ΑΘΗΝΑ , 15-05-2020

ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

2^{Ο} ΠΑΡΑΔΟΤΕΟ

ΝΙΚΟΣ ΔΗΜΗΤΡΑΚΟΠΟΥΛΟΣ 21821

ΙΩΑΝΝΗΣ (ΧΡΗ)ΣΤΟΥ 218106

ΜΙΧΑΛΗΣ ΒΛΑΣΟΠΟΥΛΟΣ 21810

ΠΕΡΙΕΧΟΜΕΝΑ

ΜΕΛΕΤΗ ΤΗΣ ΦΥΣΗΣ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΡΟΠΟΠΟΙΗΣΗ ΤΟΥ DATASET

Αφού μελετήσαμε τα διάφορα .csv αρχεία καταλήξαμε στο ότι πρέπει να γίνουν μερικές τροποποιήσεις ώστε να επιτευχθεί κανονικοποίηση της βάσης και να μειωθεί το data redundancy (επανάληψη δεδομένων) και διάφορα data anomalies. Παρακάτω θα δούμε τις διάφορες περιπτώσεις ανά .csv :

TEAMSANDLOCATIONS.CSV

Ας δούμε ένα περιορισμένο δείγμα του αρχείου :

TeamName	TeamAcro	Location
Atlanta Hawks	ATL	State Farm Arena Atlanta Georgia
Boston Celtics	BOS	TD Garden Boston Massachusetts

Παρατηρούμε ότι το Location στην ουσία περιέχει 3 πληροφορίες :

- Το όνομα της αρένας που παίζει η ομάδα
- Το όνομα της πόλης που εδρεύει η ομάδα
- Το όνομα της πολιτείας που εδρεύει η ομάδα

Οπότε φτιάξαμε 3 νέες στήλες για να διαχωριστούν τα δεδομένα. Η νέα μορφή για αυτά τα δεδομένα είναι η εξής :

TEAM_NAME	TEAM_ACRONYM	ARENA_NAME	LOCATION_CITY	LOCATION_STATE
Atlanta Hawks	ATL	State Farm Arena	Atlanta	Georgia
Boston Celtics	BOS	TD Garden	Boston	Massachusetts

Ας δούμε τώρα ένα άλλο δείγμα :

TEAM_NAME	TEAM_ACRONYM	ARENA_NAME	LOCATION_CITY	LOCATION_STATE
LA Clippers	LAC	STAPLES Center	Los Angeles	California
Los Angeles Lakers	LAL	STAPLES Center	Los Angeles	California

Εδώ παρατηρούμε ότι οι 2 αυτές ομάδες παίζουν στο ίδιο στάδιο και εδρεύουν στην ίδια πόλη και πολιτεία. Οπότε για να πετύχουμε κανονικοποίηση , τα δεδομένα αυτά πρέπει να διασπαστούν σε επιμέρους πίνακες LOCATIONS , ARENAS και TEAMS :

LOCATIONS :

LOCATION_ID	LOCATION_CITY	LOCATION_STATE
1	Atlanta	Georgia
2	Boston	Massachusetts

ARENAS :

ARENA_ID	ARENA_NAME	LOCATION_ID
1	State Farm Arena	1
2	TD Garden	2

TEAMS :

TEAM_NAME	TEAM_ACRONYM	ARENA_ID	LOCATION_ID
Atlanta Hawks	ATL	1	1
Boston Celtics	BOS	2	2

Δημιουργήσαμε οπότε 3 νέα .csv αρχεία για αυτή τη δουλειά. Η μορφή των .csv μας δεν είναι ακριβώς έτσι γιατί η ανάθεση των IDs θα είναι δουλειά των Identity Columns.

PLAYERSTATS.CSV

Ας δούμε ένα περιορισμένο δείγμα του αρχείου:

	Player	Pos	Age	Tm	G	GS	MP	PTS
117	Corey Brewer\breweco01	SF-SG	31	TOR	72	18	1208	380
118	Corey Brewer\breweco01	SF-SG	31	TOR	72	2	1208	198
119	Corey Brewer\breweco01	SF-SG	31	TOR	72	16	1208	182
120	Corey Brewer\breweco01	SF	31	LAL	54	18	694	380
121	Corey Brewer\breweco01	SF	31	LAL	54	2	694	198
122	Corey Brewer\breweco01	SF	31	LAL	54	16	694	182

Παρατηρούμε ότι υπάρχει περίπτωση ένας παίκτης να έχει και πολλές θέσεις αλλά και πολλά στατιστικά. Πάντως, μετά από μελέτη στο αρχείο παρατηρήσαμε ότι οι θέσεις που έχει παίξει ένας παίκτης σε μία συγκεκριμένη ομάδα δεν αλλάζει ανά τις σεζόν.

(Παρατηρήσαμε επίσης ότι το αρχείο είχε ένα ακρωνύμιο ομάδας TOT , το οποίο δεν υπάρχει στο TeamsAndLocations.csv . Οπότε για να υπάρχει consistency όλα τα TOT έγιναν TOR)

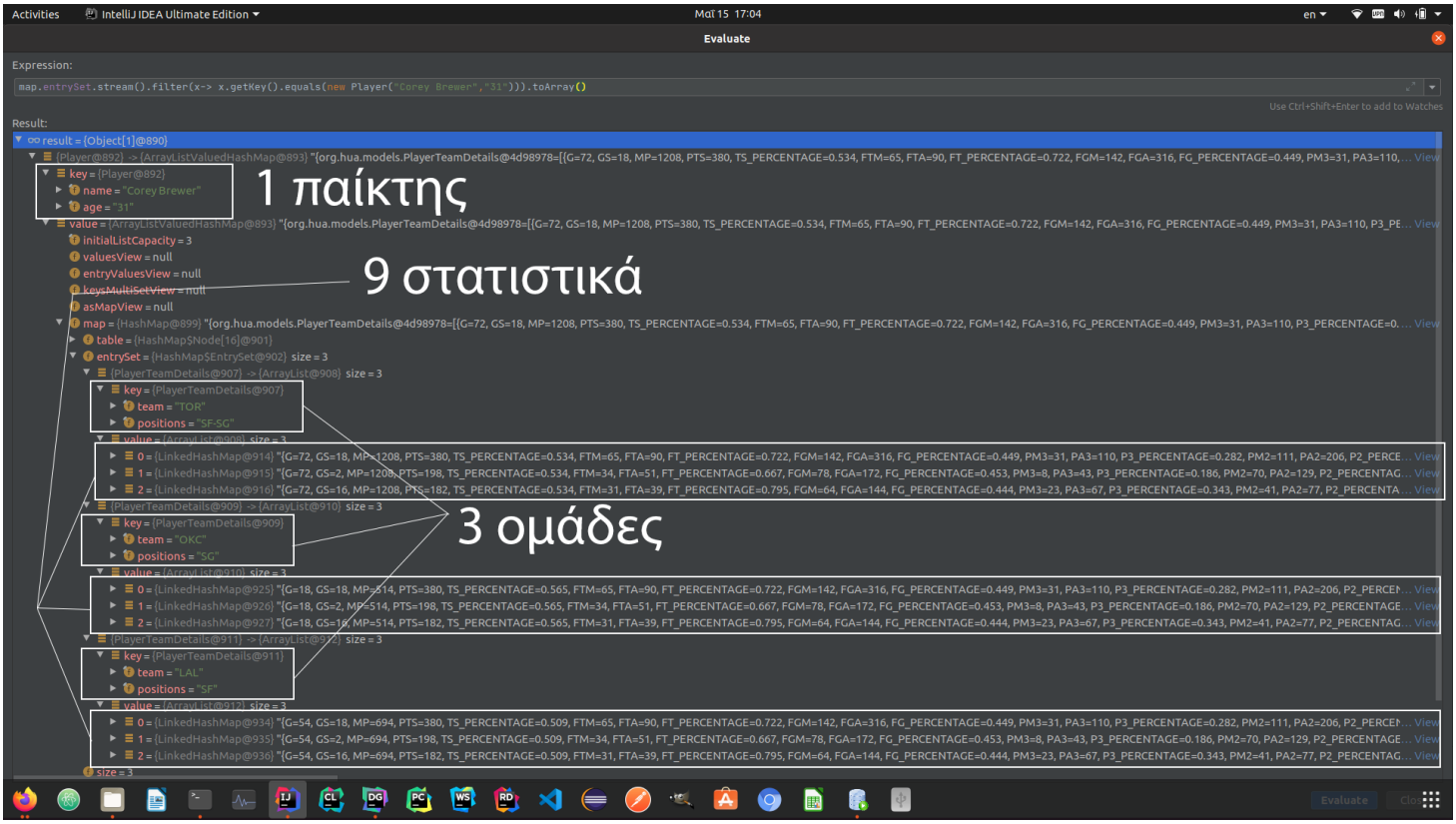
Για να πετύχουμε data redundancy πρέπει :

- Να έχουμε μόνο 1 φορά μέσα στη βάση το όνομα και την ηλικία του παίκτη.
- Να έχουμε μία σχέση μεταξύ παικτών , ομάδων και στατιστικών, δηλαδή την τριάδα (PLAYER_ID + TEAM_ID + STATS_ID).

Για να βρούμε τις θέσεις που έπαιζε ο παίκτης όταν ήταν σε μία ομάδα αρκεί να ξέρουμε τα PLAYER_ID και TEAM_ID. Πολλές θέσεις είναι και πολλές γραμμές στον πίνακα που θα δημιουργηθεί για να διαχειριστεί το πλειότιμο γνώρισμα.

Στο τελικό μας αρχείο teams.csv αφαιρέσαμε μόνο τα IDs που είναι δουλειά των Identity Columns και βγάλαμε από το όνομα ότι ακολουθεί μετά το \ (π.χ Corey Brewer\breweco01 -> Corey Brewer)

Να σημειωθεί ότι δεν καλύπτεται το ενδεχόμενο συνωνυμίας μεταξύ διαφορετικών παικτών. Αν υπήρχαν στο NBA 2 Stephen Curry εμείς θα τον παίρναμε για έναν.



Εικόνα 1: Στιγμιότυπο από τον custom SQL Generator μας. Εδώ φαίνεται η οργάνωση των δεδομένων του players.csv

GAMESTATS.CSV

Ας δούμε ένα περιορισμένο δείγμα του αρχείου:

	Team	Game	Date	Home	Opponent	WINorLOSS	TeamPoints	OpponentPoints
0	ATL	1	29/10/2014	Away	TOR	L	102	109
…	…	…	…	…	…	…	…	…
2214	TOR	1	29/10/2014	Home	ATL	W	109	102

Field Goals	FieldGoalsAttempted	FieldGoals.	…	Opp.FieldGoals	Opp.FieldGoals Attempted	Opp.FieldGoals.	…
40	80	0.5	…	37	90	0.411000000000000003	…
…	…	…	…	…	…	…	…
37	90	0.411000000000000003	…	40	80	0.5	…

Παρατηρούμε ότι για κάθε μοναδικό παιχνίδι υπάρχουν 2 γραμμές , στις οποίες γίνεται εναλλαγή της θέσης των στηλών που αφορούν γηπεδούχο – φιλοξενούμενο.

Για να πετύχουμε κανονικοποίηση, πρέπει :

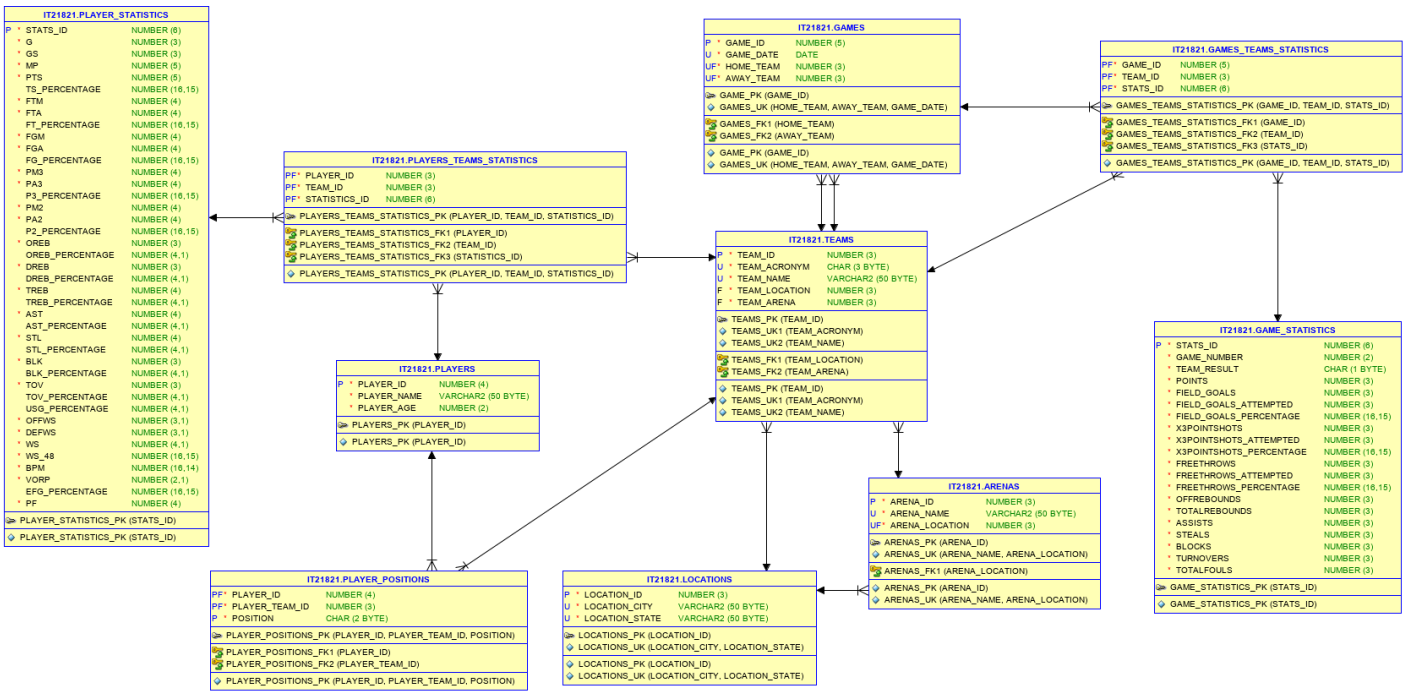
- **Να κρατήσουμε μόνο τη 1 από τις 2 εγγραφές** χωρίς όμως να χαθεί και η αντιστοιχία μεταξύ των δεδομένων της γηπεδούχου και της φιλοξενούμενης ομάδας . Για αυτό φιλτράραμε το αρχείο και κρατήσαμε μόνο τις γραμμές που στη στήλη Home έχουν τη τιμή Home . Έτσι πάντα ξέρουμε ότι η 1η ομάδα είναι η γηπεδούχος και η άλλη η φιλοξενούμενη.
- **Να προσθέσουμε 2 στήλες** : Τη WINorLOSS της φιλοξενούμενης ομάδας και το GAME της φιλοξενούμενης ομάδας.

Μπορείτε να δείτε όλες τις αλλαγές στο games.csv.

ΣΧΕΔΙΑΣΗ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

ΑΠΟ ΤΟ DATASET ΣΕ ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ

Παρακάτω φαίνεται το σχεσιακό μοντέλο του σχήματος που δημιουργείται μετά την εκτέλεση της create.sql :



Η προηγούμενη ενότητα πάνω κάτω καλύπτει τη λογική στην οποία βασίστηκε αυτό το σχεσιακό μοντέλο. Κάποιες έξτρα παρατηρήσεις που θεωρούμε σκόπιμες ότι πρέπει να γίνουν είναι οι εξής :

- Μετά από έλεγχο στα .csv καταλήξαμε στο ποια attributes μπορούν να είναι NULL και ποια NOT NULL
- Η ακρίβεια των attributes που είναι δεκαδικοί αριθμοί πάλι προσδιορίστηκαν με βάση το .csv
- Η εγκυρότητα του σχεσιακού μοντέλου ελέγχθηκε με την εισαγωγή όλου του dataset.
- Δημιουργήθηκαν κάποια πρόσθετα constraints (UNIQUE , CHECK) τα οποία φανέρωσε η ανάλυση των δεδομένων.
- Τα στατιστικά ενός παιχνιδιού διαχωρίστηκαν ανά τις 2 ομάδες που συμμετέχουν στο παιχνίδι και όχι στο παιχνίδι γενικότερα (1 αγώνας λοιπόν έχει 2 στατιστικά , 1 για κάθε ομάδα που έπαιξε στο παιχνίδι αυτό).

ΚΑΠΟΙΕΣ ΤΕΧΝΙΚΕΣ ΛΕΠΤΟΜΕΡΕΙΕΣ

IDENTITY COLUMNS – SEQUENCES

Στην εργασία θεωρήθηκε ιδανικό να αφήσουμε στη βάση τον έλεγχο της ανάθεσης των ID στις οντότητές μας ώστε να μην υπάρχει πιθανότητα κάποιου ζητήματος referential integrity.

Η σχολή μας χρησιμοποιεί την έκδοση Oracle Database 12c η οποία έφερε την υποστήριξη για τα Identity Columns . Τα Identity Columns χρησιμοποιούνται για την παραγωγή surrogate primary keys. Κάθε φορά που γίνεται εισαγωγή μιας νέας γραμμής , η Oracle δημιουργεί αυτόματα και εισάγει μια ακολουθιακή τιμή στο πεδίο της στήλης. Τα identity columns στηρίζονται πάνω στα sequences που η Oracle υποστηρίζει από την αρχή της κυκλοφορίας της.

ΕΙΣΑΓΩΓΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ

PL/SQL BLOCKS

Τα PL/SQL Blocks είναι ο τρόπος εκτέλεσης ενός γκρουπ εντολών στην Oracle. Προσφέρει κάποιες σημαντικές δυνατότητες όπως η υποστήριξη μεταβλητών και Exception Handling. Χρησιμοποιήσαμε PL/SQL Blocks για να μπορούμε να έχουμε δυναμική εκτέλεση των inserts και να ανακτούμε και να καταχωρούμε τα διάφορα IDs των identity columns που δημιουργούνται κατά την εκτέλεση και δεν ξέρουμε εκ των προτέρων.

ΤΡΟΠΟΣ ΔΗΜΙΟΥΡΓΙΑΣ ΤΟΥ INSERT.SQL

Τα insert μας δημιουργήθηκαν με τη χρήση script . Όμως πριν προχωρήσουμε στο scripting δημιουργήσαμε δείγματα insert για την κάθε οντότητα ώστε να ξέρουμε τι ακριβώς πρέπει να παράγουμε.

ΕΙΣΑΓΩΓΗ ΤΟΠΟΘΕΣΙΑΣ

Οι τοποθεσίες εισάγονται πρώτες καθώς δεν εξαρτώνται από καμία άλλη οντότητα. Δεν χρειάζεται PL/SQL εδώ.

```
INSERT INTO LOCATIONS (LOCATION_CITY,LOCATION_STATE)
VALUES ('Atlanta','Georgia');
```

ΕΙΣΑΓΩΓΗ ΑΡΕΝΑΣ

Για να μπει μια αρένα μέσα , θέλουμε το όνομα της αρένας και το ID της τοποθεσίας της , οπότε ψάχνουμε το ID με SELECT στον πίνακα των τοποθεσιών. Για καθαρή γραφή χρησιμοποιήθηκε PL/SQL αν και αυτή η εισαγωγή μπορεί να γίνει και με φωλιασμένη SELECT.

```
DECLARE
  LOC_ID NUMBER(3);
BEGIN
  SELECT LOCATION_ID
  INTO LOC_ID
  FROM LOCATIONS
  WHERE LOCATION_CITY = 'Atlanta'
  AND LOCATION_STATE = 'Georgia';

  INSERT INTO ARENAS (ARENA_NAME,ARENA_LOCATION)
  VALUES ('State Farm Arena',LOC_ID);
  COMMIT;
END;
```

ΕΙΣΑΓΩΓΗ ΟΜΑΔΑΣ

Για να μπει μια ομάδα μέσα , θέλουμε το ID της αρένας και το ID της τοποθεσίας της μαζί με το όνομα της ομάδας και το ακρωνύμιό της , οπότε ψάχνουμε τα IDs πάλι με SELECT.

```
DECLARE
  LOC_ID NUMBER(3);
  AR_ID NUMBER(3);
BEGIN
  SELECT LOCATION_ID
  INTO LOC_ID
  FROM LOCATIONS
  WHERE LOCATION_CITY = 'Atlanta'
  AND LOCATION_STATE = 'Georgia';

  SELECT ARENA_ID
  INTO AR_ID
  FROM ARENAS
  WHERE ARENA_NAME = 'State Farm Arena'
  AND ARENA_LOCATION = LOC_ID;

  INSERT INTO TEAMS (TEAM_NAME,TEAM_ACRONYM,TEAM_ARENA,TEAM_LOCATION)
  VALUES('Atlanta Hawks','ATL',AR_ID,LOC_ID);
  COMMIT;
END;
```

ΕΙΣΑΓΩΓΗ ΠΑΙΚΤΗ

```
DECLARE
   PL_ID NUMBER(3);
   TE_ID NUMBER(3);
   ST_ID NUMBER(3);
BEGIN
  /* INSERT PLAYER BASIC INFO */
  INSERT INTO PLAYERS (PLAYER_NAME,PLAYER_AGE)
  VALUES ('Omer Asik',31)
  RETURNING PLAYER_ID INTO PL_ID;

  /*THIS COVERS THE PLAYERS CAREER IN TOR*/
   SELECT TEAM_ID
   INTO TE_ID
   FROM TEAMS
   WHERE TEAM_ACRONYM = 'TOR';

   INSERT INTO PLAYER_POSITIONS (PLAYER_ID, PLAYER_TEAM_ID, POSITION) VALUES
(PL_ID,TE_ID,'C');
   INSERT INTO PLAYER_POSITIONS (PLAYER_ID, PLAYER_TEAM_ID, POSITION) VALUES
(PL_ID,TE_ID,'SG');

   INSERT INTO PLAYER_STATISTICS
(G,GS,MP,PTS,TS_PERCENTAGE,FTM,FTA,FT_PERCENTAGE,FGM,FGA,FG_PERCENTAGE
,PM3,PA3,P3_PERCENTAGE,PM2,PA2,P2_PERCENTAGE,OREB,OREB_PERCENTAGE,DREB,DREB_PERC
ENTAGE,TREB,TREB_PERCENTAGE,AST,AST_PERCENTAGE,STL,STL_PERCENTAGE,BLK,BLK_PERCEN
TAGE,TOV,TOV_PERCENTAGE,USG_PERCENTAGE,OFFWS,DEFWS,WS,WS_48
   ,BPM,VORP,EFG_PERCENTAGE,PF)
   VALUES
(18,0,182,22,0.397,4,13,0.308,9,22,0.409,0,0,null,9,22,0.409,9,5.5,38,22.2,47,14.0,3,2.1,2,0.5,4,1.9,9,24
.5,8.7,
   -0.2,0.2,-0.1,-0.021,-8.1,-0.3,0.409,20) RETURNING STATS_ID INTO ST_ID;

   INSERT INTO PLAYERS_TEAMS_STATISTICS (PLAYER_ID, TEAM_ID, STATISTICS_ID) VALUES
(PL_ID,TE_ID,ST_ID);

  INSERT INTO PLAYER_STATISTICS
(G,GS,MP,PTS,TS_PERCENTAGE,FTM,FTA,FT_PERCENTAGE,FGM,FGA,FG_PERCENTAGE

,PM3,PA3,P3_PERCENTAGE,PM2,PA2,P2_PERCENTAGE,OREB,OREB_PERCENTAGE,DREB,DREB_PERC
ENTAGE,TREB,TREB_PERCENTAGE,AST,AST_PERCENTAGE,STL,STL_PERCENTAGE,BLK,BLK_PERCEN
TAGE,TOV,TOV_PERCENTAGE,USG_PERCENTAGE,OFFWS,DEFWS,WS,WS_48
  ,BPM,VORP,EFG_PERCENTAGE,PF)
   VALUES
(20,5,182,22,0.397,4,13,0.308,9,24,0.409,0,0,null,9,22,0.409,9,5.5,38,22.2,47,14.0,3,2.1,2,0.5,4,1.9,9,24
.5,8.7,
   -0.2,0.2,-0.1,-0.021,-8.1,-0.3,0.409,20) RETURNING STATS_ID INTO ST_ID;

   INSERT INTO PLAYERS_TEAMS_STATISTICS (PLAYER_ID, TEAM_ID, STATISTICS_ID) VALUES
(PL_ID,TE_ID,ST_ID);

  COMMIT;
END;
```

Εδώ αρχίζει να φαίνεται η χρησιμότητα της PL/SQL. Με την εισαγωγή του παίκτη αυτόματα επιστρέφεται στο PL_ID το ID που η βάση έδωσε σε αυτόν τον παίκτη . Έπειτα πρέπει να καταχωρήσουμε τις θέσεις που έπαιζε και τα στατιστικά του για κάθε ομάδα στην οποία έπαιζε. Οπότε τραβάμε το ID της ομάδας , μετά κάνουμε εισαγωγή των θέσεων και βάζουμε και όσα στατιστικά αντιστοιχούν με αυτόν τον παίκτη και ομάδα. Μετά από κάθε εισαγωγή στατιστικό γίνεται και η τριπλή συσχέτιση στο PLAYERS_TEAMS_STATISTICS.

ΕΙΣΑΓΩΓΗ ΠΑΙΧΝΙΔΙΟΥ

```
ALTER SESSION SET NLS_DATE_FORMAT = 'DD-MM-YYYY HH24:MI:SS';

DECLARE
   HOME_TEAM_ID  NUMBER(3);
   AWAY_TEAM_ID  NUMBER(3);
   GA_ID        NUMBER(5);
   GA_STATS     NUMBER(6);
BEGIN
   SELECT TEAM_ID
   INTO HOME_TEAM_ID
   FROM TEAMS
   WHERE TEAM_ACRONYM = 'BOS';

   SELECT TEAM_ID
   INTO AWAY_TEAM_ID
   FROM TEAMS
   WHERE TEAM_ACRONYM = 'CLE';

   INSERT INTO GAMES (GAME_DATE, HOME_TEAM, AWAY_TEAM)
   VALUES ('28/10/2014', HOME_TEAM_ID, AWAY_TEAM_ID)
   RETURNING GAME_ID INTO GA_ID;

   INSERT INTO GAME_STATISTICS (GAME_NUMBER, TEAM_RESULT, POINTS, FIELD_GOALS,
FIELD_GOALS_ATTEMPTED, FIELD_GOALS_PERCENTAGE, X3POINTSHOTS,
X3POINTSHOTS_ATTEMPTED, X3POINTSHOTS_PERCENTAGE, FREETHROWS,
FREETHROWS_ATTEMPTED, FREETHROWS_PERCENTAGE, OFFREBOUNDS, TOTALREBOUNDS,
ASSISTS, STEALS, BLOCKS, TURNOVERS, TOTALFOULS)
   VALUES (90, 'W', 121, 48, 88, 0.545, 6, 20, 0.3, 19, 23, 0.826, 8, 39, 32, 7, 1, 14, 26) RETURNING
STATS_ID INTO GA_STATS;

   INSERT INTO GAMES_TEAMS_STATISTICS(GAME_ID,TEAM_ID,STATS_ID) VALUES (GA_ID,
HOME_TEAM_ID, GA_STATS);

   INSERT INTO GAME_STATISTICS (GAME_NUMBER, TEAM_RESULT, POINTS, FIELD_GOALS,
FIELD_GOALS_ATTEMPTED,FIELD_GOALS_PERCENTAGE, X3POINTSHOTS,
X3POINTSHOTS_ATTEMPTED, X3POINTSHOTS_PERCENTAGE,  FREETHROWS,
FREETHROWS_ATTEMPTED, FREETHROWS_PERCENTAGE, OFFREBOUNDS, TOTALREBOUNDS,
 ASSISTS, STEALS, BLOCKS, TURNOVERS, TOTALFOULS)
VALUES (90, 'L', 121, 48, 88, 0.545, 6, 20, 0.3, 19, 23, 0.826, 8, 39, 32, 7, 1, 14, 26) RETURNING
STATS_ID INTO GA_STATS;

 INSERT INTO GAMES_TEAMS_STATISTICS(GAME_ID,TEAM_ID,STATS_ID) VALUES (GA_ID,
AWAY_TEAM_ID, GA_STATS);

COMMIT;
END;
```

Για να εισάγουμε όλες τις πληροφορίες σχετικά με ένα παιχνίδι , πρώτα παίρνουμε τα IDs των 2 ομάδων. Μετά εισάγουμε το παιχνίδι και έπειτα τα στατιστικά κάθε ομάδας στο GAME_STATISTICS. Μετά την εισαγωγή κάθε στατιστικού εισάγουμε την τριπλή συσχέτιση στο GAMES_TEAMS_STATISTICS.