

PROJECT 2

Άσκηση 1:

1) Αρχικά δημιουργήσαμε την μια βάση δεδομένων με όνομα UMDW. Στην συνέχεια τρέξαμε με επιτυχία το έτοιμο script για την δημιουργία του πίνακα raceData και την φόρτωση των δεδομένων.

2) Αρχικά το σχήμα STAR-SCHEMA θα αποτελείται από έναν fact table(FactRacePerformance) και 4 dimension tables(Race,Runner,AgeCategory,dDate). Ο κώδικας που γράψαμε για την δημιουργία των 4 dimension tables είναι ο εξής:

```
CREATE TABLE Race
(RaceID INT PRIMARY KEY,
RaceName VARCHAR(200),
RaceDistance VARCHAR(10),
RaceCountry VARCHAR(3)
);

CREATE TABLE Runner
(RunnerID INT PRIMARY KEY,
RunnerBirthYear INT,
RunnerGender VARCHAR(1),
RunnerCountry VARCHAR(3)
);

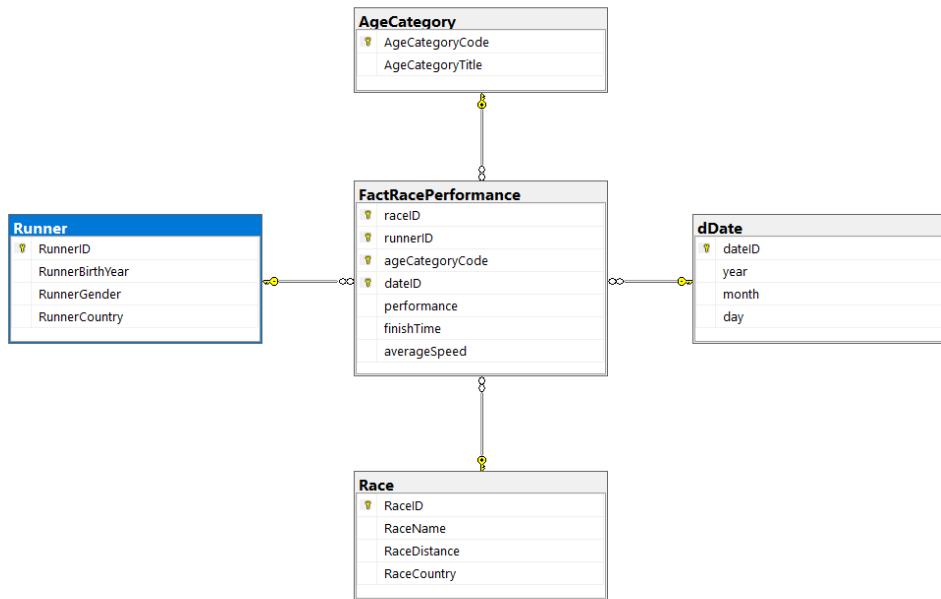
CREATE TABLE AgeCategory |
(AgeCategoryCode VARCHAR(4) PRIMARY KEY,
AgeCategoryTitle VARCHAR(50)
);

CREATE TABLE dDate
(dateID DATE PRIMARY KEY,
year INT,
month INT,
day INT
);
```

Και για τον fact table ο εξής:

```
CREATE TABLE FactRacePerformance
(raceID INT,
runnerID INT,
ageCategoryCode VARCHAR(4),
dateID DATE,
performance VARCHAR(20),
finishTime DECIMAL(4,2),
averageSpeed DECIMAL(5,3),
PRIMARY KEY (raceId,runnerID,ageCategoryCode,dateID),
FOREIGN KEY (raceID) REFERENCES Race(raceID),
FOREIGN KEY (runnerID) REFERENCES Runner(runnerID),
FOREIGN KEY (ageCategoryCode) REFERENCES AgeCategory(ageCategoryCode),
FOREIGN KEY (dateID) REFERENCES dDate(dateID)
);
```

3)Το διάγραμμα που προκύπτει είναι το εξής:



4)

Τέλος οι εντολές που γράψαμε για το για να κάνουμε insert τα κατάλληλα δεδομένα στους πίνακες dimension είναι οι εξής.

```
INSERT INTO Race
SELECT DISTINCT raceID, raceName, raceDistance, racecountry
FROM raceData;

INSERT INTO Runner
SELECT DISTINCT runnerID, runnerBirthYear, runnerGender, runnerCountry
FROM raceData;

INSERT INTO AgeCategory
SELECT DISTINCT ageCategoryCode, ageCategoryTitle
FROM raceData;

INSERT INTO dDate
SELECT DISTINCT raceDate, YEAR(raceDate), MONTH(raceDate), DAY(raceDate)
FROM raceData;
```

και για τον πίνακα fact:

```
INSERT INTO FactRacePerformance
SELECT raceID, runnerID, ageCategoryCode, raceDate, performance, finishTime, averageSpeed
FROM raceData;
```

ΑΣΚΗΣΗ 2:

1)

```
SELECT r.raceCountry, d.year, COUNT(DISTINCT f.raceID) AS totalRaces
FROM FactRacePerformance f
JOIN Race r ON f.raceID = r.raceID
JOIN dDate d ON f.dateID = d.dateID
GROUP BY r.raceCountry, d.year
ORDER BY r.raceCountry ASC, d.year ASC;
```

2)

```
SELECT a.AgeCategoryCode, ru.RunnerGender, AVG(f.finishTime)
FROM FactRacePerformance f
join AgeCategory a on f.ageCategoryCode=a.AgeCategoryCode
join Runner ru on f.runnerID=ru.RunnerID
join Race r on f.raceID = r.raceID
where r.RaceDistance='50km'
GROUP BY a.AgeCategoryCode, ru.RunnerGender
ORDER BY a.AgeCategoryCode ASC
```

3)

```
SELECT d.year, D.month, COUNT(DISTINCT r.RunnerID) as GreekRunners
FROM FactRacePerformance f
JOIN Runner r on f.runnerID = r.RunnerID
JOIN dDate d ON f.dateID = d.dateID
WHERE r.RunnerCountry='GRE'
GROUP BY d.year, d.month
ORDER BY d.year DESC, d.month DESC
```

4)

```
SELECT r.RaceCountry, r.RaceDistance, ru.RunnerGender, COUNT(*) AS NumberOfParticipations
FROM FactRacePerformance f
JOIN Race r on f.raceID = r.RaceID
JOIN Runner ru on f.runnerID = ru.RunnerID
GROUP BY CUBE (r.RaceCountry, r.RaceDistance, ru.RunnerGender)
```