**API implantation-**

**CRUD API**  
**AddTeam(int teamId)**- אנו מוסיפים שורה לטבלה Teams שיצרנו עם ערך הteamId שמקבלים.  
ערכי החזרה של הפונקציה כפי שמפורטים בגליון.

**addMatch(Match match)-** בהינתן class match מטיפוס Match, אנו מוציאים את הערכים הרלוונטים לattributes בטבלה שיצרנו- Matches, ומוסיפים את הmatch לטבלה .(הוספת שורה)  
ערכי החזרה של הפונקציה כפי שמפורטים בגליון.  
   
**getMatchProfile(int matchId) –** בהינתן הid של הmatch, אנו מוצאים אותו בטבלה Matches ומוציאים את הערכים הרלוונטים אליו, בעזרתם אנו יוצרים match מטיפוס Match ומחזירים אותו.  
במידה ולא קיים משחק בטבלה matches עם matched שקיבלנו, נחזיר BadMatch().  
   
**deleteMatch(Match match)-** נחלץ את הmatchId של match על מנת לחפש אותו בטבלה Matches ולבצע מחיקה אם קיים שם.(מחיקת שורה בטבלה)  
ערכי החזרה של הפונקציה כפי שמפורטים בגליון.  
  
**addPlayer(Player player)-** בהינתן player מטיפוס Player, נוציא ממנו את כל הערכים הרלוונטים לattributes בטבלה שיצרנו- Players ומוסיפים את הplayer לטבלה.(הוספת שורה)  
ערכי החזרה של הפונקציה כפי שמפורטים בגליון.  
  
  
**getPlayerProfile(int playerId)-** בהינתן playerId, נחפש בטבלה Players את השחקן המתאים לid הנתון. במידה ומצאנו אותו, ניצור player מטיפוס Player עם הערכים המתאים לשורה של השחקן שמצאנו, ונחזיר אותו. אחרת נחזיר BadPlayer().  
  
**deletePlayer(Player player)-** בהינתן player מטיפוס Player, נוציא מהטיפוס את הplayerId ואותו נחפש בטבלה Players. במידה וקיים, נבצע מחיקה של השורה המיוצגת ע"י playerId בטבלה Players ונחזיר OK . אחרת, במידה ולא קיים נחזיר NOT\_EXISTS וערך ההחזרה הנוסף הוא לפי הנתון בגליון.  
  
**addStadium(Stadium stadium)-** בהינתן stadium מטיפוס Stadium, אנו מוציאים את הערכים הרלוונטים לattributes בטבלה שיצרנו- Stadiums, ומוסיפים את הstadium לטבלה .(הוספת שורה)  
ערכי החזרה של הפונקציה כפי שמפורטים בגליון.  
  
**getStadiumProfile(int stadiumID)-** נחפש בטבלה Stadiums באמצעות stadiumID. במידה ונמצא שורה מתאימה בטבלה, נוציא את הattributes של השורה ונבנה stadium מטיפוסStadium ונחזיר אותו. אחרת, במידה ולא קיים נחזיר NOT\_EXISTS וערך ההחזרה הנוסף הוא לפי הנתון בגליון.  
 **deleteStadium(Stadium stadium)-** בהינתן stadium מטיפוס Stadium, נוציא מהטיפוס את ערך הstadiumId ונבצע לפיו חיפוש בטבלה Stadiums. במידה ונמצא שורה מתאימה, נמחק אותה מהטבלה ונחזיר OK. אחרת, במידה ולא נמצא שורה מתאימה, נחזיר NOT\_EXISTS וערך ההחזרה הנוסף הוא לפי הנתון בגליון.

**BASIC API-**

**playerScoredInMatch(Match match, Player player, int Amount)-** נחלץ מmatch ומplayer את matchID ו-playerID בהתאמה ונכניס לטבלה scores שורה חדשה עם הערכים amount,playerID,matchId. ערכי החזרה כפי שמפורטים בגליון.

**playerDidntScoreInMatch(Match match, Player player)-** בהינתן match וplayer נחלץ מהם את matchID ואת playerID בהתאמה ונמחק את השורה בscores בה מופיעים ערכים אלו תחת העמודות המתאימות. ערכי החזרה כפי שמפורטים בגליון.  
 **matchInStadium(Match match, Stadium stadium, int attendance)** – נחלץ מmatch ומstadium את matchID ואת stadiumID בהתאמה. כעת נוסיף שורה חדשה לטבלה matchInStadium עם הערכים match,stadium,attendance. ערכי החזרה כפי שמפורטים בגליון.  
  
  
**matchNotInStadium(Match match, Stadium stadium)** – נחלץ מmatch ומstadium את matchID ואת stadiumID בהתאמה. כעת נחפש את השורה המתאימים לערכים אלו בטבלה matchInStadium ובמידה ונמצא, נמחק אותה. ערך החזרה של כל מקרה אחר- לפי המפורט בגליון.

**averageAttendanceInStadium(int stadiumID)-** בהינתן stadiumID נפעיל SELECT על הטבלה matchInStadium כך שהשורות שיישארו יהיו השורות של stadiumID בלבד. לאחר מכן נפעיל AVG על עמודת attendance ונחזיר את הערך המתקבל. ערכי החזרה כפי שמפורטים בגליון.

**stadiumTotalGoals(int stadiumID)**- השתמשנו בטבלאות matchInStadium ובטבלה scores על מנת להפיק עמודה ושורה אחת שמכילה את המספר הנדרש להחזיר- כמות הגולים שהובקעו באצטדיון עם stadiumID. במידה והstasiumId לא קיים , נחזיר 0 ובמקרה של error נחזיר 1-.

**playerIsWinner(int playerID, int matchID)**- השתמשנו בטבלאות view (עליהן הרחבנו תחילה) personalStats ו- goalsPerMatch. ביצענו עליהן union all על מנת לא לאבד כפילויות והחזרנו true במידה ומתקיים .

**getActiveTallTeams**()- השתמשנו בactiveTallTeams (טבלת view עליה הסברנו בהתחלה) והחזרנו ממנה רשימה בסדר יורד באורך 5 כנדרש. במקרה שactiveTallTeams ריקה נחזיר רשימה ריקה.

**getActiveTallRichTeams**()- החזרנו רשימה בסדר יורד ובאורך של עד 5, של הid של הקבוצות שנמצאות בחיתוך של activeTallTeams (טבלת view) עם טבלת stadium שסיננו בה את השורות בהם . במידה ואין בחיתוך אף קבוצה, נחזיר רשימה ריקה.

**popularTeams**()- אנחנו משתמשים בטבלאות Teams ו-minAttendancePerTeam(טבלת view אשר פירטנו עליה). מטילים את minAttendancePerTeam על Teams לקבלת מספר הצפיות המינימלי או null- במקרה ולא שיחקו משחקים באצטדיון. נזכור כי ב minAttendancePerTeam כבר מעניקים מס' צופים = 0 לאצטדיונים ששיחקו בהם משחקים אשר לא דווח בהם מספר הצופים. נסנן שורות עם  *40000 צופים, נמיין , נגביל ל10 ונחזיר את הרשימה.*