Cody Eichelberger

November 16th, 2014

Baseball Database

**\*CREATE SCRIPT WILL FOLLOW QUESTION #3**\*

**1. Functional Dependencies**

People

Pid 🡪 FirstName, LastName, StreetAddress, City, PhoneNumber

Players

None

Coaches

Pid 🡪 YearsCoaching

Teams

Tid 🡪 AgeGroupID, TeamName

TeamPlayers

None

TeamCoaches

Pid, tid, AgeGroupID 🡪 CoachLevel

Tid 🡪 AgeGroupID

**2. See PDF**

**3. Third Normal Form**

Database Normalization builds upon each rule so I must prove the first two before the third. Firstly, all data in the tables is atomic so it abides by 1NF. All of my non-key attributes are dependent only upon the primary keys so 2NF is achieved. 3NF is reached because there are no transitive dependencies within the tables. Every non-prime attribute depends on the whole of the primary key. The tables were constructed along those lines of dependency. I believe my tables are in BCNF because any candidate key still dictates the non-key attributes and no non-key attributes determine any of the key attributes.

I may be incorrect as I still find BCNF to be tricky concept to understand when applied to other examples.

**Create Script**

-- People --

CREATE TABLE people (

pid char(4) not null unique,

FirstName text,

LastName text,

StreetAddress text,

City text,

PhoneNumber integer,

primary key(pid)

);

-- Players --

CREATE TABLE players (

pid char(4) not null references people(pid),

primary key(pid),

foreign key(pid) references people(pid)

);

-- Coaches --

CREATE TABLE coaches (

pid char(4) not null references people(pid),

YearsCoaching integer,

primary key(pid),

foreign key(pid) references people(pid)

);

-- Teams --

CREATE TABLE teams (

tid char(4) not null,

AgeGroupID char(4) not null references AgeGroups(AgeGroupID),

TeamName text,

primary key(tid),

foreign key(AgeGroupID) references AgeGroups(AgeGroupID)

);

-- This table establishes the associative entity between teams and players --

-- TeamPlayers --

CREATE TABLE teamPlayers (

tid char(4) not null references teams(tid),

pid char(4) not null references people(pid),

primary key(tid, pid),

foreign key(tid) references teams(tid),

foreign key(pid) references people(pid)

);

-- This table establishes the associative entity between teams and coaches --

-- TeamPlayers --

CREATE TABLE teamCoaches (

tid char(4) not null references teams(tid),

pid char(4) not null references people(pid),

AgeGroupID char(4) not null references AgeGroups(AgeGroupID),

CoachLevel text,

primary key(tid, pid, AgeGroupID),

foreign key(tid) references teams(tid),

foreign key(pid) references people(pid),

foreign key(AgeGroupID) references AgeGroups(AgeGroupID),

unique (pid, AgeGroupID)

);

-- Age Groups --

CREATE TABLE ageGroups (

AgeGroupID char(4) not null,

AgeRange text,

primary key(AgeGroupID)

);