Explanation

The Manager table has Manager_ID, First_Name, Last_Name, and Age, where **Manager_ID** is the **Primary Key**. Each manager is linked to only one team. The Team table has Team_ID, Team_Name, Team_Abbreviation, Manager_ID, and Year_Founded, where **Team_ID** is the **Primary Key** and **Manager_ID** is the **Foreign Key** referencing to the Manager table. Each team may have multiple players. The Player table has Player_ID, First_Name, Last_Name, Team_ID, Age, and Shirt_Number, where **Player_ID** is the **Primary Key** and **Team_ID** is the **Foreign Key** referencing the Team table. Total participation means every team has one manager, and every player is associated with one team. Referential integrity is enforced by Manager_ID and Team_ID foreign keys.

Normalisation

All three tables are in BCNF, since each non-key attribute is functionally dependent on the primary key of the table it is contained in. No non-prime attributes are dependent on any part of the table except the candidate keys, and there are no transitive dependencies.

DDL Statements

```
CREATE TABLE Manager (
        Manager_ID INTEGER NOT NULL,
        First_Name VARCHAR(50),
        Last_Name VARCHAR(50),
        Age INTEGER,
        PRIMARY KEY (Manager_ID)
);

CREATE TABLE Team (
        Team_ID INTEGER NOT NULL,
        Team_Name VARCHAR(100),
        Team_Abbreviation VARCHAR(5),
        Manager_ID INTEGER NOT NULL,
        Year_Founded INTEGER,
        PRIMARY KEY (Team_ID),
        FOREIGN KEY (Manager_ID) REFERENCES Manager(Manager_ID)
);

CREATE TABLE Player (
        Player_ID INTEGER NOT NULL,
        First_Name VARCHAR(50),
        Last_Name VARCHAR(50),
        Team_ID INTEGER NOT NULL,
        Age INTEGER,
        Shirt_Number INTEGER,
        PRIMARY KEY (Player_ID),
        FOREIGN KEY (Team_ID) REFERENCES Team(Team_ID)
);
```

Relational Algebra Formulations

Deletions

$\text{Manager} \leftarrow \text{Manager} - \sigma_{\text{Manager\_ID} = 1} (\text{Manager})$

$\text{Team} \leftarrow \text{Team} - \sigma_{\text{Team\_ID} = 1} (\text{Team})$

These will both be rejected since the Manager table and Team table has a foreign key in the Team tables and Player table respectively.

Group By

$\pi_{\text{SUM(Age)}} (\sigma_{\text{Team\_ID} = 1} (\text{Player}))$

$\pi_{\text{COUNT(Player\_ID)}} (\sigma_{\text{Team\_ID} = 1} (\text{Player}))$

DML Statements

Deletions

```
DELETE FROM Manager
WHERE Manager_ID = 1;
```

```
DELETE FROM Team
WHERE Team_ID = 1;
```

Group By

```
SELECT SUM(Age)
FROM Player WHERE Team_ID = 1
GROUP BY Team_ID;
```

```
SELECT COUNT(Player_ID)
FROM Player WHERE Team_ID = 1
GROUP BY Team_ID;
```