ('United Kingdom', 'England'),
  ('Belgium', 'Wallonia'),
  ('Japan', 'Mie'),
  ('United States', 'Texas'),
  ('Hungary', 'Pest'),
  ('Brazil', 'Sao Paulo'),
  ('Australia', 'Victoria'),
  ('Monaco', 'Monte Carlo'),
  ('Germany', 'Rhineland-Palatinate'),
  ('Austria', 'Styria'),
  ('Azerbaijan', 'Baku'),
  ('China', 'Shanghai'),
  ('Italy', 'Lombardy'),
  ('Germany', 'Baden-Wurttemberg'),
  ('United Arab Emirates', 'Abu Dhabi'),
  ('Italy', 'Emilia-Romagna'),
  ('Canada', 'Quebec'),
  ('Russia', 'Krasnodar Krai'),
  ('Portugal', 'Algarve');
```

	#	Time	Action	Duration / Fetch	Message
✓	1	01:17:21	INSERT INTO Location (Country, State) VALUES ('Italy', 'Rome'), ('United Kingdom', 'England'), ('Belgium', 'Wallonia')...	0.016 sec	20 row(s) affected Records: 20 Duplicates: 0 Warnings: 0

- **SELECT Queries**

1) Selecting Result using JOIN query on Driver

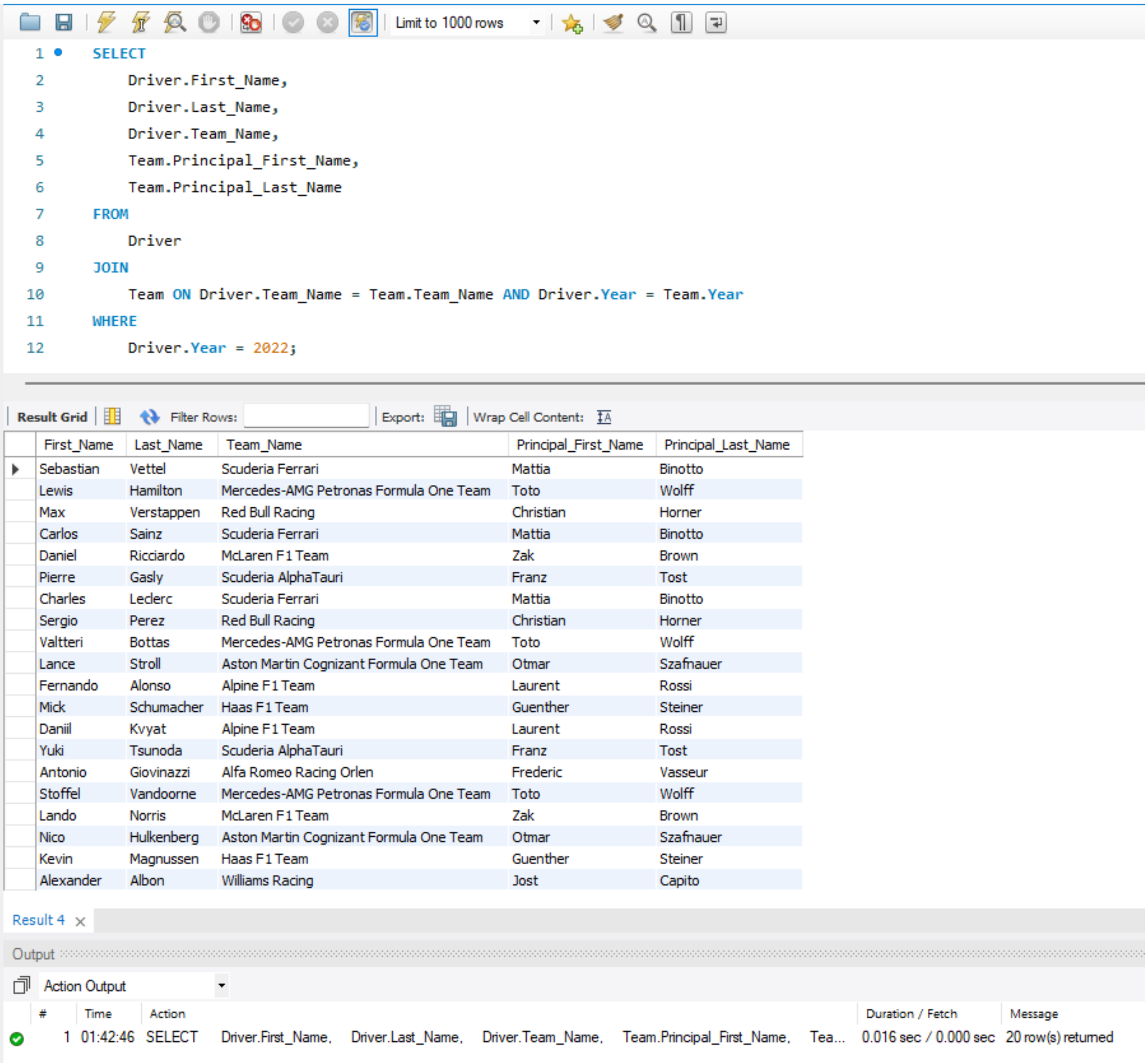


```
SELECT
    Result.Result_ID,
    Result.Position,
    Result.Points,
    Driver.First_Name,
    Driver.Last_Name,
    Driver.Nationality
FROM
    Result
JOIN
    Driver ON Result.Driver_ID = Driver.Driver_ID
    where Result.Result_ID ='Monza-Race-Result1' OR Result.Result_ID = 'Monza-Race-Result2'
```

Result_ID	Position	Points	First_Name	Last_Name	Nationality
Monza-Race-Result1	1	25	Sebastian	Vettel	Italian
Monza-Race-Result2	2	18	Lewis	Hamilton	British

```
   #  Time      Action                                                                                             Duration / Fetch       Message
✓  1  01:28:35  SELECT Result.Result_ID,  Result.Position,  Result.Points,  Driver.First_Name,  Driver.Last_Name,  Driver...  0.000 sec / 0.000 sec  2 row(s) returned
```

2) Selecting Driver and using JOIN query on Team and WHERE clause for Year:

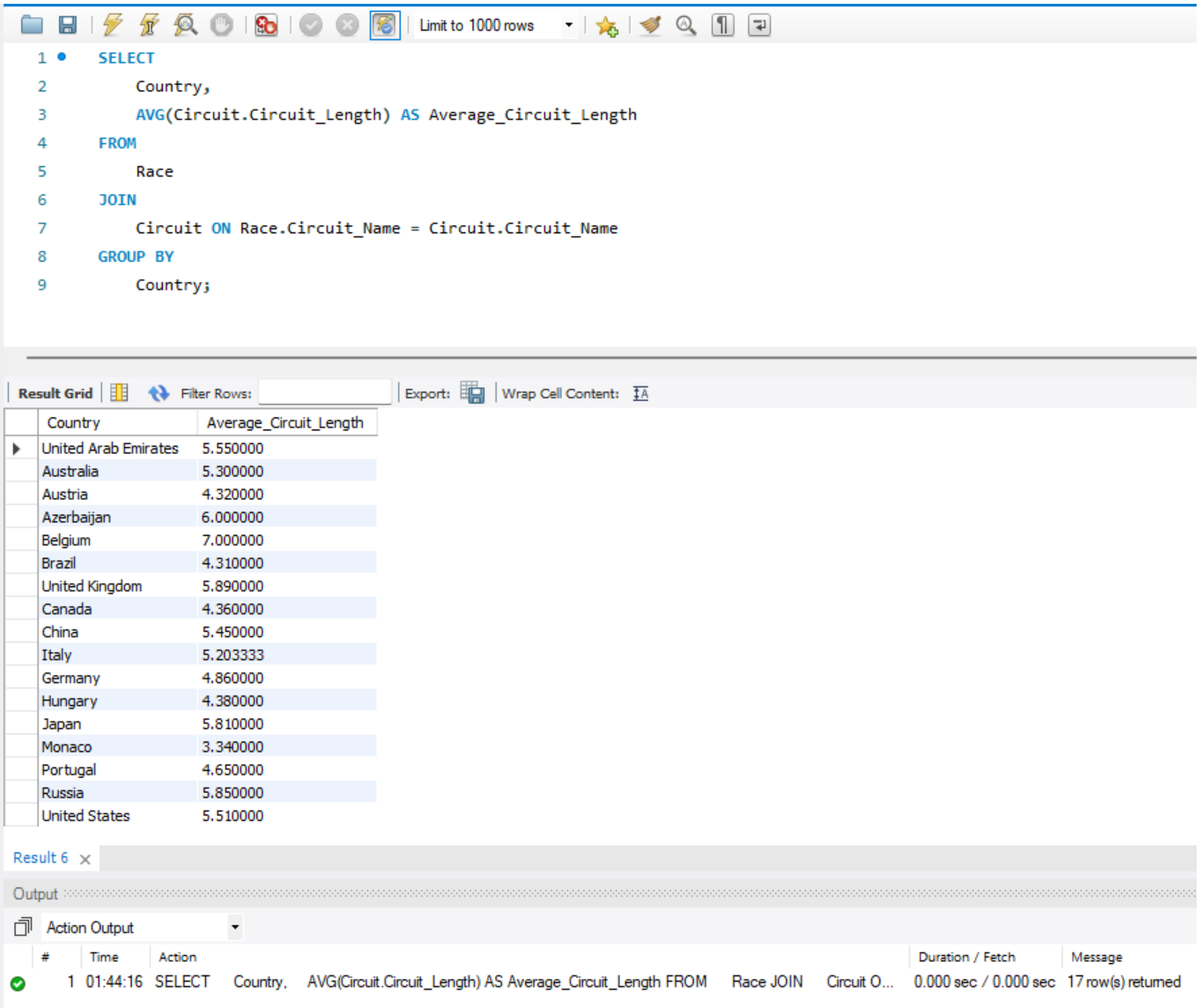


```
SELECT
    Driver.First_Name,
    Driver.Last_Name,
    Driver.Team_Name,
    Team.Principal_First_Name,
    Team.Principal_Last_Name
FROM
    Driver
JOIN
    Team ON Driver.Team_Name = Team.Team_Name AND Driver.Year = Team.Year
WHERE
    Driver.Year = 2022;
```

First_Name	Last_Name	Team_Name	Principal_First_Name	Principal_Last_Name
Sebastian	Vettel	Scuderia Ferrari	Mattia	Binotto
Lewis	Hamilton	Mercedes-AMG Petronas Formula One Team	Toto	Wolff
Max	Verstappen	Red Bull Racing	Christian	Horner
Carlos	Sainz	Scuderia Ferrari	Mattia	Binotto
Daniel	Ricciardo	McLaren F1 Team	Zak	Brown
Pierre	Gasly	Scuderia AlphaTauri	Franz	Tost
Charles	Leclerc	Scuderia Ferrari	Mattia	Binotto
Sergio	Perez	Red Bull Racing	Christian	Horner
Valtteri	Bottas	Mercedes-AMG Petronas Formula One Team	Toto	Wolff
Lance	Stroll	Aston Martin Cognizant Formula One Team	Otmar	Szafnauer
Fernando	Alonso	Alpine F1 Team	Laurent	Rossi
Mick	Schumacher	Haas F1 Team	Guenther	Steiner
Daniil	Kvyat	Alpine F1 Team	Laurent	Rossi
Yuki	Tsunoda	Scuderia AlphaTauri	Franz	Tost
Antonio	Giovinazzi	Alfa Romeo Racing Orlen	Frederic	Vasseur
Stoffel	Vandoorne	Mercedes-AMG Petronas Formula One Team	Toto	Wolff
Lando	Norris	McLaren F1 Team	Zak	Brown
Nico	Hulkenberg	Aston Martin Cognizant Formula One Team	Otmar	Szafnauer
Kevin	Magnussen	Haas F1 Team	Guenther	Steiner
Alexander	Albon	Williams Racing	Jost	Capito

```
   #  Time      Action                                                                                      Duration / Fetch       Message
✓  1  01:42:46  SELECT  Driver.First_Name,  Driver.Last_Name,  Driver.Team_Name,  Team.Principal_First_Name,  Tea...  0.016 sec / 0.000 sec  20 row(s) returned
```

3) Selecting Race and using JOIN query on Circuit and Grouping by Country

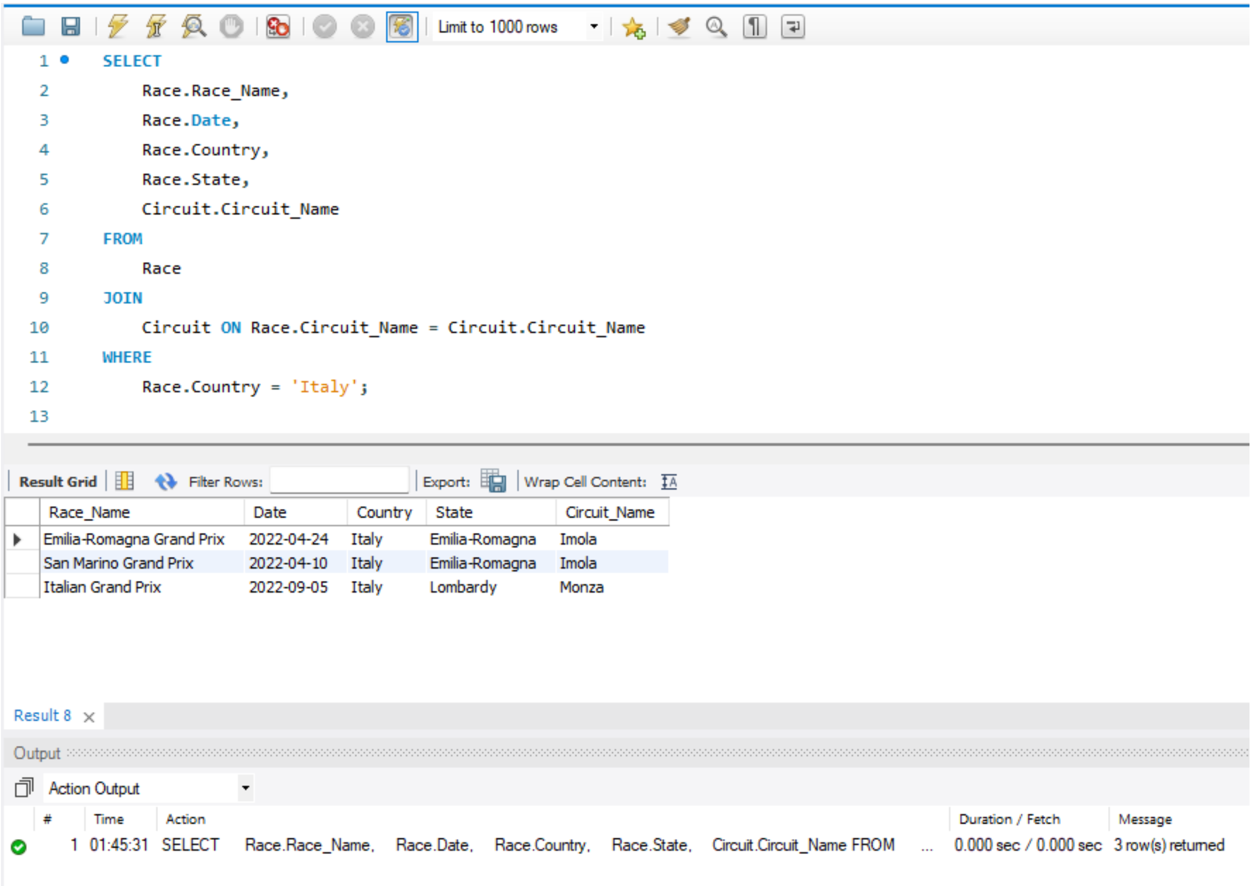


```
SELECT
    Country,
    AVG(Circuit.Circuit_Length) AS Average_Circuit_Length
FROM
    Race
JOIN
    Circuit ON Race.Circuit_Name = Circuit.Circuit_Name
GROUP BY
    Country;
```

Country	Average_Circuit_Length
United Arab Emirates	5.550000
Australia	5.300000
Austria	4.320000
Azerbaijan	6.000000
Belgium	7.000000
Brazil	4.310000
United Kingdom	5.890000
Canada	4.360000
China	5.450000
Italy	5.203333
Germany	4.860000
Hungary	4.380000
Japan	5.810000
Monaco	3.340000
Portugal	4.650000
Russia	5.850000
United States	5.510000

#	Time	Action	Duration / Fetch	Message
1	01:44:16	SELECT Country, AVG(Circuit.Circuit_Length) AS Average_Circuit_Length FROM Race JOIN Circuit O...	0.000 sec / 0.000 sec	17 row(s) returned

4) Selecting from Race and using JOIN query on Circuit and WHERE clause for Country:

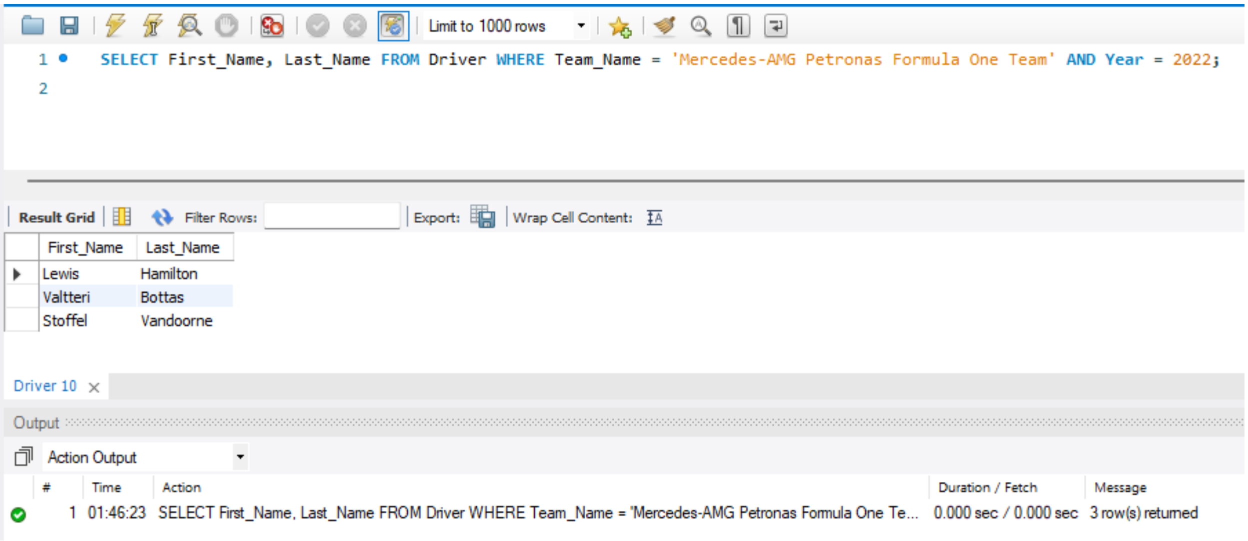


```sql
SELECT
    Race.Race_Name,
    Race.Date,
    Race.Country,
    Race.State,
    Circuit.Circuit_Name
FROM
    Race
JOIN
    Circuit ON Race.Circuit_Name = Circuit.Circuit_Name
WHERE
    Race.Country = 'Italy';
```

Race_Name	Date	Country	State	Circuit_Name
Emilia-Romagna Grand Prix	2022-04-24	Italy	Emilia-Romagna	Imola
San Marino Grand Prix	2022-04-10	Italy	Emilia-Romagna	Imola
Italian Grand Prix	2022-09-05	Italy	Lombardy	Monza

```
#   Time      Action                                                                                        Duration / Fetch      Message
1   01:45:31  SELECT   Race.Race_Name,   Race.Date,   Race.Country,   Race.State,   Circuit.Circuit_Name FROM   ...   0.000 sec / 0.000 sec   3 row(s) returned
```

5) Selecting from Driver and using WHERE clause for Team Name and Year:

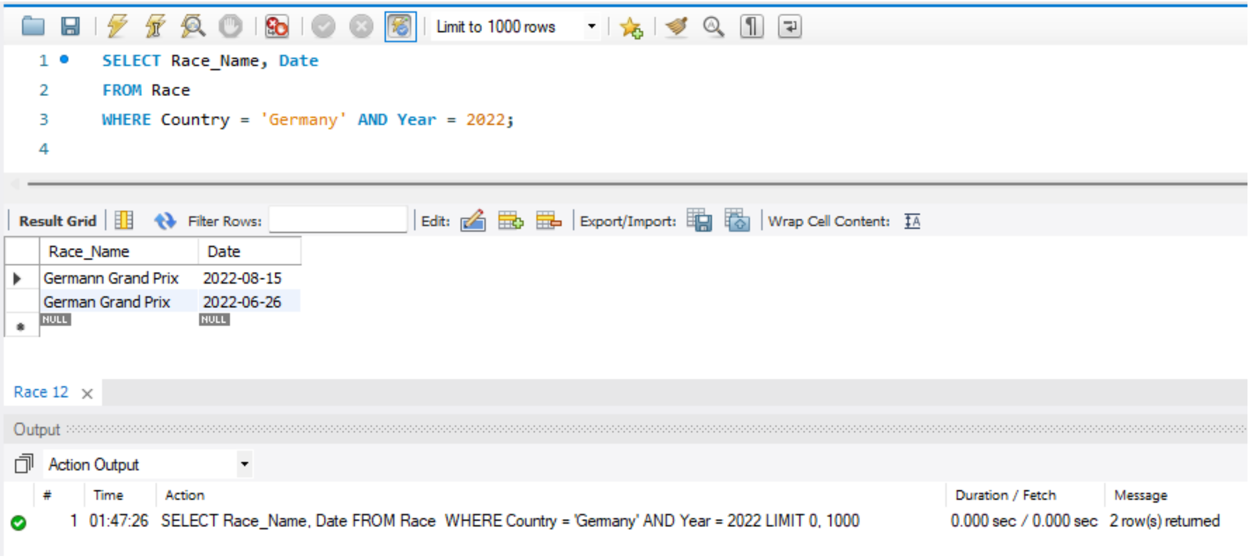


```sql
SELECT First_Name, Last_Name FROM Driver WHERE Team_Name = 'Mercedes-AMG Petronas Formula One Team' AND Year = 2022;
```

First_Name	Last_Name
Lewis	Hamilton
Valtteri	Bottas
Stoffel	Vandoorne

```
#   Time      Action                                                                                   Duration / Fetch        Message
1   01:46:23  SELECT First_Name, Last_Name FROM Driver WHERE Team_Name = 'Mercedes-AMG Petronas Formula One Te...   0.000 sec / 0.000 sec   3 row(s) returned
```

6) Selecting from Race and using WHERE clause for Country and Year:

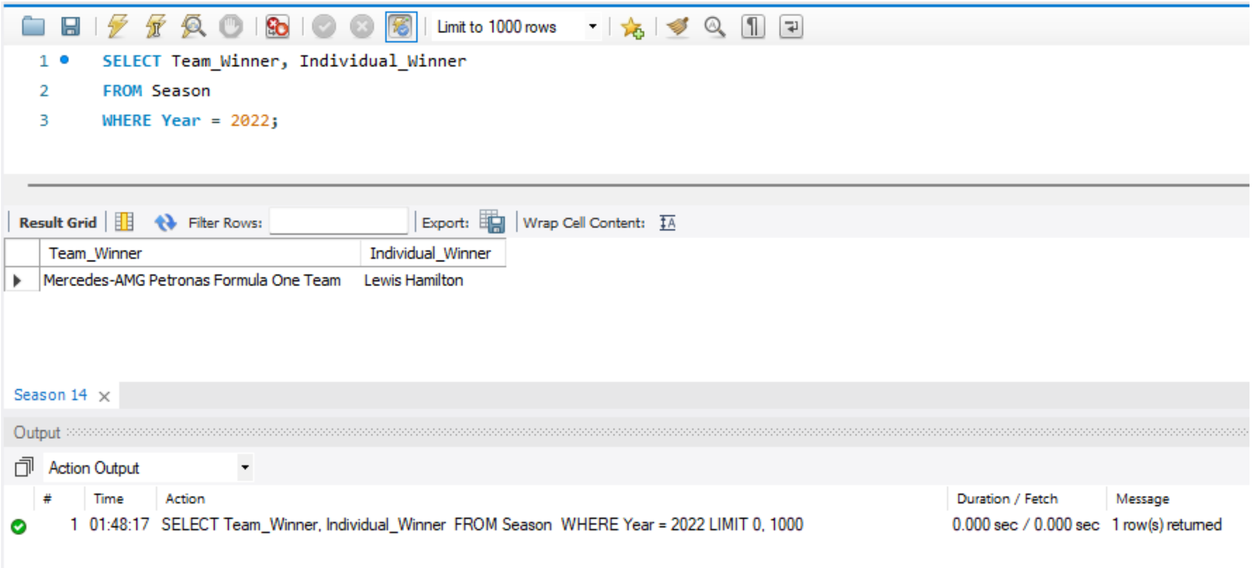


```sql
SELECT Race_Name, Date
FROM Race
WHERE Country = 'Germany' AND Year = 2022;
```

Race_Name	Date
Germann Grand Prix	2022-08-15
German Grand Prix	2022-06-26
NULL	NULL

```
#  Time      Action                                                                              Duration / Fetch       Message
1  01:47:26  SELECT Race_Name, Date FROM Race  WHERE Country = 'Germany' AND Year = 2022 LIMIT 0, 1000   0.000 sec / 0.000 sec  2 row(s) returned
```

7) Selecting from Season and using WHERE clause for Year:

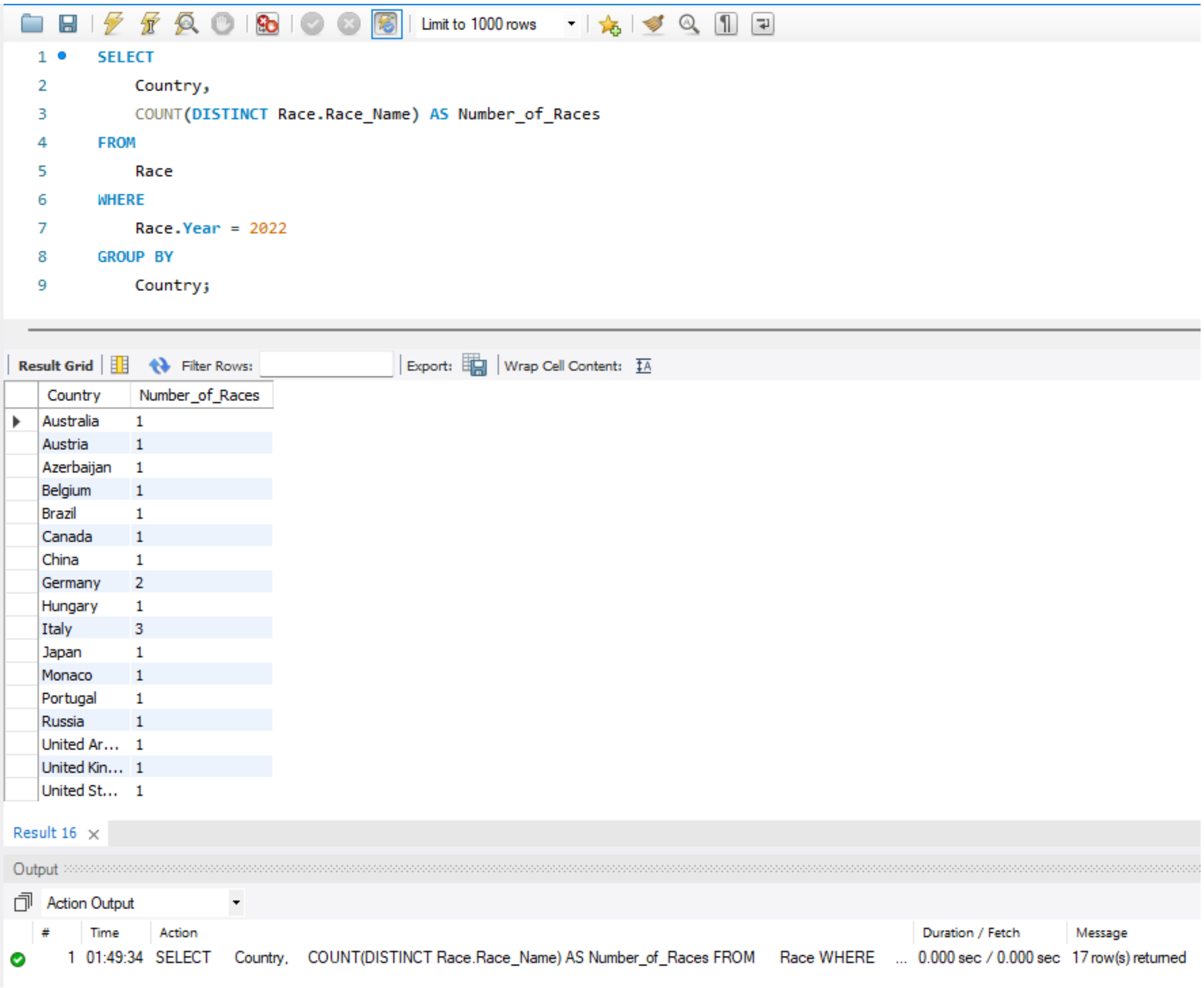


```sql
SELECT Team_Winner, Individual_Winner
FROM Season
WHERE Year = 2022;
```

Team_Winner	Individual_Winner
Mercedes-AMG Petronas Formula One Team	Lewis Hamilton

```
#  Time      Action                                                                           Duration / Fetch       Message
1  01:48:17  SELECT Team_Winner, Individual_Winner  FROM Season  WHERE Year = 2022 LIMIT 0, 1000   0.000 sec / 0.000 sec  1 row(s) returned
```

8) Selecting from Race and using WHERE clause for Year and GROUP BY for Country:

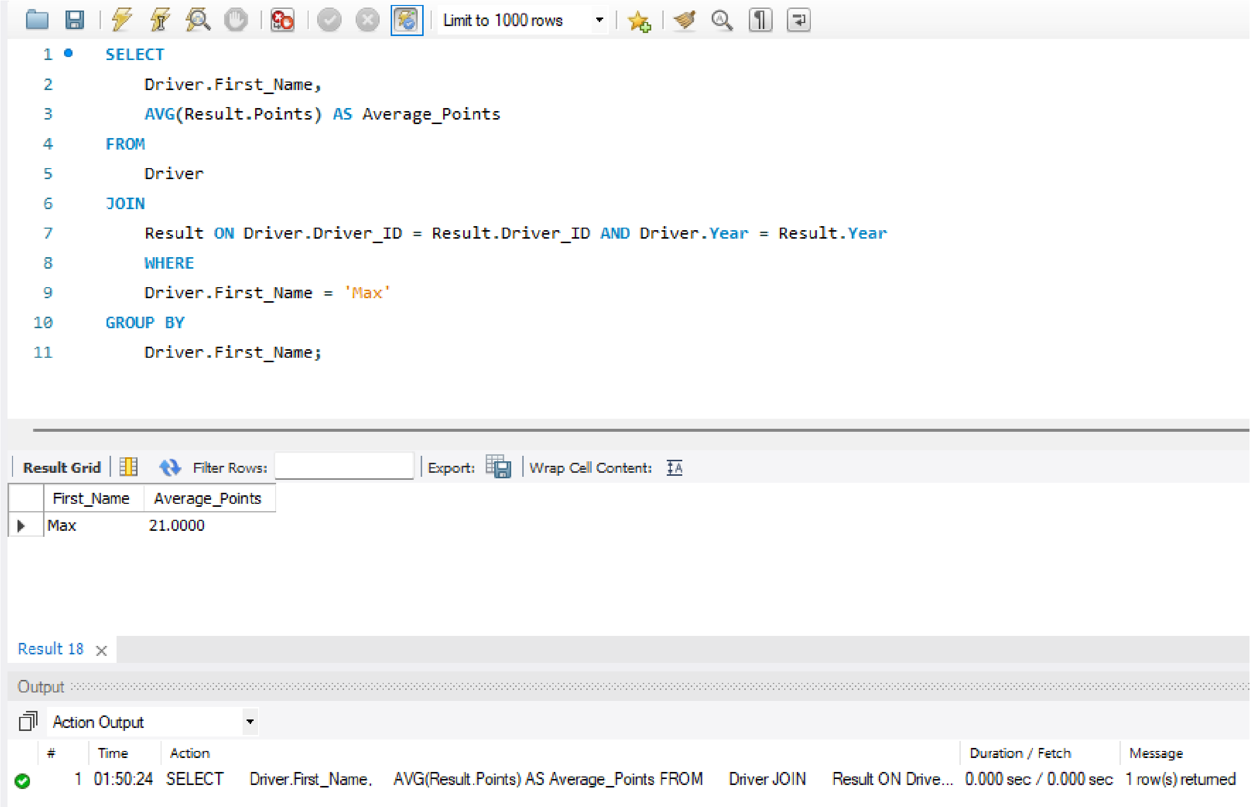


```sql
SELECT
    Country,
    COUNT(DISTINCT Race.Race_Name) AS Number_of_Races
FROM
    Race
WHERE
    Race.Year = 2022
GROUP BY
    Country;
```

Country	Number_of_Races
Australia	1
Austria	1
Azerbaijan	1
Belgium	1
Brazil	1
Canada	1
China	1
Germany	2
Hungary	1
Italy	3
Japan	1
Monaco	1
Portugal	1
Russia	1
United Ar...	1
United Kin...	1
United St...	1

#	Time	Action	Duration / Fetch	Message
1	01:49:34	SELECT Country, COUNT(DISTINCT Race.Race_Name) AS Number_of_Races FROM Race WHERE ...	0.000 sec / 0.000 sec	17 row(s) returned

9) Selecting from Driver for average point and using JOIN query on Result, WHERE clause for Driver Name and GROUP BY for First Name:

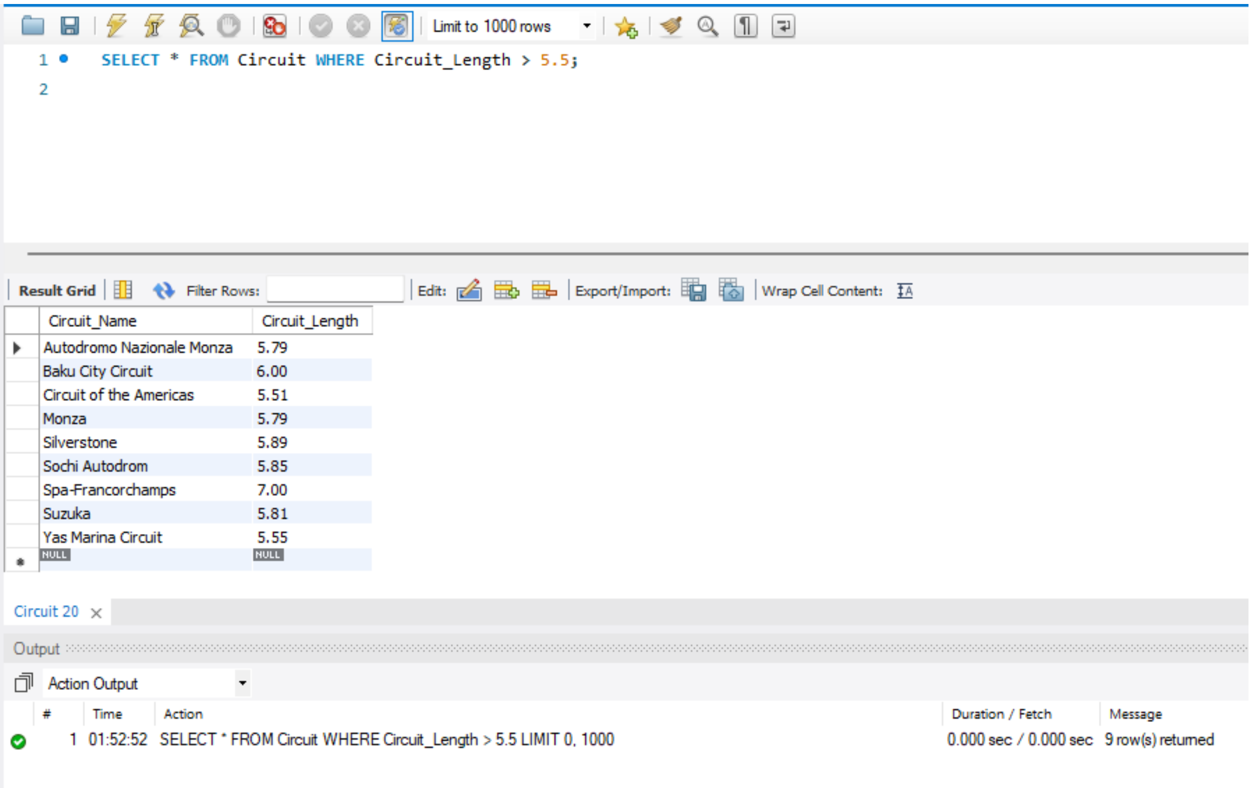


```
SELECT
    Driver.First_Name,
    AVG(Result.Points) AS Average_Points
FROM
    Driver
JOIN
    Result ON Driver.Driver_ID = Result.Driver_ID AND Driver.Year = Result.Year
    WHERE
    Driver.First_Name = 'Max'
GROUP BY
    Driver.First_Name;
```

First_Name	Average_Points
Max	21.0000

```
    #   Time      Action                                                                                  Duration / Fetch        Message
    1   01:50:24  SELECT    Driver.First_Name,    AVG(Result.Points) AS Average_Points FROM    Driver JOIN    Result ON Drive...  0.000 sec / 0.000 sec   1 row(s) returned
```

10) Selecting from Circuit using WHERE clause for Circuit length:

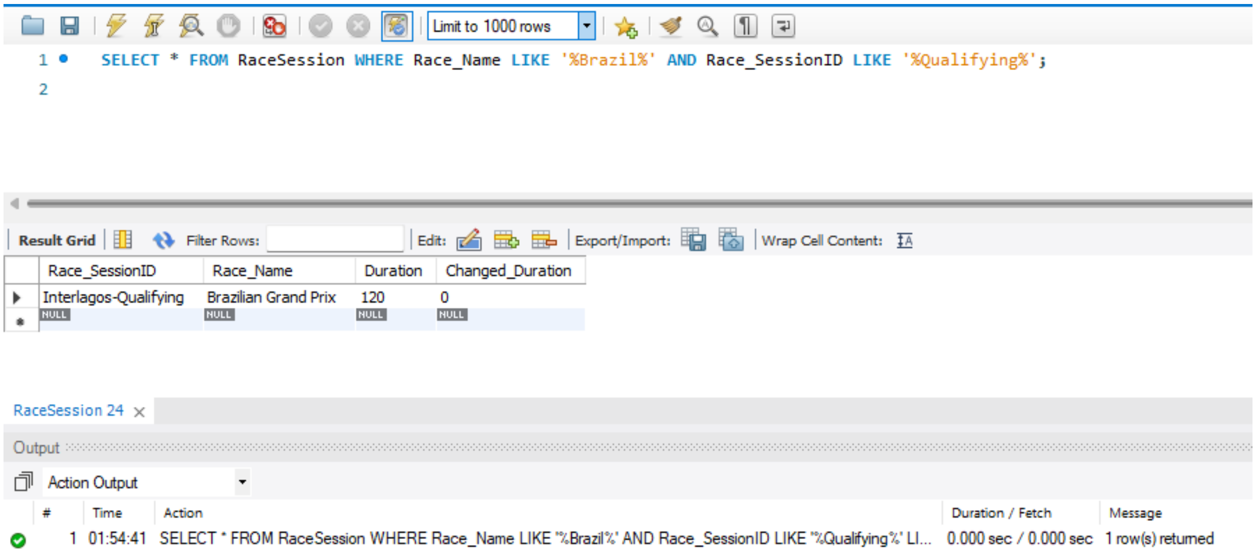


```sql
SELECT * FROM Circuit WHERE Circuit_Length > 5.5;
```

Circuit_Name	Circuit_Length
Autodromo Nazionale Monza	5.79
Baku City Circuit	6.00
Circuit of the Americas	5.51
Monza	5.79
Silverstone	5.89
Sochi Autodrom	5.85
Spa-Francorchamps	7.00
Suzuka	5.81
Yas Marina Circuit	5.55
NULL	NULL

```
#   Time      Action                                                        Duration / Fetch        Message
1   01:52:52  SELECT * FROM Circuit WHERE Circuit_Length > 5.5 LIMIT 0, 1000  0.000 sec / 0.000 sec  9 row(s) returned
```

11) Selecting from RaceSession, using WHERE clause on Race Name and LIKE operator for Specific Race Name and Race Session ID:

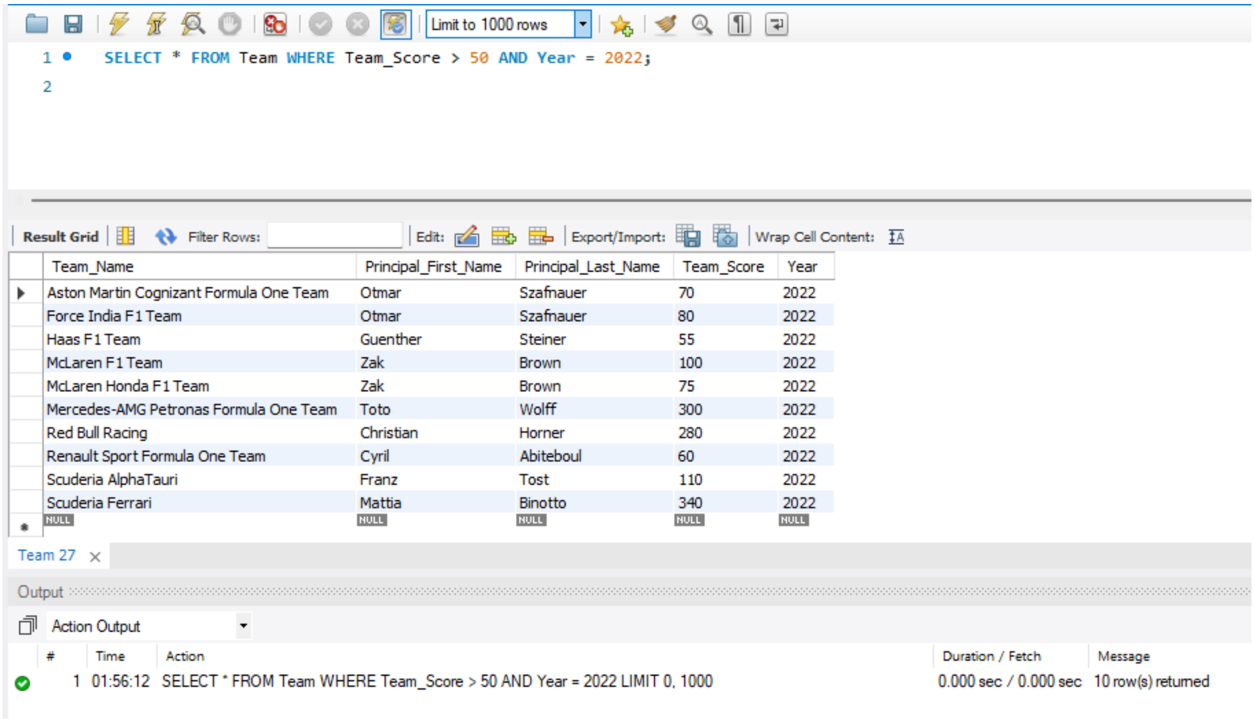


```sql
SELECT * FROM RaceSession WHERE Race_Name LIKE '%Brazil%' AND Race_SessionID LIKE '%Qualifying%';
```

Race_SessionID	Race_Name	Duration	Changed_Duration
Interlagos-Qualifying	Brazilian Grand Prix	120	0
NULL	NULL	NULL	NULL

```
#   Time      Action                                                                          Duration / Fetch        Message
1   01:54:41  SELECT * FROM RaceSession WHERE Race_Name LIKE '%Brazil%' AND Race_SessionID LIKE '%Qualifying%' LI...  0.000 sec / 0.000 sec  1 row(s) returned
```

12) Selecting from Team and WHERE clause for Team Score and Year:

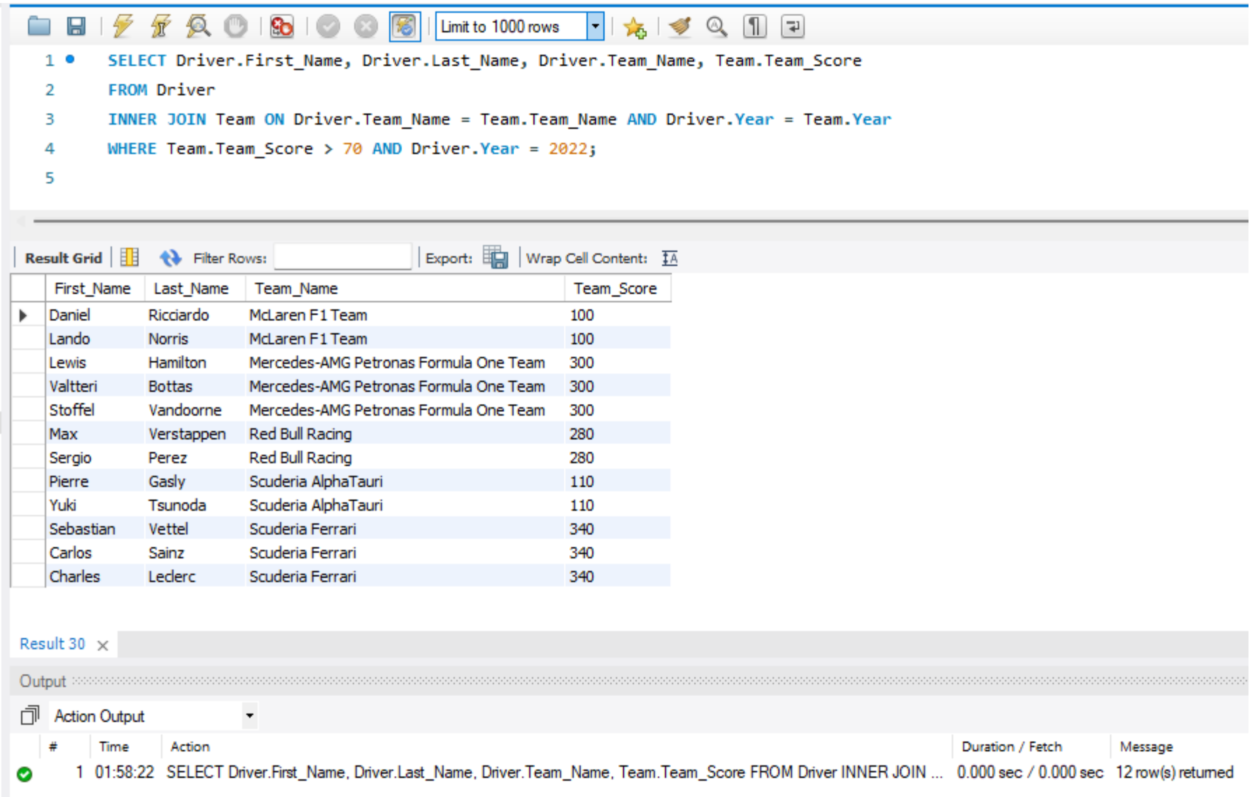


```sql
SELECT * FROM Team WHERE Team_Score > 50 AND Year = 2022;
```

Team_Name	Principal_First_Name	Principal_Last_Name	Team_Score	Year
Aston Martin Cognizant Formula One Team	Otmar	Szafnauer	70	2022
Force India F1 Team	Otmar	Szafnauer	80	2022
Haas F1 Team	Guenther	Steiner	55	2022
McLaren F1 Team	Zak	Brown	100	2022
McLaren Honda F1 Team	Zak	Brown	75	2022
Mercedes-AMG Petronas Formula One Team	Toto	Wolff	300	2022
Red Bull Racing	Christian	Horner	280	2022
Renault Sport Formula One Team	Cyril	Abiteboul	60	2022
Scuderia AlphaTauri	Franz	Tost	110	2022
Scuderia Ferrari	Mattia	Binotto	340	2022
NULL	NULL	NULL	NULL	NULL

#	Time	Action	Duration / Fetch	Message
1	01:56:12	SELECT * FROM Team WHERE Team_Score > 50 AND Year = 2022 LIMIT 0, 1000	0.000 sec / 0.000 sec	10 row(s) returned

13) Selecting Driver and INNER JOIN query on Team and WHERE clause Team Score with the Year:

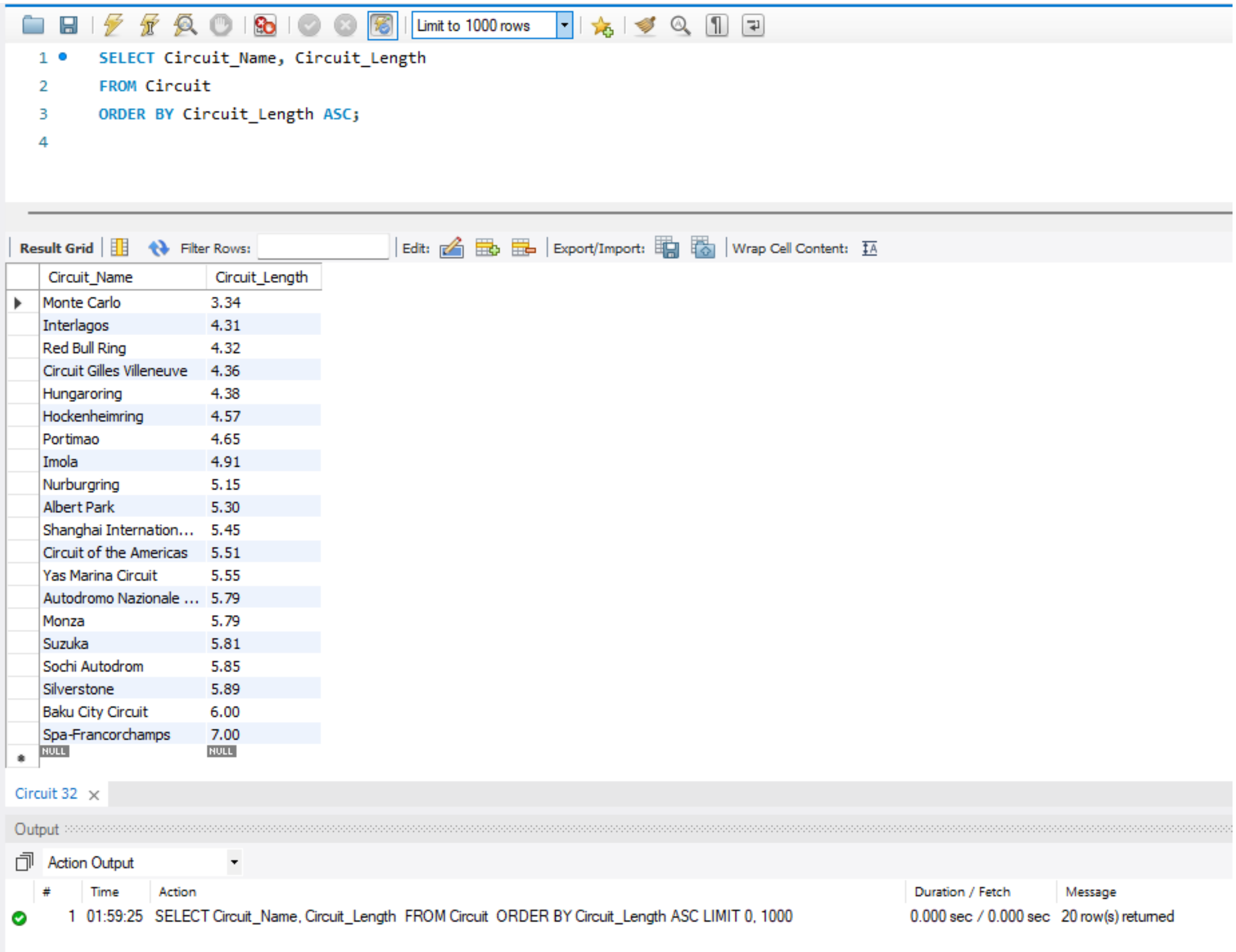


```sql
SELECT Driver.First_Name, Driver.Last_Name, Driver.Team_Name, Team.Team_Score
FROM Driver
INNER JOIN Team ON Driver.Team_Name = Team.Team_Name AND Driver.Year = Team.Year
WHERE Team.Team_Score > 70 AND Driver.Year = 2022;
```

First_Name	Last_Name	Team_Name	Team_Score
Daniel	Ricciardo	McLaren F1 Team	100
Lando	Norris	McLaren F1 Team	100
Lewis	Hamilton	Mercedes-AMG Petronas Formula One Team	300
Valtteri	Bottas	Mercedes-AMG Petronas Formula One Team	300
Stoffel	Vandoorne	Mercedes-AMG Petronas Formula One Team	300
Max	Verstappen	Red Bull Racing	280
Sergio	Perez	Red Bull Racing	280
Pierre	Gasly	Scuderia AlphaTauri	110
Yuki	Tsunoda	Scuderia AlphaTauri	110
Sebastian	Vettel	Scuderia Ferrari	340
Carlos	Sainz	Scuderia Ferrari	340
Charles	Leclerc	Scuderia Ferrari	340

#	Time	Action	Duration / Fetch	Message
1	01:58:22	SELECT Driver.First_Name, Driver.Last_Name, Driver.Team_Name, Team.Team_Score FROM Driver INNER JOIN ...	0.000 sec / 0.000 sec	12 row(s) returned

14) Selecting from Circuit and using ORDER BY query to sort the length of the circuit:



```
SELECT Circuit_Name, Circuit_Length
FROM Circuit
ORDER BY Circuit_Length ASC;
```

Circuit_Name	Circuit_Length
Monte Carlo	3.34
Interlagos	4.31
Red Bull Ring	4.32
Circuit Gilles Villeneuve	4.36
Hungaroring	4.38
Hockenheimring	4.57
Portimao	4.65
Imola	4.91
Nurburgring	5.15
Albert Park	5.30
Shanghai Internation...	5.45
Circuit of the Americas	5.51
Yas Marina Circuit	5.55
Autodromo Nazionale ...	5.79
Monza	5.79
Suzuka	5.81
Sochi Autodrom	5.85
Silverstone	5.89
Baku City Circuit	6.00
Spa-Francorchamps	7.00
NULL	NULL

	#	Time	Action	Duration / Fetch	Message
✓	1	01:59:25	SELECT Circuit_Name, Circuit_Length FROM Circuit ORDER BY Circuit_Length ASC LIMIT 0, 1000	0.000 sec / 0.000 sec	20 row(s) returned

Observation :

In our Formula 1 database project, initially unfamiliar with the topic, we faced confusion understanding attributes and entities. To bridge the knowledge gap, in-house experts explained the sport, guiding us in shaping our database. Building a small database underscored the complexity of design, requiring us to create assumptions, entities, and attributes logically.

The project shifted our individual learning paradigm, fostering teamwork and collaboration. Overcoming the challenge of asking for help, we relied on each other's strengths, breaking from the individual learning model of previous classes.

Unexpectedly, our ER diagram underwent significant changes, such as merging Circuit attributes into Race. This highlighted the interconnected nature of entities and relationships, necessitating adjustments and redesigns.

Choosing Formula 1 Racing as our database topic provided a unique challenge, allowing some to explore a new domain while testing the expertise of others. The project's uncharted territory required initial reliance on experts and later on the collective knowledge acquired during database development.

In conclusion, the Formula 1 database project was a captivating challenge, offering insights into database design, teamwork, and the dynamic world of Formula 1.

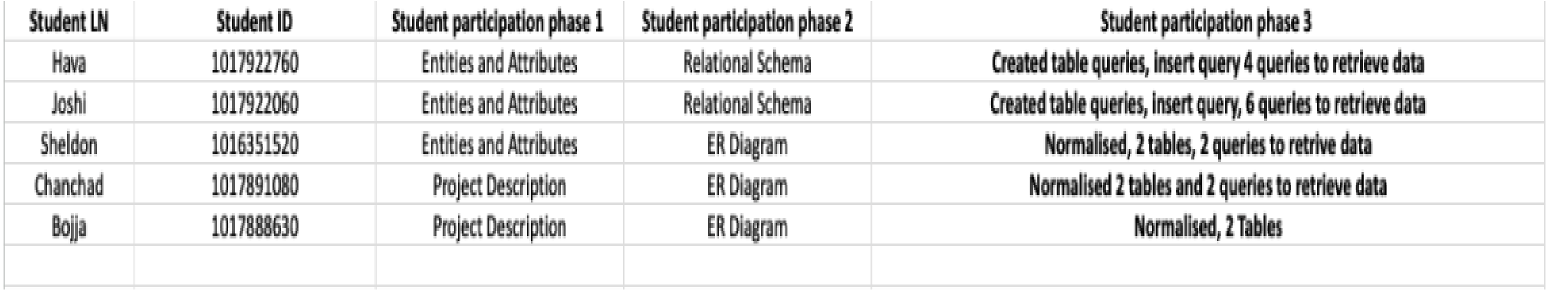
Conclusion :

In summary, our Formula 1 database project was a transformative journey, challenging our understanding of database design and teamwork. The initial hurdles due to Formula 1's unfamiliarity highlighted the importance of domain knowledge. Collaborating with experts fostered a culture of shared learning.

The practical application of concepts emphasized the dynamic nature of design, seen in the evolving ER diagram. Teamwork played a key role, breaking the mold of individual learning and enhancing problem-solving.

A notable shift occurred when we adapted assumptions, integrating Circuit attributes into the Race entity, showcasing real-world database interconnectedness.

This project went beyond academia, offering a platform for learning and collaboration in the dynamic world of Formula 1. It equipped us with valuable skills and a deeper appreciation for teamwork in real-world projects. As we celebrate its completion, we carry forward knowledge, refined teamwork, and newfound enthusiasm for Formula 1.



Student LN	Student ID	Student participation phase 1	Student participation phase 2	Student participation phase 3
Hava	1017922760	Entities and Attributes	Relational Schema	**Created table queries, insert query 4 queries to retrieve data**
Joshi	1017922060	Entities and Attributes	Relational Schema	**Created table queries, insert query, 6 queries to retrieve data**
Sheldon	1016351520	Entities and Attributes	ER Diagram	**Normalised, 2 tables, 2 queries to retrive data**
Chanchad	1017891080	Project Description	ER Diagram	**Normalised 2 tables and 2 queries to retrieve data**
Bojja	1017888630	Project Description	ER Diagram	**Normalised, 2 Tables**