**ASSIGNMENT**

Student name: Lê Tiến Đạt | HE151019

Student name: Tạ Văn Tân | HE151110

STUDENT NAME: Khương Văn Hùng | HE153096

STUDENT NAME: Nguyễn Duy Hùng | HE153495

STUDENT NAME: Lưu Hoàng Hải | HE153224

Teacher: Phạm Ngọc Thọ

DBI202 – DATABASE SYSTEM OF LEAGUE OF LEGENDS CHAMPIONS Korea

July 20, 2021

# TABLE OF CONTENTS

## INTRODUCE THE PROBLEM…………………………………………………………………………………………….

### Describe the problem……………………………………………………………………………………………………….

### Management objectives…………………………………………………………………………………………………..

## entity – relationship – erD………………………………………………………………………………………

### difinITION entity – attributE…………………………………………………………………………………………..

### set-up entity – relationship…………………………………………………………………………………………….

## data dictionary………………………………………………………………………………………………………….

### DATABASE AND TABLE……………………………………………………………………………………….………………….

## ENTITY RELATIONSHIP DIAGRAM (ERD)…………………………………………………………………………

### Player………………………………………………………………………………………………………………………………….

### Role……………………………………………………………………………………………………………………………..........

### match………………………………………………………………………………………………………………………………….

### Team…………………………………………………………………………………………………………………………………….

### organize……………………………………………………………………………………………………………………………..

### take care…………………………………………………………………………………………………………………………….

## SQL COMMAND

### QUERY USING ORDER BY………………………………………………………………………………………………………

### QUERY USING INNER JOIN……………………………………………………………………………………………………

### QUERY USING AGGREGATE FUNCTIONS…………………………………………………………………………………

### QUERY USING THE GROUP BY AND HAVING CLAUSES……………………………………………………………

### QUERY THAT USES A SUB-QUERY AS A RELATION………………………………………………………………….

### QUERY THAT USES PARTIAL MATCHING IN THE WHERE CLAUSE…………………………………………….

### QUERY THAT USES A SELF-JOIN……………………………………………………………………………………………..

### STORE PROCEDURE……………………………………………………………………………………………………………….

### trigger……………………………………………………………………………………………………………………………….

# INTRODUCE THE PROBLEM

## Describe the problem

Nowadays, E-sport is more popular and a lot of tournaments are organized in many countries around the world, attract millions of viewers. **League of Legends Champions Korea (LCK)** is the primary competition for League of Legends esports in South Korea. Contested by ten teams, the league runs two seasons per year (Spring and Summer). The LCK has been long considered one of the strongest League of Legends leagues in the world, with the game's World Championship having been won by teams from the league from 2013 through 2017. With the great attraction from the league, we decided to create a database to manage the LCK tournament.

**Format:**

* The tournament has 10 teams, competing 2 season per year (Spring and Summer).
* In regular season, 10 teams compete in a round robin group stage, matches are best of three and top 6 teams qualify for Playoffs.
* In Playoffs, top 2 teams play from the semi-final, the other four start from the quarter-finals.

All matches in Playoffs are best of five.

* The winner of the Spring Season qualify for the Mid-Season Invitational.
* The winner of the Summer Season (seed 1), the team with the most championship points (seed 2), and the winner of the regional qualifier (seed 3) qualify for the World Championship.

**Request:**

* View player information, team information.
* View organize team (Teachnical, referee..).
* View matches, result, point of every team.
* View winning team, MVP.

## Management objectives

* Manage player and team.
* Manage every matches of tournament.
* Manage matches, time, result.
* Manager organize team.

**Important output**

* Information of all player and team.
* Result of every matches in tournament.
* Information of team organize for each match.

# entity – relationship – er

## difinITION entity – attributE

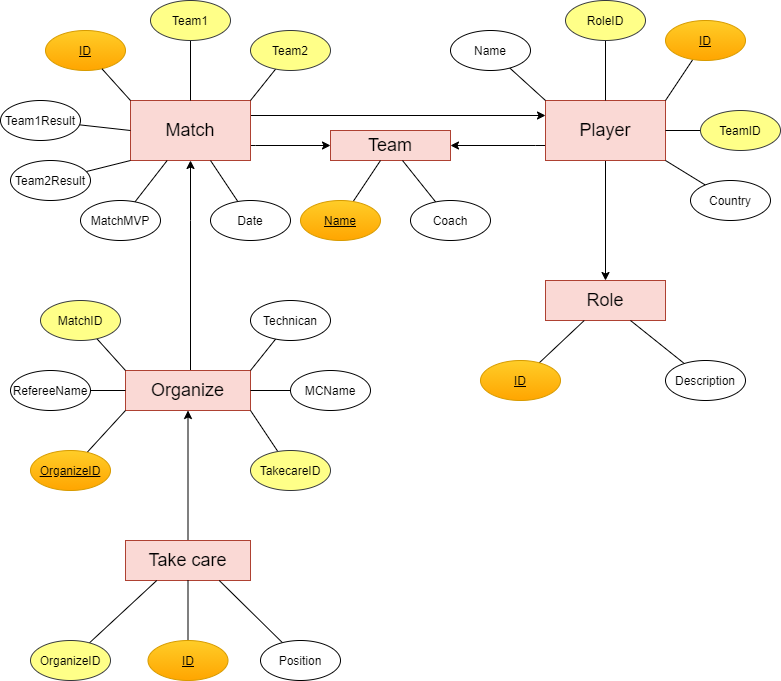
Base on the problem description and management objectives, we can present several entities and attributes of the entity as follow:

* Player: **ID**, **TeamID, RoleID**, Name, Country.
* Role: **ID**, Description.
* Match: **ID**, **Team1, Team2**, Team1Result, Team2Result, MatchMVP, Date.
* Team: **Name**, Coach.
* Organize: **OrganizeID, MatchID, TakecareID,** Technican, RefereeName, MCName
* Take care: **ID, OrganizeID**, Position

## set-up entity – relationship

\* Some symbols used in the model

|  |  |
| --- | --- |
| * Key / identifier attribute | **Attibute** |
| * Attribute description / description | Attribute |
| * Entity | **ENTITY** |
| * Weak entity | **WEAK ENTIRY** |
| * Relationship | *Relationship* |
| * Connectivity (force) = 1 |  |
| * Connectivity = N |  |



## LINK: <https://drive.google.com/file/d/1IkP_4acuC-PdL0Cu-L5UwHSahNEqTqzn/view?usp=sharing>

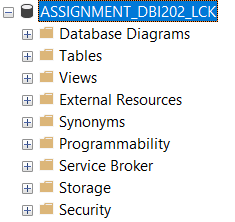
# data dictionary

## database and table

### cREATE DATABASE **ASSIGNMENT\_DBI202\_LCK**

--create database

CREATE DATABASE ASSIGNMENT\_DBI202\_LCK



### **Create table PLayer**

| Column Name | Data Type | Key/ Index/ Constraint |
| --- | --- | --- |
| ID | Nvarchar(30) | Primary key |
| Name | Nvarchar(30) | Not null |
| Role | Nvarchar(10) | Not null,  FOREIGN KEY(RoleID) REFERENCES dbo.Role(ID) |
| Country | Nvarchar(10) | Not null |
| Team | Nvarchar(10) | FOREIGN KEY(TeamID) REFERENCES dbo.Team(ID) |

***Code:***

--create table player

CREATE TABLE Player

(

ID NVARCHAR(15),

[Name] NVARCHAR(30) not null,

RoleID NVARCHAR(10) not null,

Country NVARCHAR(10) not null,

TeamID NVARCHAR(10),

PRIMARY KEY(ID),

FOREIGN KEY(RoleID) REFERENCES dbo.Role(ID),

FOREIGN KEY(TeamID) REFERENCES dbo.Team(ID)

)

***Example:***

| GuardID | Name | Role | Country | TeamID |
| --- | --- | --- | --- | --- |
| 5kid | Park Jeong-hyeon | Bot Laner | KR | KT |
| Arthur | Park Mi-reu | Jungler | KR | HLE |
| Bang | Bae Jun-sik | Bot lane | KR | AF |

### **Create table Role**

| Column Name | Data Type | Key/ Index/ Constraint |
| --- | --- | --- |
| ID | Nvarchar(10) | Primary key |
| Description | Text | Not null |

***Code:***

--create table role

CREATE TABLE [Role]

(

ID NVARCHAR(10) not null,

[Description] TEXT not null

PRIMARY KEY(ID)

)

***Example:***

| ID | Description |
| --- | --- |
| Top Laner |  |
| Jungler |  |
| Mid Laner |  |
| Bot Laner |  |
| Support |  |

### **Create table match**

| Column Name | Data Type | Key/ Index/ Constraint |
| --- | --- | --- |
| ID | Int | Primary key |
| Team 1 | Nvarchar(10) | Not null, FOREIGN KEY(Team1) REFERENCES dbo.Team(ID) |
| Team 2 | Nvarchar(10) | Not null, FOREIGN KEY(Team2) REFERENCES dbo.Team(ID), |
| Team 1 Result | Int | Not null |
| Team 2 Result | Int | Not null |
| Date | Date |  |
| MatchMVP | Nvarchar(15) | Not null, FOREIGN KEY(MatchMVP) REFERENCES dbo.Player(ID) |

***Code:***

CREATE TABLE [Match]

(

ID INT,

Team1 NVARCHAR(10) NOT NULL,

Team2 NVARCHAR(10) NOT NULL,

Team1Result INT,

Team2Result INT,

[Date] DATE,

MatchMVP NVARCHAR(15),

PRIMARY KEY(ID),

FOREIGN KEY(Team1) REFERENCES dbo.Team(ID),

FOREIGN KEY(Team2) REFERENCES dbo.Team(ID),

FOREIGN KEY(MatchMVP) REFERENCES dbo.Player(ID)

)

***Example:***

| ID | Team 1 | Team 2 | Team 1 result | Team 2 result | Date | Match MVP |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | KT | T1 | 0 | 2 | 1/9/2021 | Hoit |
| 2 | NS | DRX | 1 | 2 | 1/7/2021 | Effort |
| 3 | HLE | NS | 1 | 2 | 1/5/2021 | Dove |

### Create TABLE team

| Column Name | Data Type | Key/ Index/ Constraint |
| --- | --- | --- |
| ID | Nvarchar(10) | Primary key |
| Coach | Nvarchar(10) | Not null |

***Code:***

--create table manager team

CREATE TABLE Team

(

ID NVARCHAR(10),

Coach NVARCHAR(10)

PRIMARY KEY(ID)

)

***Example:***

| ID | Coach |
| --- | --- |
| AF | 8571 |
| BRO | 55141 |
| DRX | 87469 |

### create table ORganize

| Column Name | Data Type | Key/ Index/ Constraint |
| --- | --- | --- |
| OrganizeID | Int | Primary key |
| MatchID | Int | FOREIGN KEY(MatchID) REFERENCES dbo.Match(ID) |
| TakecareID | Int |  |
| RefereeName | Nvarchar(20) |  |
| Technican | Nvarchar(20) |  |
| MCName | Nvarchar(20) |  |

***Code:***

--create table register

CREATE TABLE Organize

(

OrganizeID INT,

MatchID INT,

TakecareID INT,

RefereeName NVARCHAR(10),

Technican NVARCHAR(20),

MCName NVARCHAR(30)

PRIMARY KEY(OrganizeID)

FOREIGN KEY(MatchID) REFERENCES dbo.Match(ID),

)

***Example:***

| OrganizeID | MatchID | TakecareID | RefereeName | Technican | MC name |
| --- | --- | --- | --- | --- | --- |
| 1 | 69 | 43 | Melvin | LI 39 11 60 I | Salvatore0 |
| 2 | 83 | 99 | Harvey | OA 35 40 14 S | Larry |
| 3 | 4 | 52 | Malcolm | QM 72 75 59 K | Marshall |

### create table take care

| Column Name | Data Type | Key/ Index/ Constraint |
| --- | --- | --- |
| ID | Int | Primary key |
| Position | Nvarchar(20) |  |
| OrganizeID | Int | FOREIGN KEY(OrganizeID) REFERENCES dbo.Organize(OrganizeID) |

***Code:***

--create table items

CREATE TABLE [Take care]

(

ID INT,

Position NVARCHAR(20),

OrganizeID INT,

PRIMARY KEY(ID),

FOREIGN KEY(OrganizeID) REFERENCES dbo.Organize(OrganizeID)

)

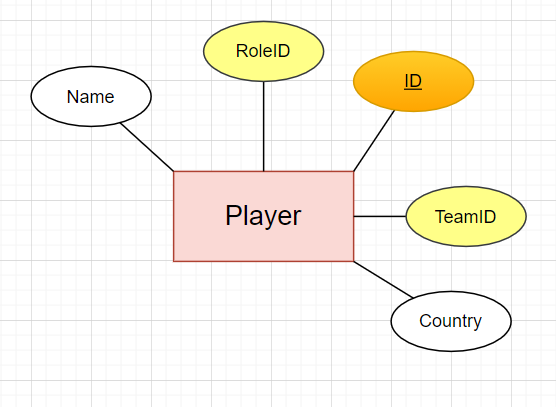
***Example:***

| ID | Position | OrganizeID |
| --- | --- | --- |
| 1 | Technical | 5 |
| 2 | Prepaid Customer | 6 |
| 3 | Prepaid Customer | 17 |

### 

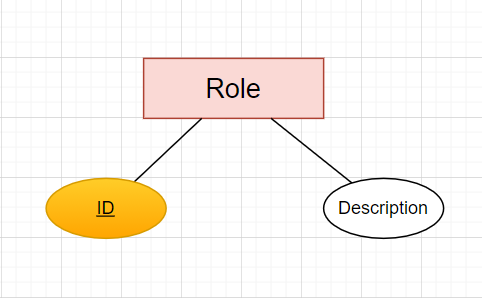
# IV. entity relationship diagram (erd)

## Player



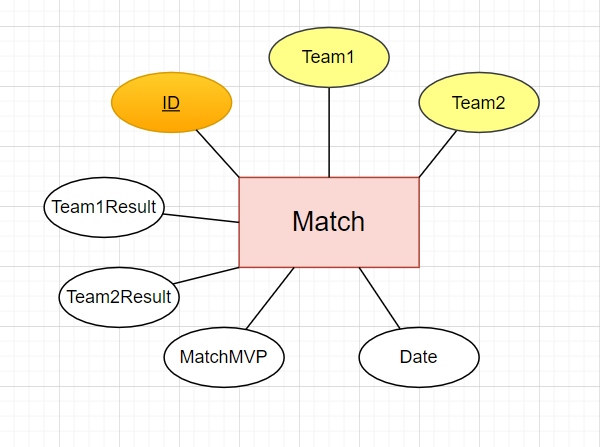
|  |  |
| --- | --- |
| This is the Player entity. Entity has 5 attributes. The ID attribute is the primary key of this entity. Each player has a Name, Role, Team and Country. In which, Role is RoleID, Team is TeamID. |  |

## Role



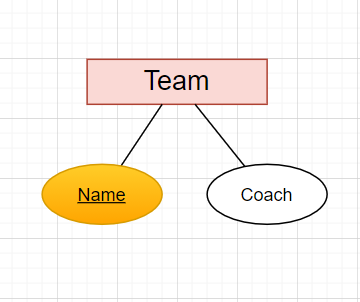
|  |  |
| --- | --- |
| This is the Role entity. Entity has 2 properties. The ID attribute is the primary key of this entity. Each role has a description describing that role. |  |
|  |  |

## Match

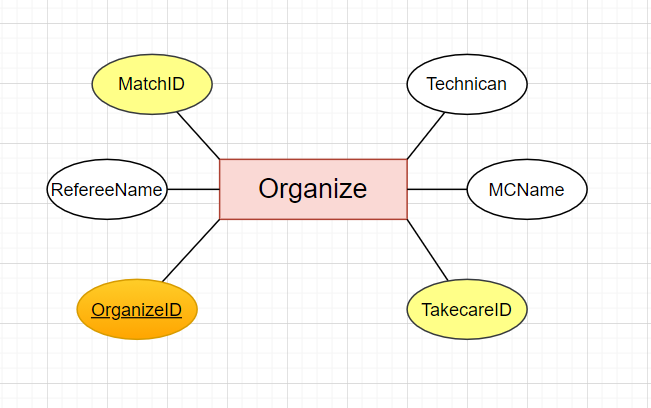


|  |  |
| --- | --- |
| Two teams play against each other called a match. It knows the names of two teams that play against each other like Team1 and Team2. The ID is the primary key to know which match it is and Day indicates what day the match will take place. Finally, Team1Result and Team2Result are the number of games won by the two teams after the match. |  |
|  |  |

## Team



|  |  |
| --- | --- |
| The Team entity has 2 properties. Name is the name of the team and is the primary key. Each team will have a Coach. |  |
| Organize |  |



|  |  |
| --- | --- |
| The Organize entity has 6 properties. Each organization of a match needs technicians, referees, MCs, Take Care. With OrganizeID as the primary key. |  |

|  |  |  |
| --- | --- | --- |
| Take care  |  |  | | --- | --- | | Each Take Care squirrel has a Position. Take Care entity has 3 properties where ID is primary key which is the name of the caregiver and the OrganizeID indicates which organization the caregiver works for. |  |  full diagram |

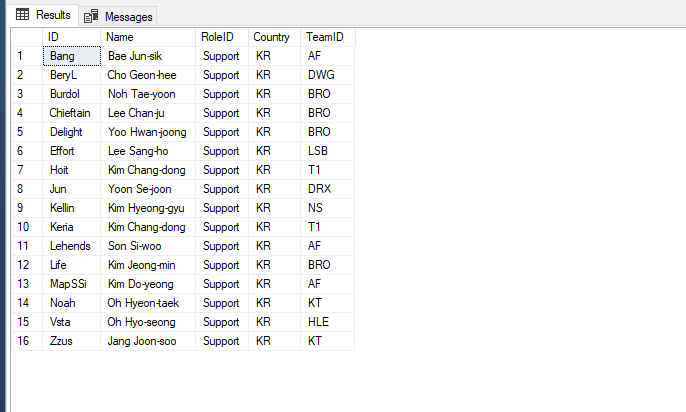
# V. sql command

**\*SETUP QUERY**

-- lay ra cac thong tin bang Player khi RoleID bang support

SELECT \* from Player

## WHERE RoleID = 'Support'



-- Các tuyển thủ đến từ các quốc gia nào?

SELECT DISTINCT Country FROM dbo.Player



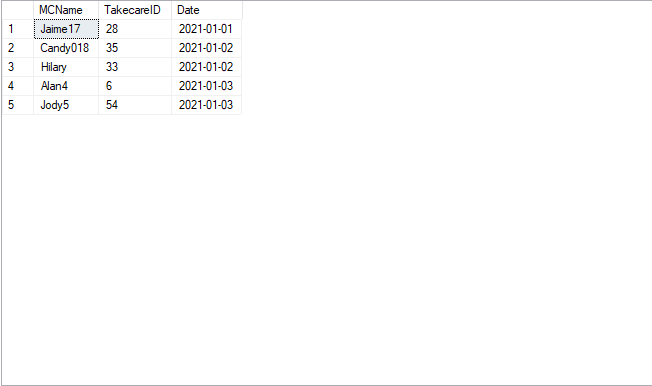
-- lấy ra thông tin takecare và MC trong các trận đấu diễn ra trong 5 trận đầu tiên của giải đấu

SELECT TOP(5) o.MCName, o.TakecareID, m.Date

from Organize o inner join Match m

on o.OrganizeID = m.ID

ORDER BY m.Date ASC



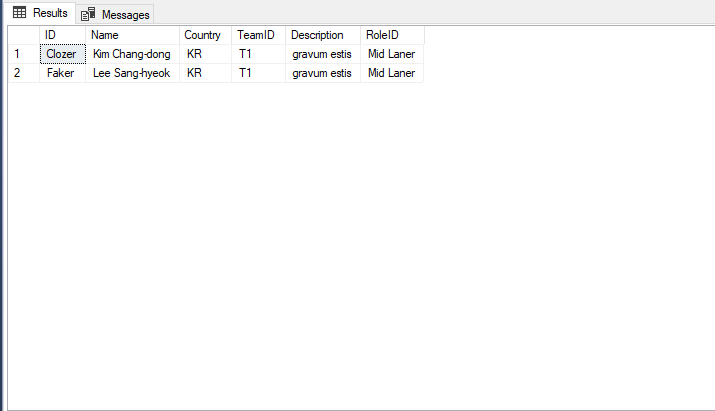
-- Viet mot cau truy van lay ra thong tin (ID, Name, Country, Team, Description, Role) cu tuyen thu den tu team T1 và Role bat dau bang M

select p.ID , p.Name, p.Country, p.TeamID, r.Description , p.RoleID

from Role r inner join Player p

on r.ID = p.RoleID

where p.TeamID = 'T1' and p.RoleID like 'M%'



-- Lấy ra thông tin trận đấu vào ngày cuối cùng của diễn ra giải đấu

select \*

from Match

where Date in

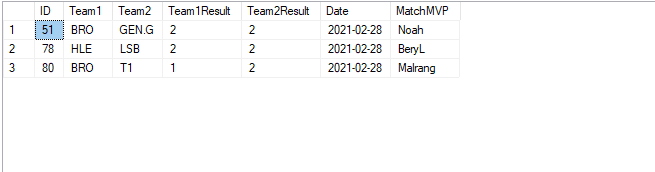
(

select top 1 Date

from Match

order by Date DESC

)



-- Tạo view tính tổng số trận thắng của đội T1 trong cả giải đấu

CREATE VIEW [totalWinGames] AS

SELECT DISTINCT

(   
 SELECT SUM(M.Team1Result)

FROM dbo.Match M

WHERE M.Team1 = 'T1'

)

+

(

SELECT SUM(M.Team2Result)

FROM dbo.Match M

WHERE M.Team2 = 'T1'

)

AS 'Total Win Games'

FROM dbo.Match

SELECT \* FROM totalWinGames



--Lấy ra thông tin của tuyển thủ (Name , RoleID, Country) và số lượt MVP trong đó số lượt MVP cao nhất giải đấu

select p.ID,p.Name , p.RoleID, p.Country, count(m.MatchMVP) as 'Number of MVP'

from Match m inner join Player p

on p.ID = m.MatchMVP

Group by p.ID, p.Name , p.RoleID, p.Country

Having count(m.MatchMVP) >= All

(

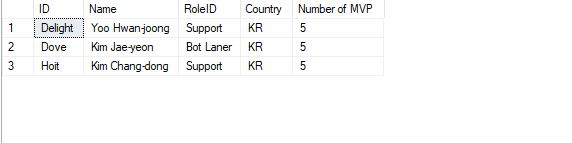
SELECT count(m.MatchMVP) as 'Number of MVP'

FROM Match m inner join Player p

ON p.ID = m.MatchMVP

GROUP by p.ID

)



-- nhập vào tên 1 đội, in ra người đi đường giữa của đội đó

CREATE PROCEDURE midLanerOfTeam @teamID NVARCHAR(10)

AS

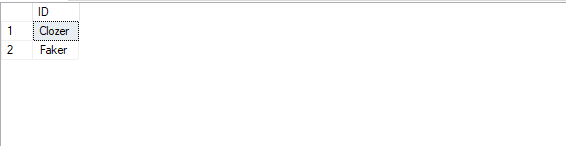
BEGIN

SELECT P.ID FROM Player P

WHERE P.TeamID = @teamID AND P.RoleID = 'Mid Laner'

END

EXEC dbo.midLanerOfTeam @teamID = N'T1' -- nvarchar(10)



-- Tìm trận đấu của 1 đội tuyển theo ngày

CREATE PROCEDURE findScheduleTeam @teamID NVARCHAR(10), @date DATE

AS

BEGIN

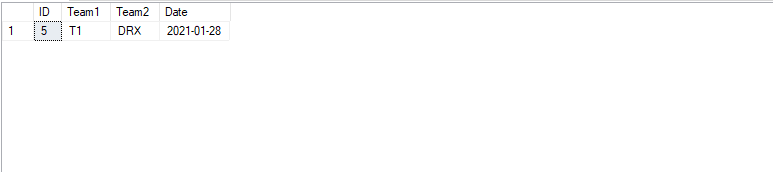
SELECT m.ID,m.Team1, m.Team2, m.Date FROM dbo.Match M

WHERE (m.Team1 = @teamID OR m.Team2 = @teamID) AND m.Date = @date

END

EXEC dbo.findScheduleTeam @teamID = N'T1', -- nvarchar(10)

@date = '2021-01-28' -- date



-- Tạo trigger khi insert 1 player sẽ hiện thị ra team và coach của player đó

CREATE trigger Tr1

on Player

for insert

as

begin

Select i.name,t.ID AS 'Team', t.Coach from inserted i   
inner join Team t on i.TeamID=t.ID

END

-- Test

INSERT INTO dbo.Player

(

ID,

Name,

RoleID,

Country,

TeamID

)

VALUES

( N'Bo Trần', -- ID - nvarchar(15)

N'Trần Đức Bo', -- Name - nvarchar(30)

N'Support', -- RoleID - nvarchar(10)

N'Việt Nam', -- Country - nvarchar(10)

N'LSB' -- TeamID - nvarchar(10)

)



-- Hiện thị ra thông tin Take care đã bị xóa

CREATE TRIGGER Tr2

ON [Take care]

AFTER DELETE

AS

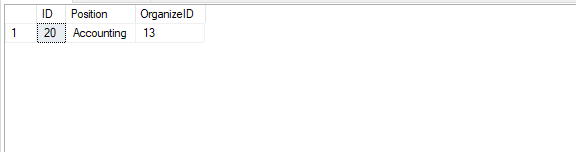
BEGIN

SELECT \* FROM deleted

END

DELETE FROM dbo.[Take care]

WHERE ID = 20



THE END