**Sri Lanka Institute of Information**

**Technology**



**Data warehousing and Business Intelligence**

**Assignment 1**

**Student Registration No: IT20762186**

**Student Name: Gunasekara B.A.J.C.**

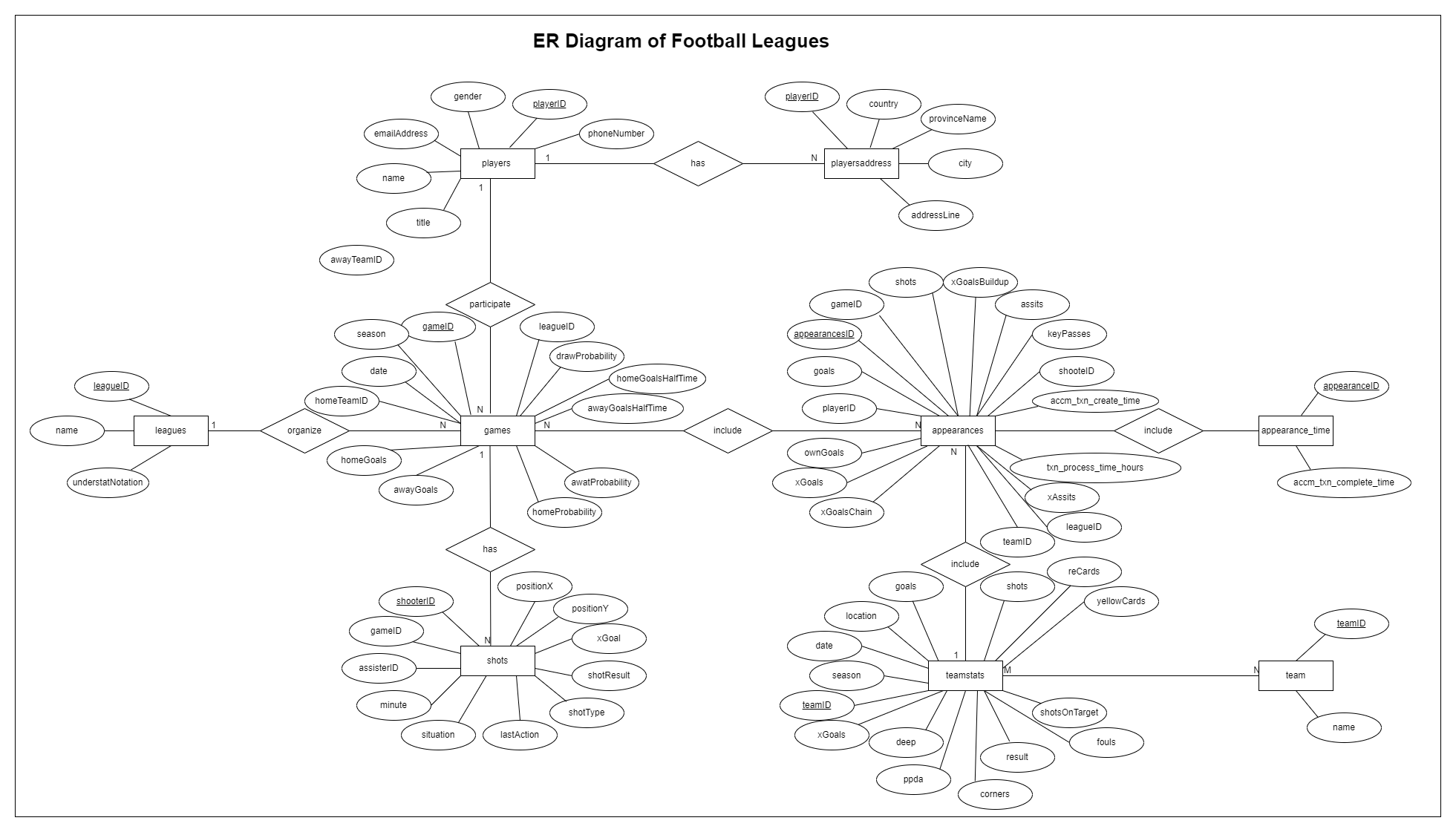
**Step 1: Data Set Selection**

This dataset contains football-related data covering the Top5 leagues in Europe from 2014-2020. Mainly players, leagues, games, teams, shots and appearances are interacted with this data set. Some modifications were done accordingly to the tables of above data set. Two .csv files added to the data source additionally. This .csv files are playersaddress.csv and appearance\_time.csv.

The link to the source data set is mentioned below:

<URL:-> <https://www.kaggle.com/datasets/technika148/football-database>

**Following ER- diagram will describe the scenario of the selected dataset.**



**Step 2: Preparation of Data Sources**

The whole of data was in ‘csv’ file type and they were separated into the following data sources, Database, Text and csv. And they were used to create the following,

**1. Database (.bak)**

Players.csv, games.csv, appearances.csv, appearance.time.csv and teamstats.csv were imported to the Football Leagues Source Database. This data was used as the DB source data.

**2. Text (.txt)**

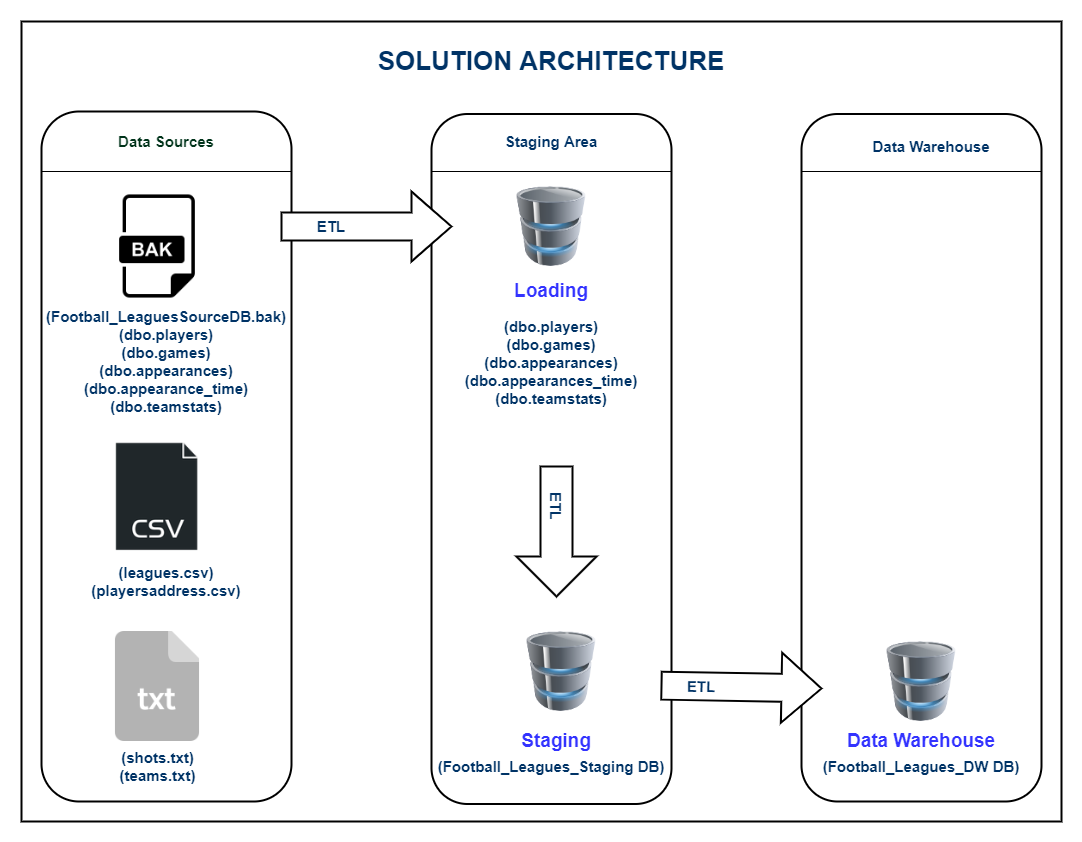
teams.txt and shots.txt were used directly.

**3. Comma Separated Values (.csv)**

leagues.csv and playersaddress.csv were used

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Source**  **Type** | **Source Name** | **Column Name** | **Data Type** | **Description** |
| Database  File(.bak) | dbo.players | playerID | numeric | Players details  of Football |
|  |  | gender | nvarchar(50) | Leagues. |
|  |  | name | nvarchar(50) |  |
|  |  | title | nvarchar(50) |  |
|  |  | phoneNumber | nvarchar(50) |  |
|  |  | emailAddress | nvarchar(50) |  |
|  | dbo.games | gameID | numeric | Games details of |
|  |  | leagueID | numeric | Football |
|  |  | season | int | Leagues. |
|  |  | date | datetime |  |
|  |  | homeTeamID | int |  |
|  |  | awayTeamID | int |  |
|  |  | homeGoals | int |  |
|  |  | awayGoals | int |  |
|  |  | homeProbability | int |  |
|  |  | awayProbability | int |  |
|  |  | dawProbability | int |  |
|  |  | homeGoalsHaltTime | int |  |
|  |  | awayGoalsHalfTime | int |  |
|  | dbo.appearances | appearancesID | numeric | Summary of |
|  |  | gameID | numeric | games, players, |
|  |  | playerID | numeric | teams, shots |
|  |  | teamID | numeric | and leagues |
|  |  | goals | int |  |
|  |  | ownGoals | int |  |
|  |  | shots | int |  |
|  |  | date | datetime |  |
|  |  | xGoals | int |  |
|  |  | xGoalsChain | int |  |
|  |  | xGoalsBuildup | int |  |
|  |  | assists | int |  |
|  |  | keyPasses | int |  |
|  |  | xAssists | int |  |
|  |  | position | nvarchar(50) |  |
|  |  | positionOrder | int |  |
|  |  | yellowCard | nvarchar(50) |  |
|  |  | redCard | int |  |
|  |  | substituteIn | int |  |
|  |  | substitueOut | int |  |
|  |  | leagueID | numeric |  |
|  | dbo.appearance\_time | appearancesID | numeric | Additional |
|  |  | accm\_txn\_complete\_  time | datetime | Appearances table. |
|  | dbo.teamstats | teamID | numeric | Teams |
|  |  | season | int | performance |
|  |  | date | datetime | details of |
|  |  | location | nvarchar(50) | Football |
|  |  | goals | int | Leagues. |
|  |  | xGoals | int |  |
|  |  | shots | int |  |
|  |  | shotsOnTarget | int |  |
|  |  | deep | int |  |
|  |  | ppda | int |  |
|  |  | fouls | int |  |
|  |  | corners | int |  |
|  |  | yellowCards | nvarchar(50) |  |
|  |  | redCards | int |  |
|  |  | result | nvarchar(50) |  |
| CSV File | leagues.csv | leagueID | numeric | Leagues details |
|  |  | name | nvarchar(50) | of Football |
|  |  | understatNotation | nvarchar(50) | Leagues. |
|  | playersaddress.csv | playerID | numeric | Players |
|  |  | addressLine | nvarchar(50) | Addresses detail |
|  |  | city | nvarchar(50) | -s of Football |
|  |  | provinceName | nvarchar(50) | Leagues. |
|  |  | country | nvarchar(50) |  |
| Text File | shots.txt | gameID | numeric | Shots details |
|  |  | shooterID | numeric | of Football |
|  |  | assisterID | nvarchar(50) | Leagues |
|  |  | minute | nvarchar(50) |  |
|  |  | situation | nvarchar(50) |  |
|  |  | lastAction | nvarchar(50) |  |
|  |  | shotType | nvarchar(50) |  |
|  |  | shotResult | nvarchar(50) |  |
|  |  | xGoal | int |  |
|  |  | PositionX | int |  |
|  |  | positionY | int |  |
|  | teams.txt | teamID | numeric | Teams details |
|  |  | name | nvarchar(50) | of Football  Leagues. |

**Step 3: Solution Architecture**

****

Above architecture shows the high-level BI solution to the warehouse design.

**Data Sources**

‘.txt’ component represents Text files, ‘.csv’ component is used to display Comma Separated files and ‘.bak’ component represents database files.

**Staging Area**

Loading DB component represents the process of the creating database tables. The games.csv, players.csv, appearances.csv, appearance\_time.csv and teamstats.csv files were imported to the database and was used to create the tables. And these tables were used as the DB source data.

Staging DB component represents creating staging level tables through the ‘Extract’.

**Data Warehouse**

Data warehouse DB component is used display the cratering dimension tables in the warehouse using ‘Transform’ and ‘Load.’

**Step 4: Data Warehouse Design & Development**

|  |
| --- |
| DimDate |
| DateKey  Date  FullDateUK  FullDateUSA  DayOfMonth  DaySuffix  DayName  DayOfWeekUSA  DayOfWeekUK  DayOfWeekInMonth  DayOfWeekInYear  DayOfQuater  DayOfYear  WeekOfMonth  WeekOfQuater  WeekOfYear  Month  MonthName  MonthOfQuater  Quarter  QuaterName  Year  YearName  MonthYear  MMYYYY  FirstDayOfMonth  LastDayOfMonth  FirstDayOfQuater  LastDayOfQuater  FirstDayOfYear  LastDayOfYear  IsHolidaySL  IsWeekday  HolidaySL  isCurrentDay  isDateAvailable  isLatestDateAvailable |

Following figure will show how the fact table and dimension tables was combined in a rational manner.

|  |
| --- |
| Dimleagues |
| leageSk  AlternateleagueID  leagueName  undestatNotation  insertDate  modifiedDate |

|  |
| --- |
| Dimplayers |
| playerSK  AlternateplayerID  Gender  Name  Title  phoneNumber  emailAddress  addessLine  city  provinceName  country  StartDate  EndDate  InserDate  ModifiedDate |

|  |
| --- |
| Factappearances |
| appearanceID  gameKey  playerKey  teamKey  goals  ownGoals  shots  dateKey  xGoals  xGoalsChain  xGoalsBuildup  assists  keyPasses  xAssists  position  positionOrder  yellowCard  redCard  substituteIn  subastiteOut  leagesKey  accm\_txn\_complete\_time  accm\_txn\_create\_time  txn\_process\_time\_hours  modifiedDate |

|  |
| --- |
| Dimpteamstats |
| teamSK  AlternateteamID  Season  Date  Location  Goals  xGoals  shots  shotsOnTarget  deep  ppda  fouls  corners  yellowCards  redCards  result  teamName  insertDate  modifiedDate |

|  |
| --- |
| Dimgames |
| gameSk  AlternategameID  leagueKey  season  date  homeTeamID  awayTeamID  homeGoals  awayGoals  homeProbability  awayProbability  drawProbability  homeGoalsHalfTime  awayGoalsHalfTime  inserDate  modefiedDate |

|  |
| --- |
| Dimshots |
| shooterSK  AlternateshooterID  gameKey  assisterID  minute  situation  lastAction  shotType  shotResult  xGoal  position  position  insertDate  modefiedDate |

**Schema Type**

For this scenario, snowflake schema type was used.

**Dimension Types**

* Hierarchical Dimension
  + Date – all the hierarchies in date
  + players – country→ province name → city → address line
* Slowly Changing Dimension
* players – changing attribute →Phone Number
* Fact Table
* Numbers – accm\_txn\_create\_time, accm-txn\_complete\_time,

txn\_process\_time\_hours

* FK - gameID, playerD, teamID, Date Key, leagueID

**Step 5: ETL development**

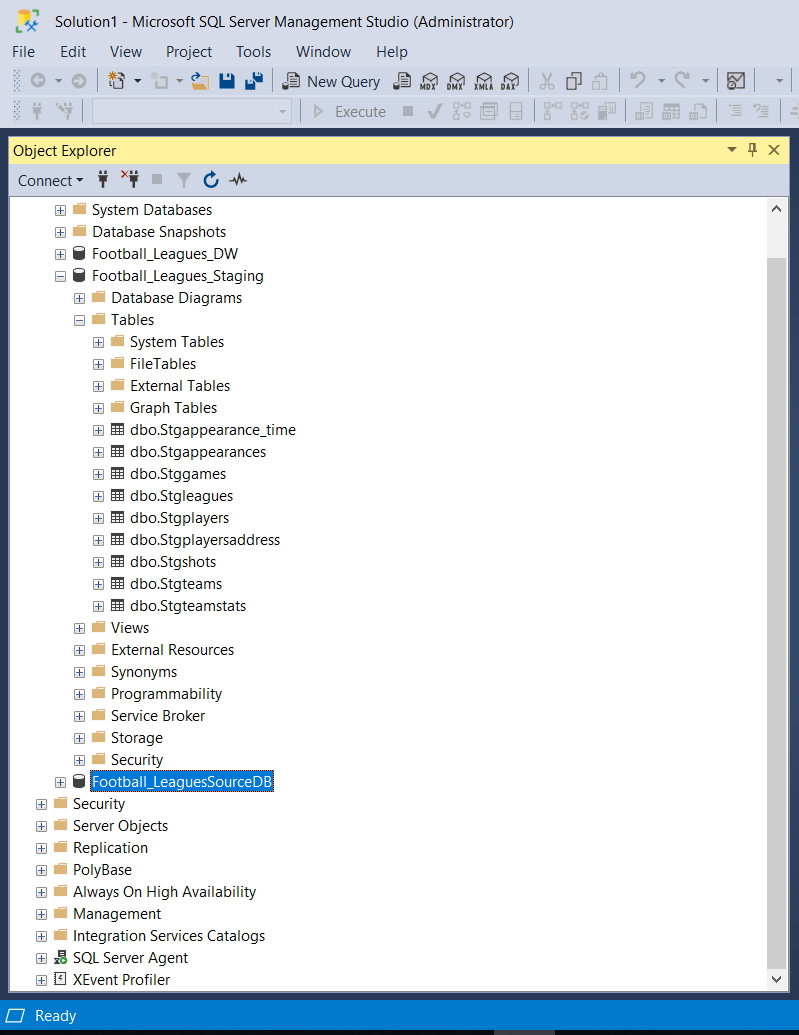
1. Extract

In this step, all the data sources were imported to the staging tables by using the relevant Data connection.

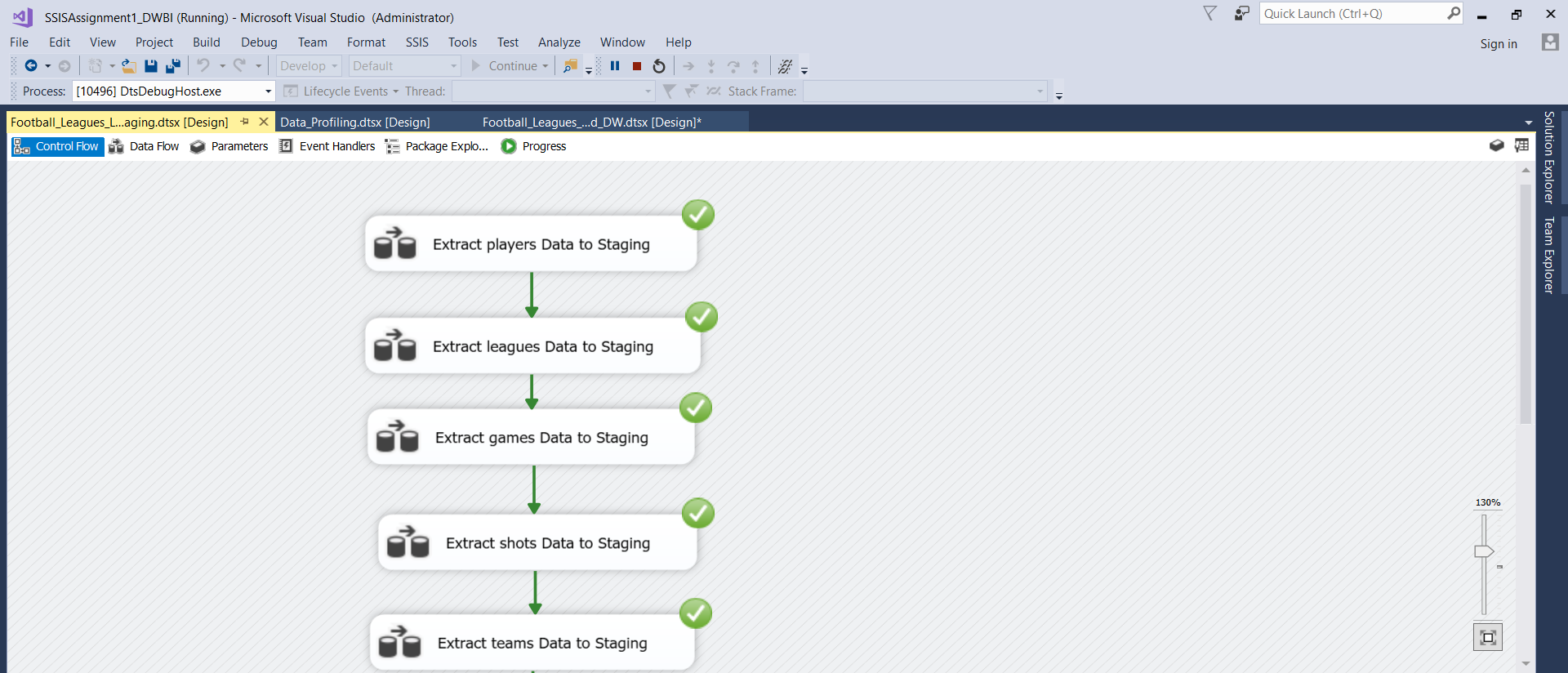
Flat file connection was used for text files and csv files, DB source connection for DB file. All those tables were imported to the Footbal\_Leagues\_Staging DB, which contains the below tables,

1. dbo.Stgappearance\_time
2. dbo.Stgappearances
3. dbo.Stggames
4. dbo.Stgleagues
5. dbo.Stgplayers
6. dbo.Stgplayersaddress
7. dbo.Stgshots
8. dbo.Stgteams
9. dbo.Stgteamstats

* **Snapshot of SSMS Staging Database**

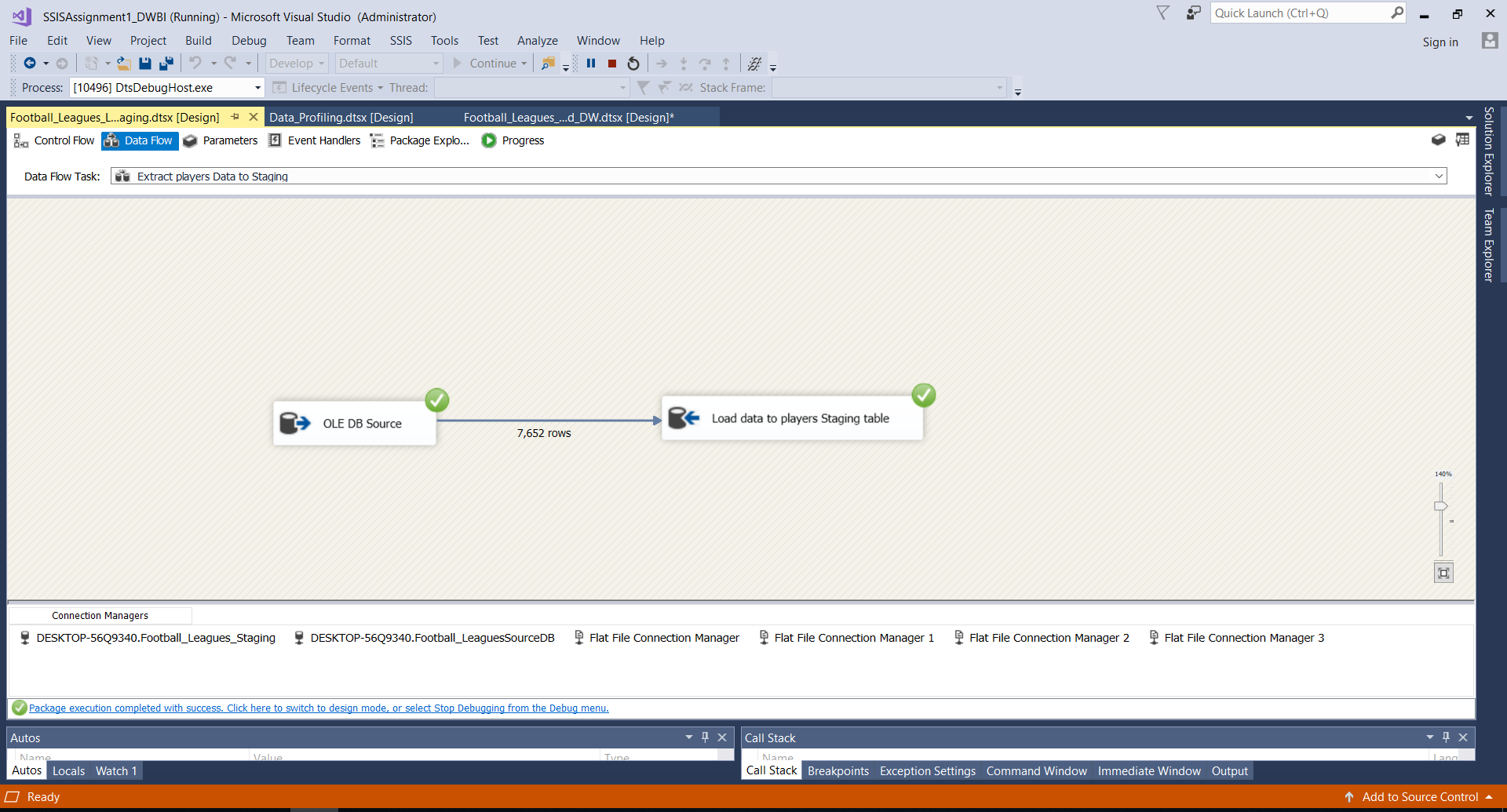


**• Snapshot of Visual Studio Control Flow of Extract**

****

• **Snapshots of several data types of Data Flows**

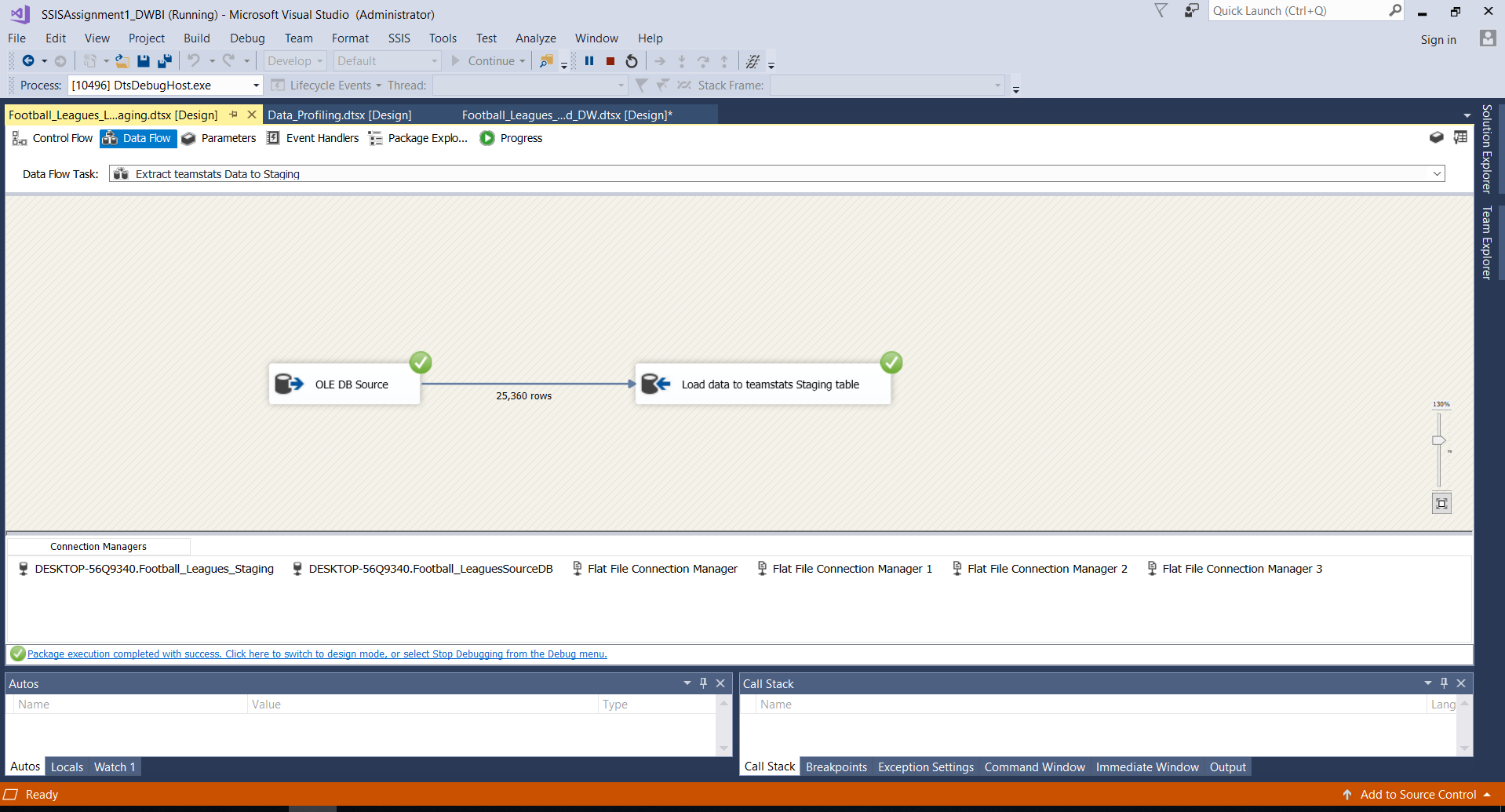
* **PLAYERS DATA STAGING**



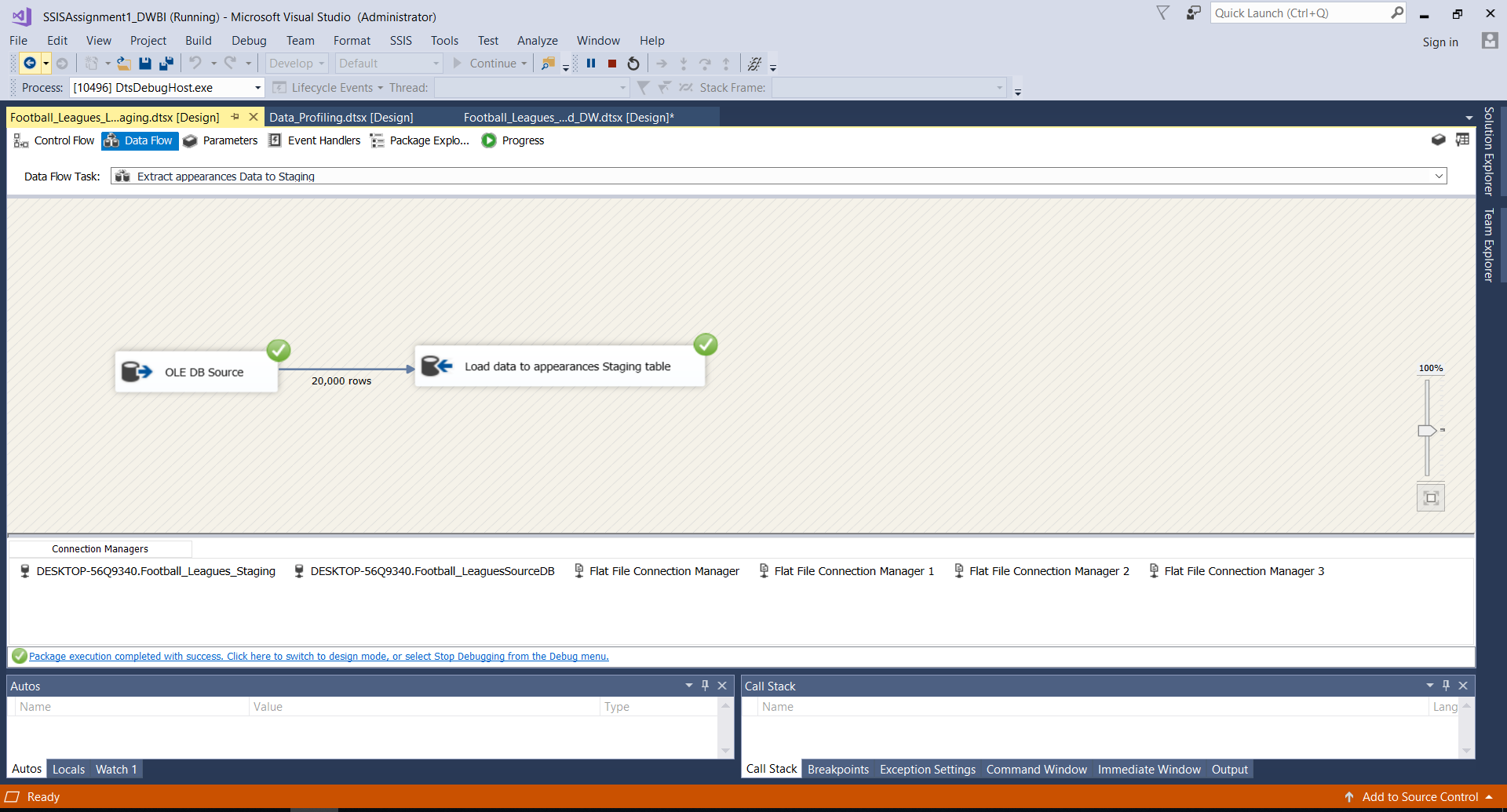
* **GAMES DATA STAGING**

****

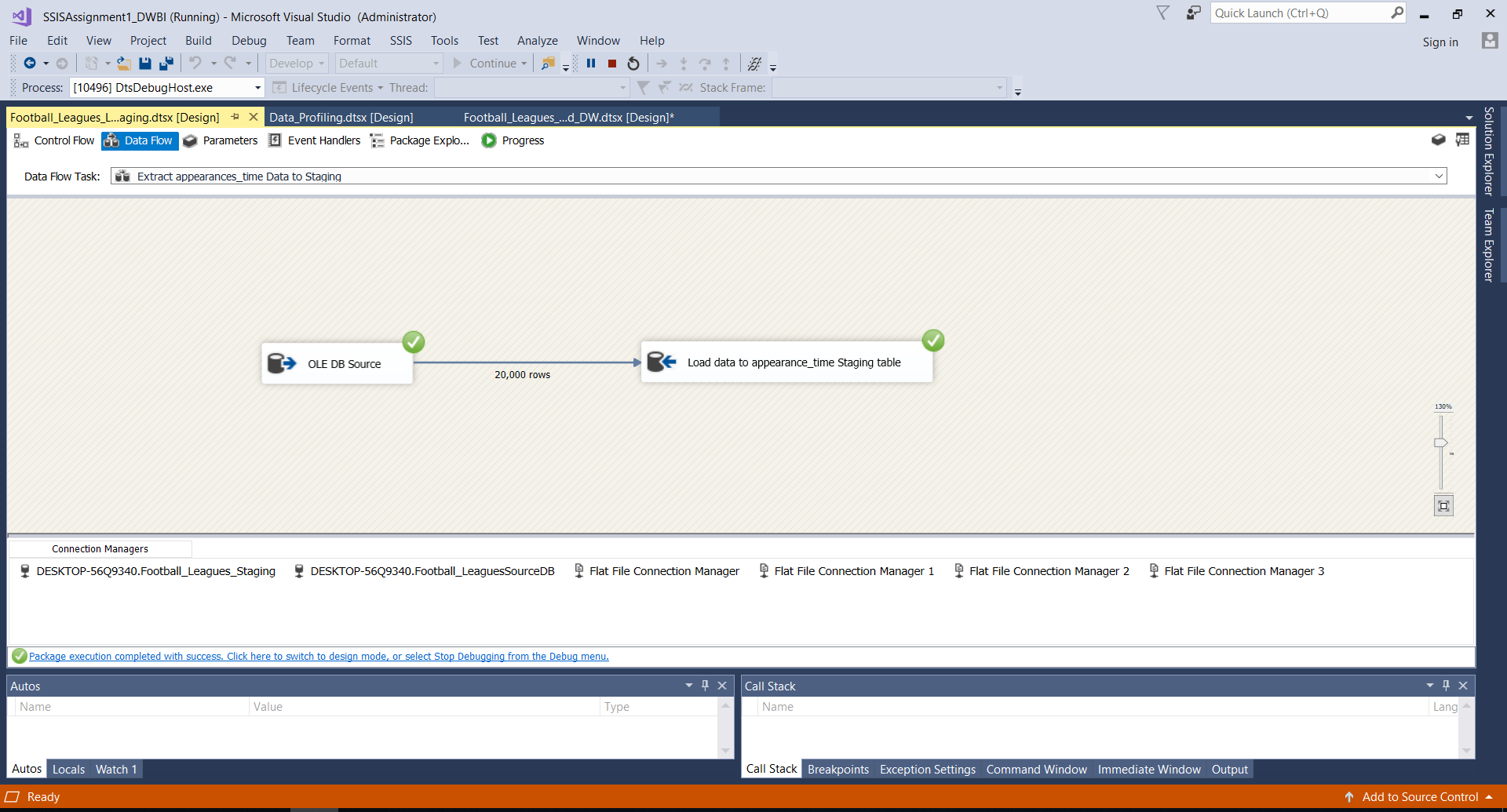
* **TEAMSTATS DATA STAGING**



* **APPEARANCES DATA STAGING**

****

* **APPEARANCE TIME DATA STAGING**

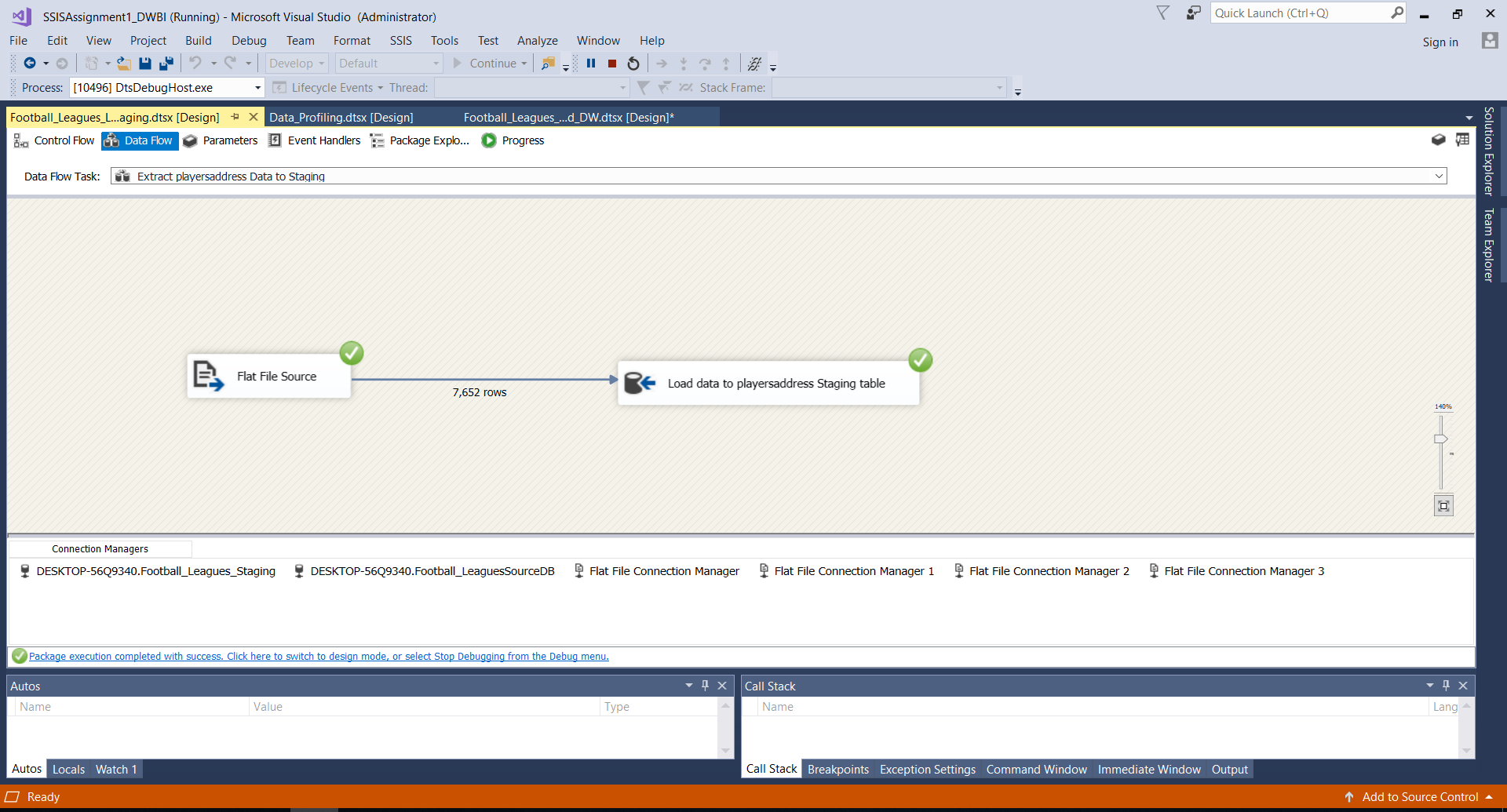
****

• **Snapshots of several data types of Data Flows**

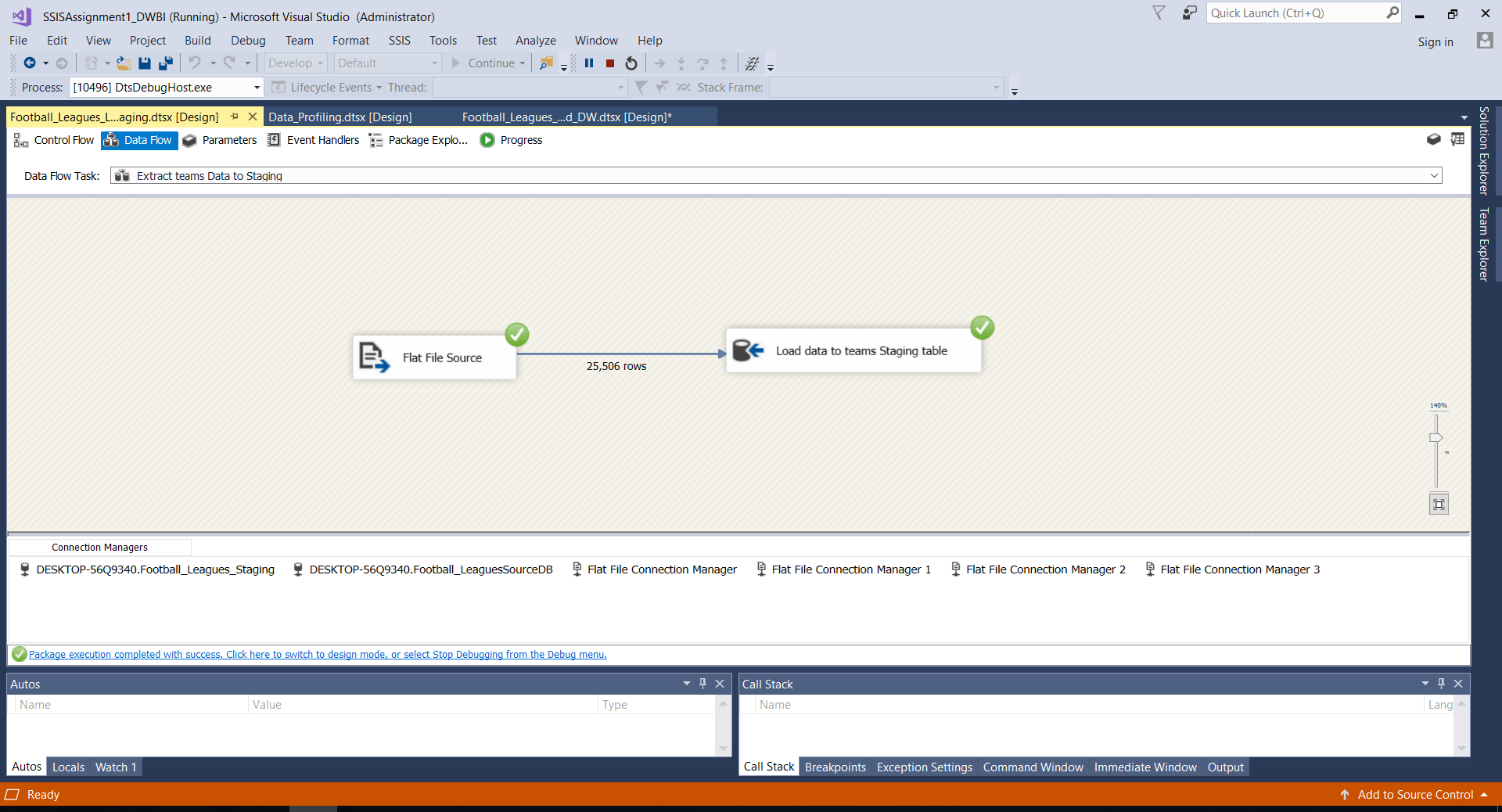
* **LEAGUES DATA STAGING**



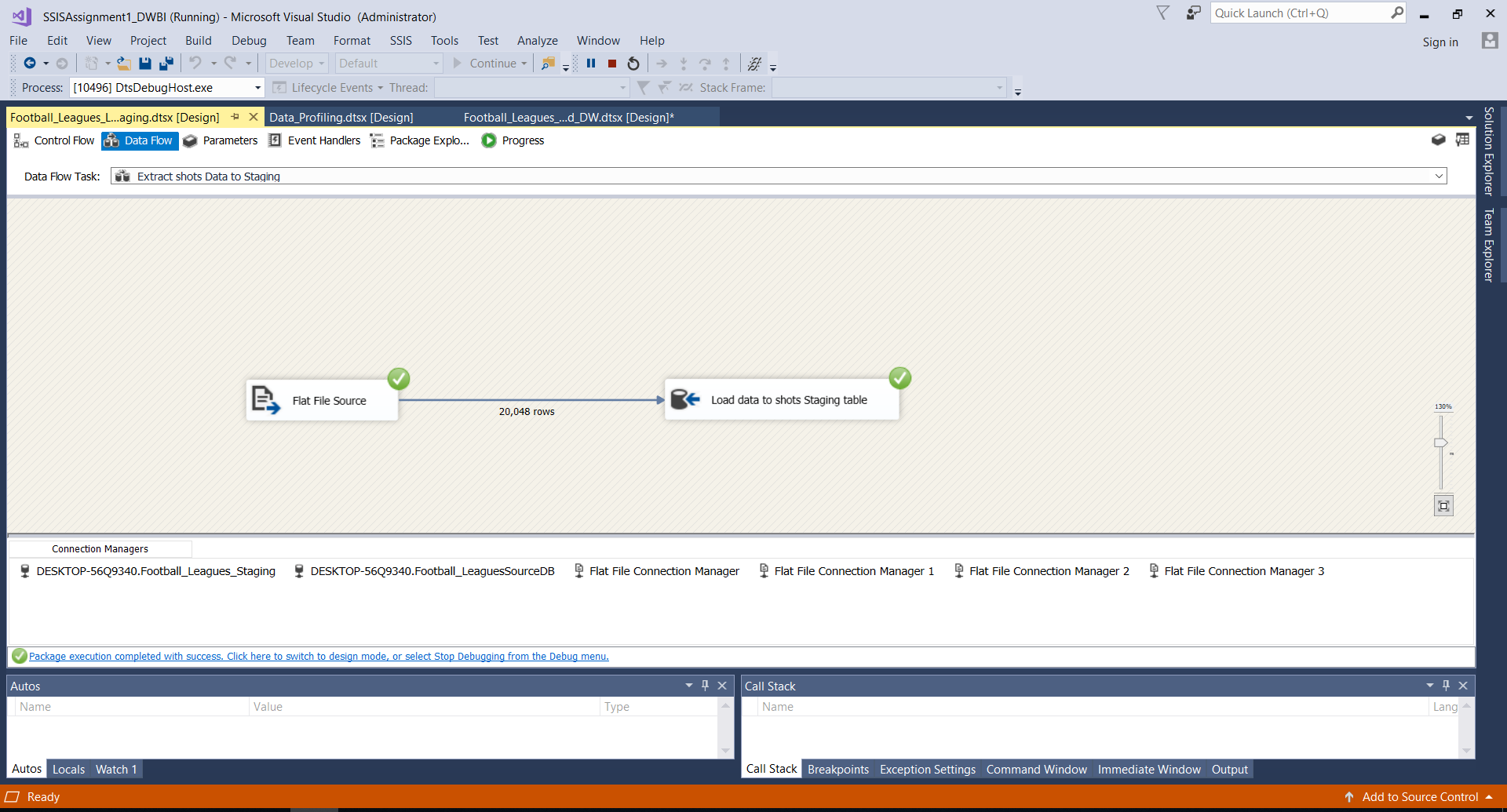
* **PLAYERS ADDRESS DATA STAGING**

****

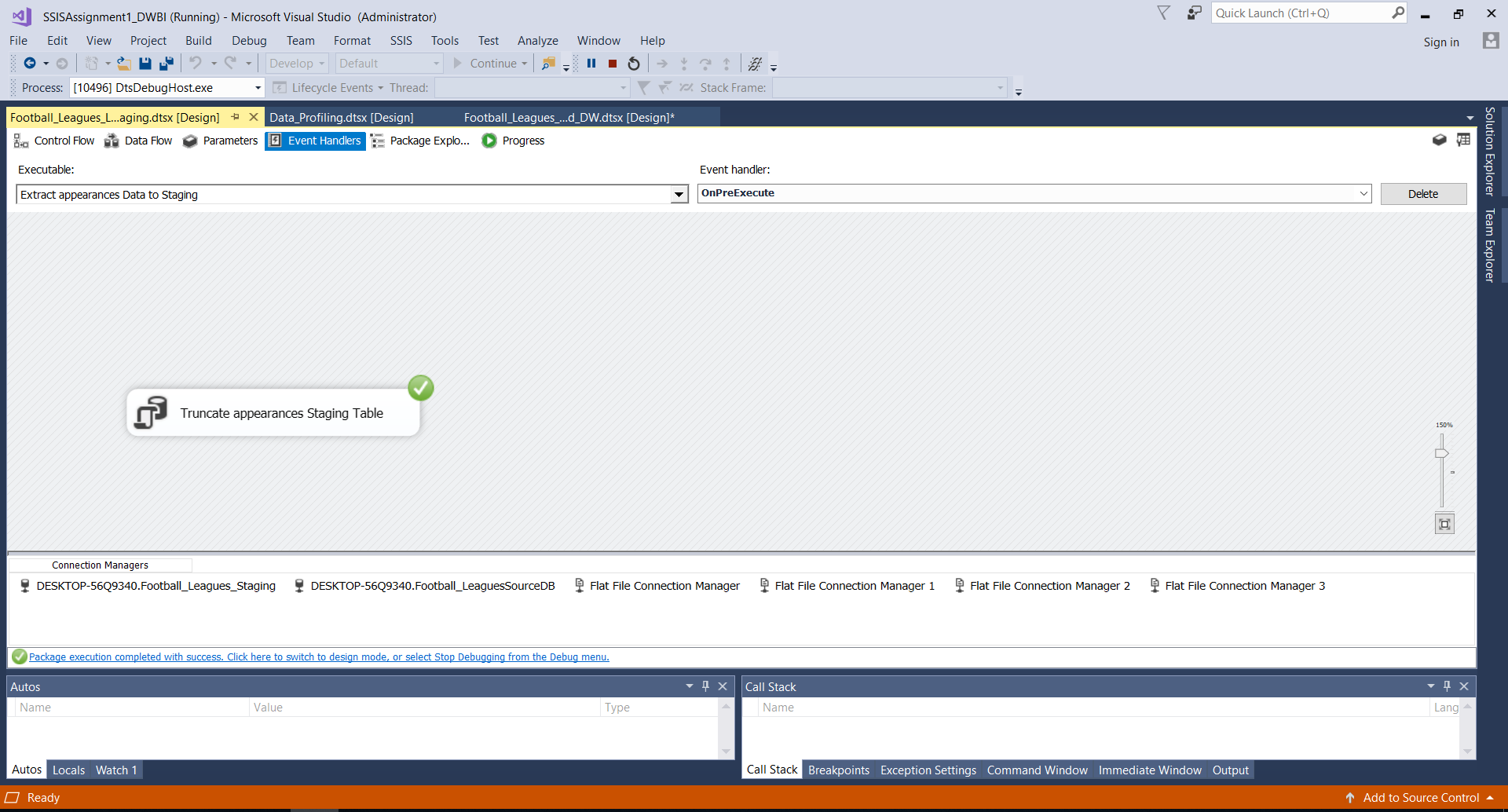
* **TEAMS DATA STAGING**



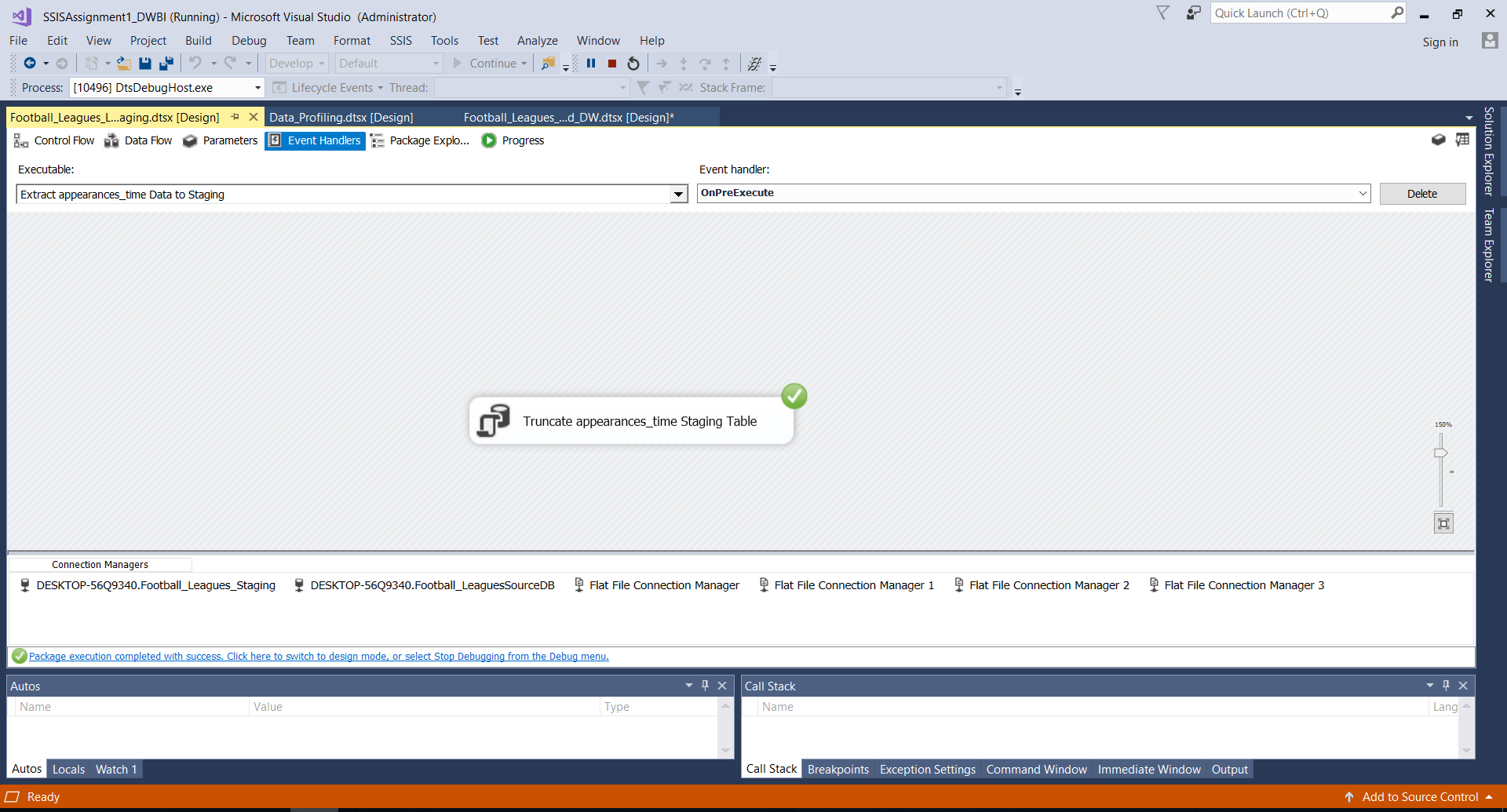
* **SHOTS DATA STAGING**

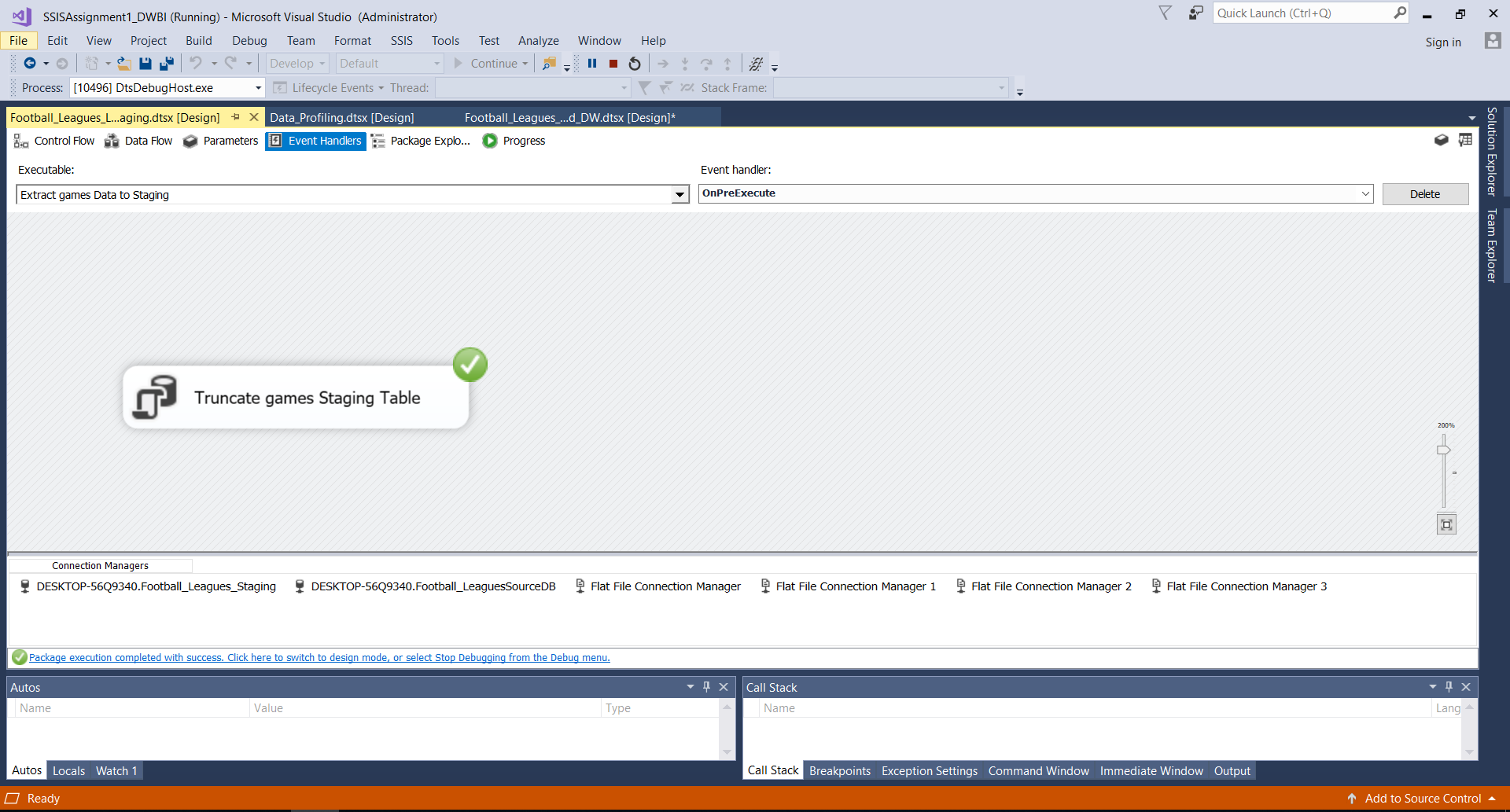
****

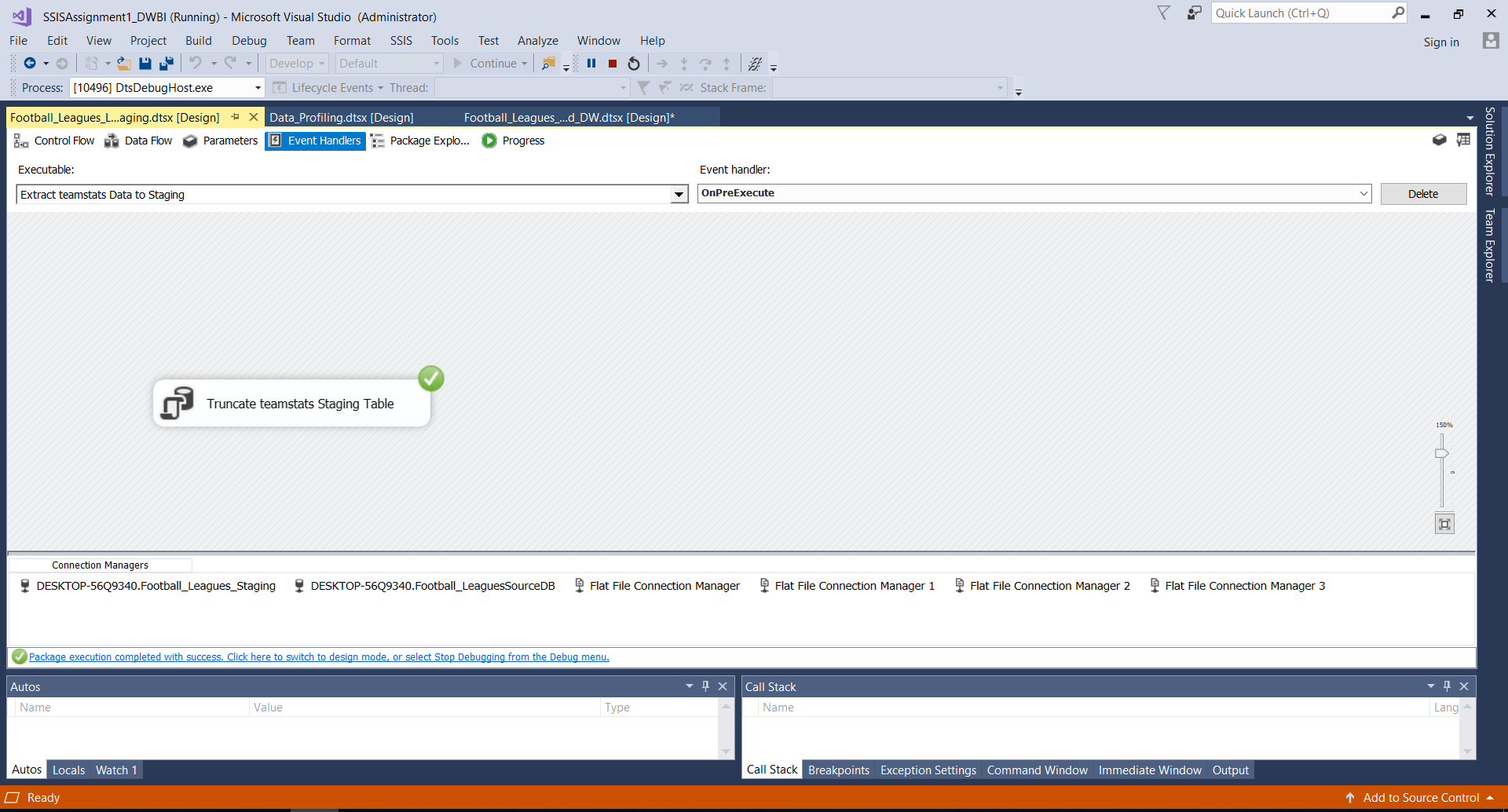
**• Event Handling (Truncate Staging Data)**

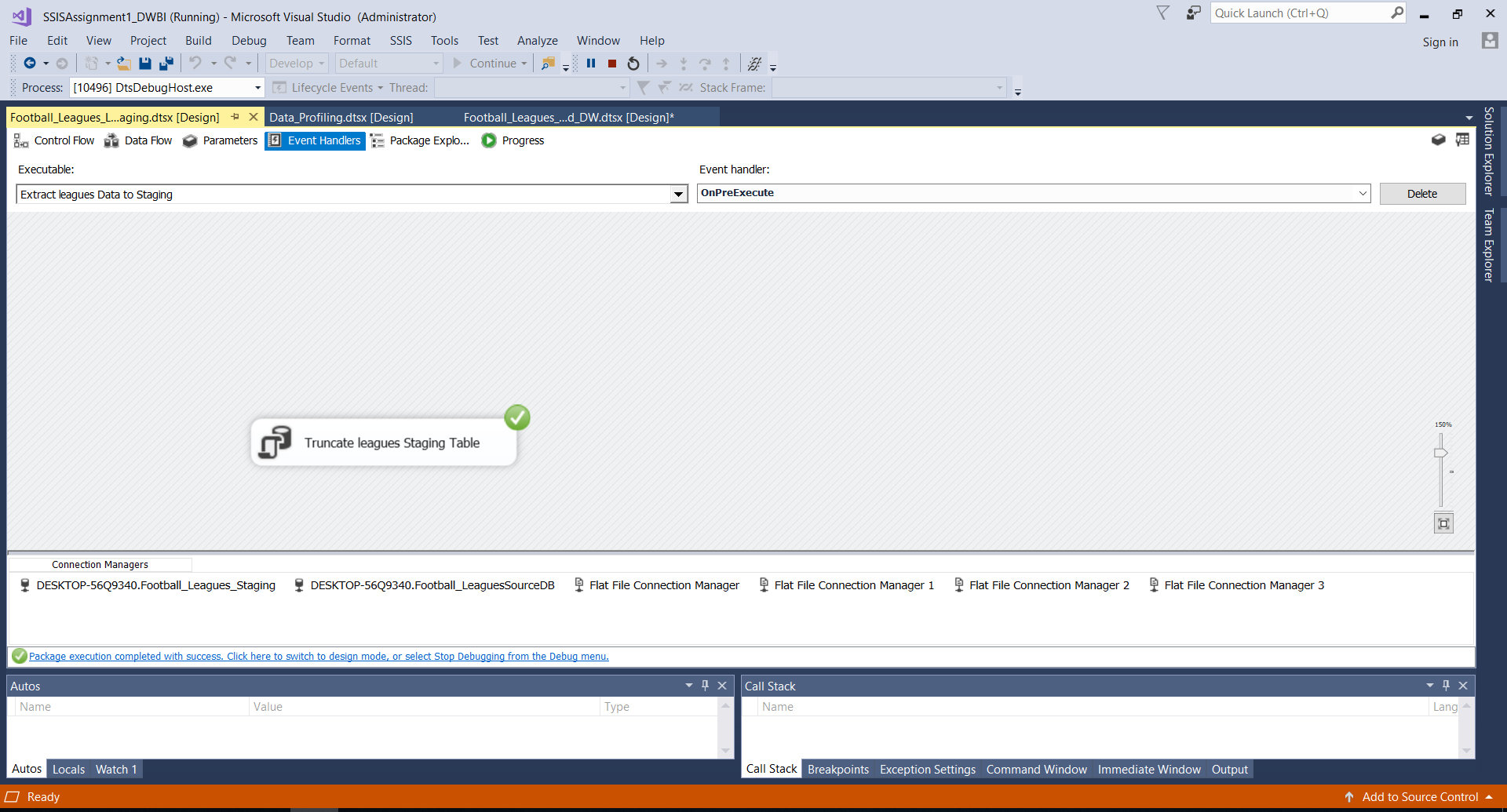


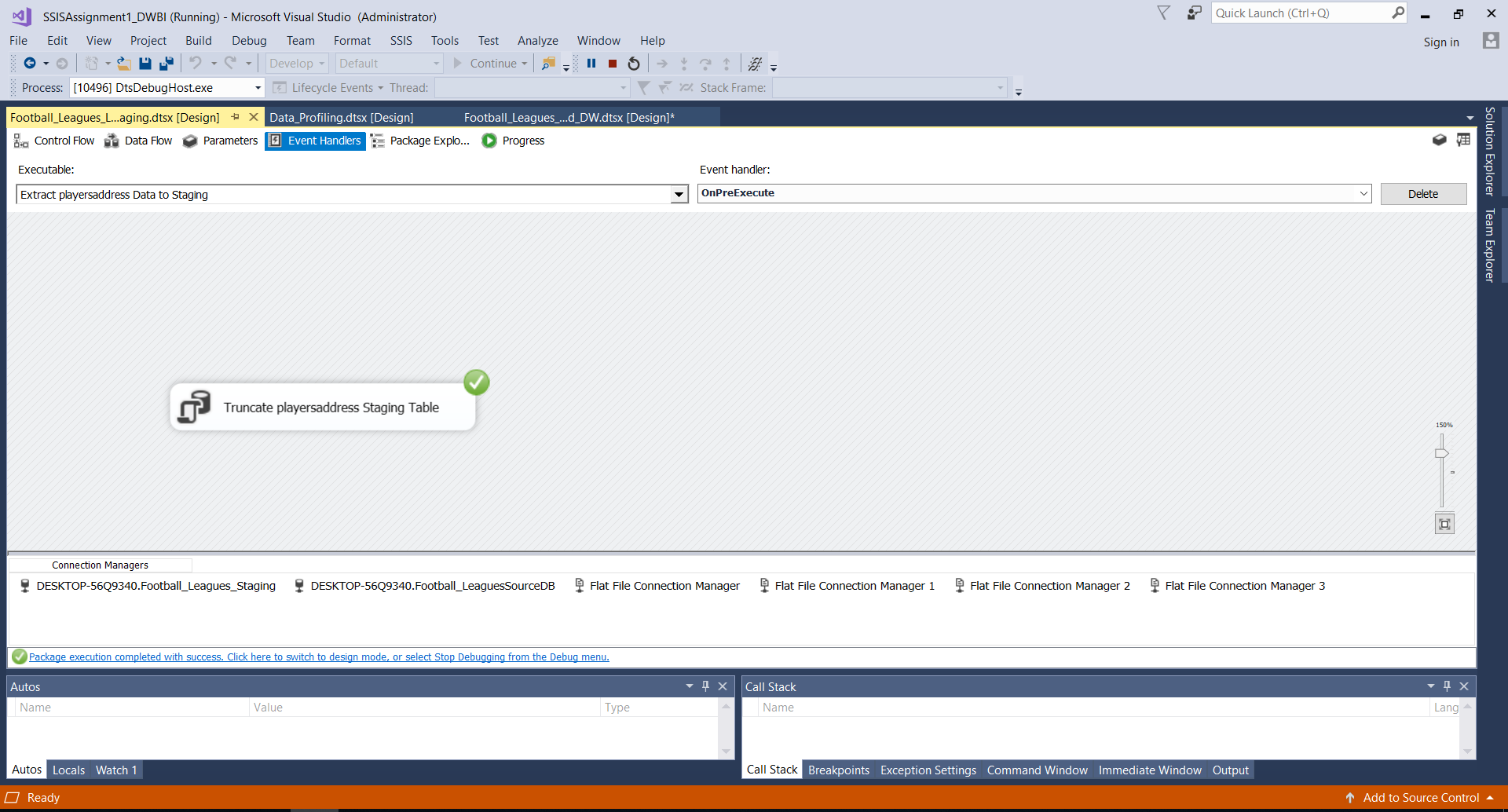


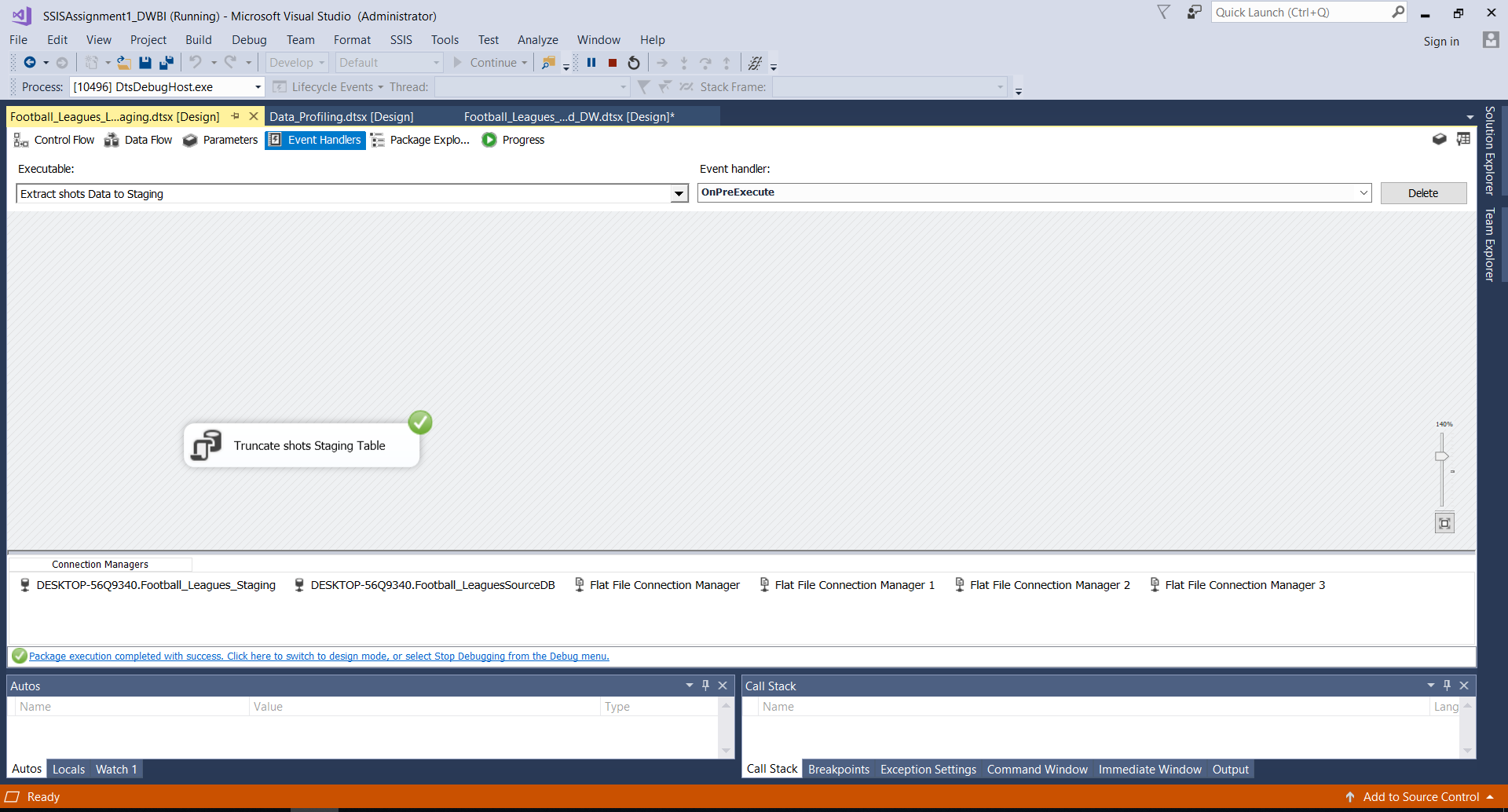


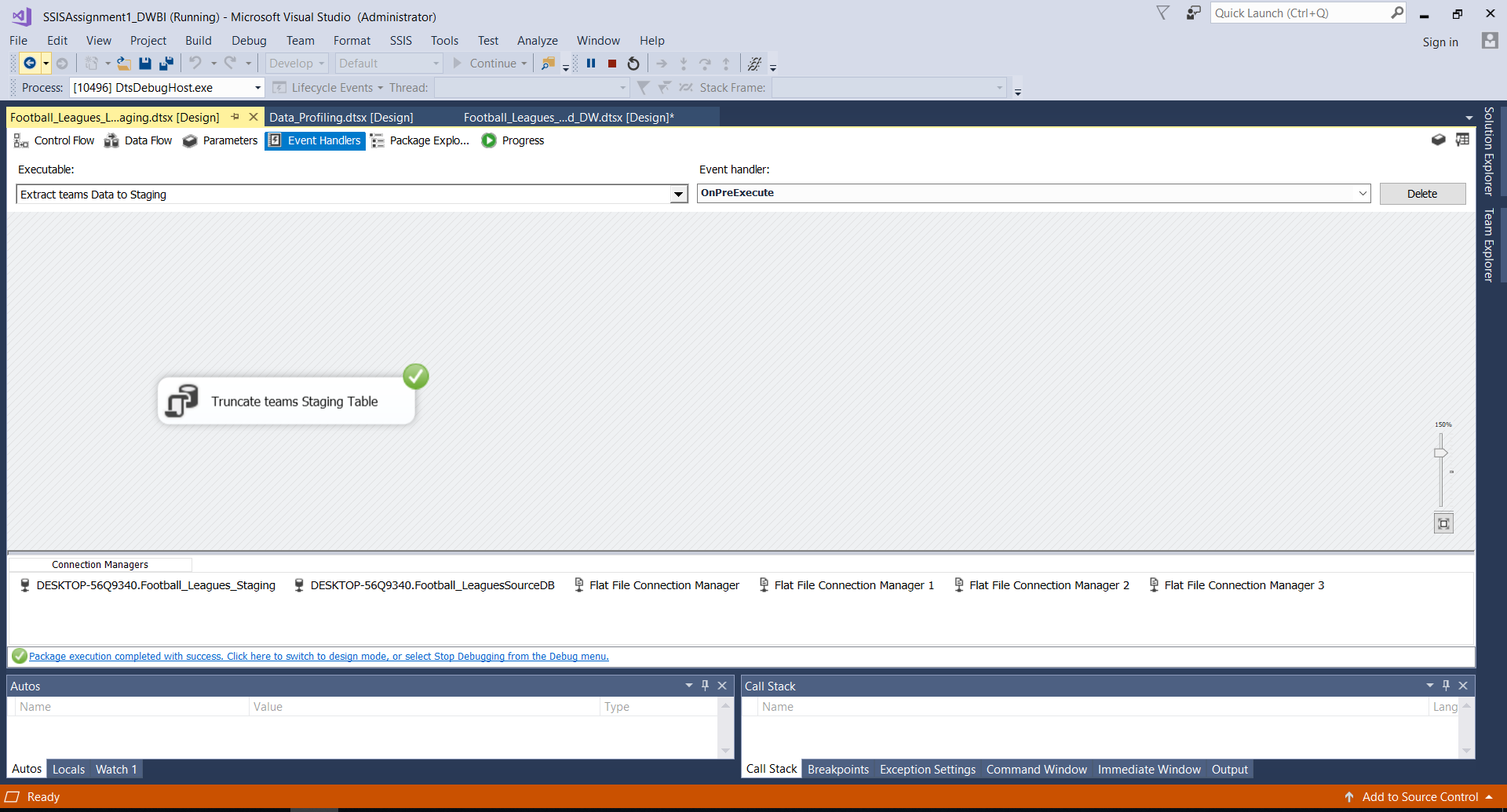






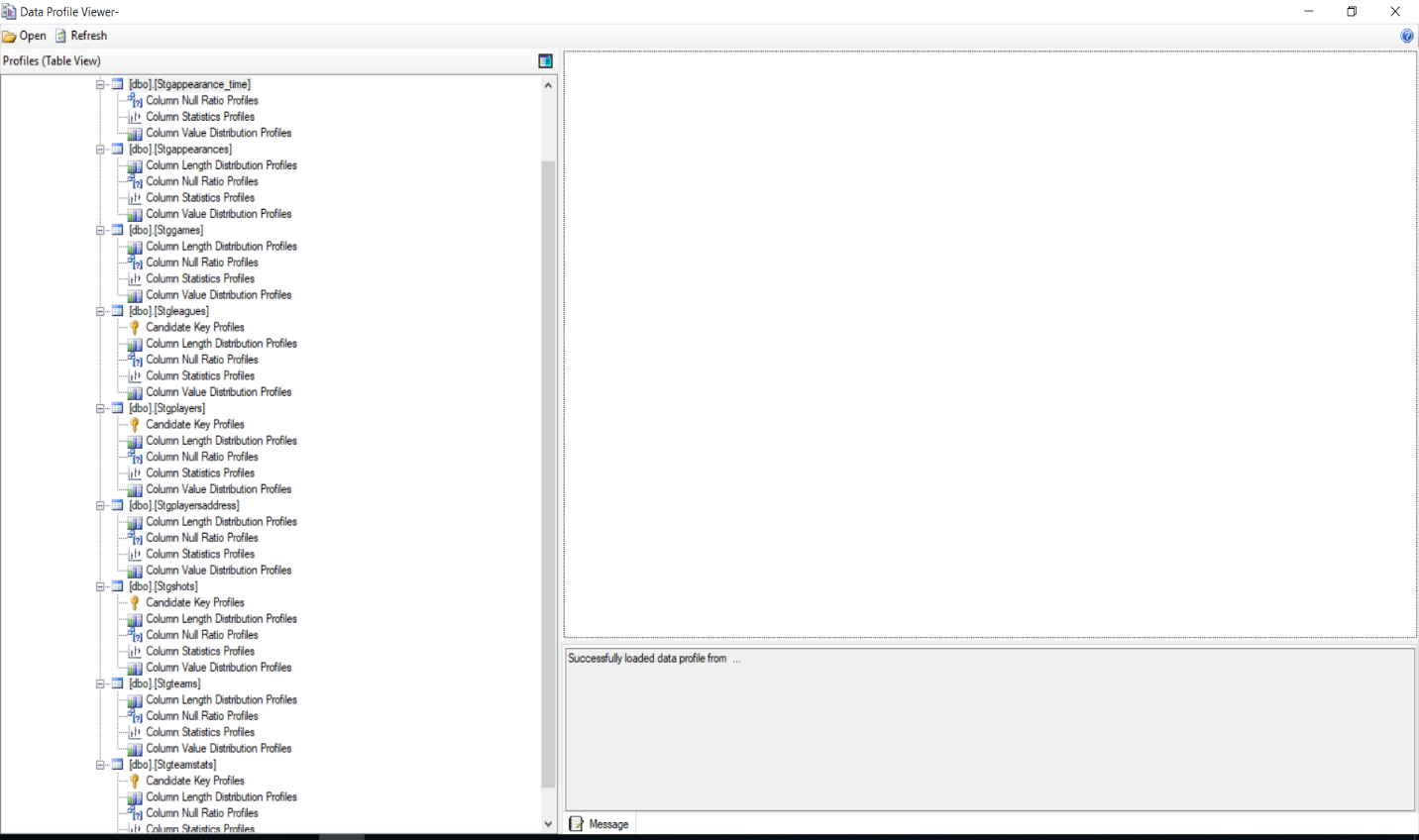
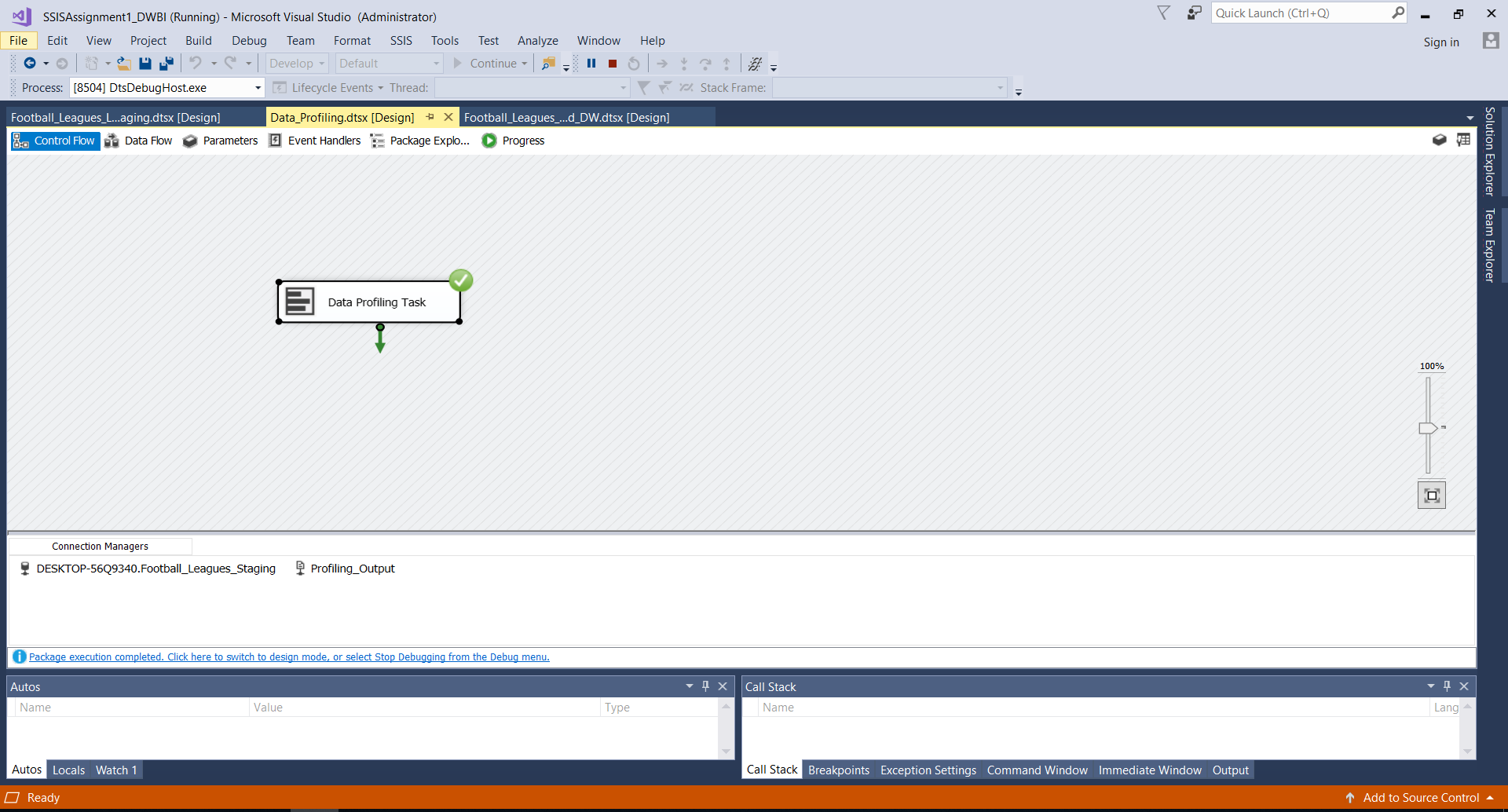






**• Data profiling**

Used one Data\_Profiling package to profiling the staging tables.

****

**3. Transform & Load**

In this step, both the ‘Transform’ and ‘Load’ are done. Firstly, The Dimension tables in the Datawarehouse DB data were created. Then, using the relevant components, data from the staging tables was loaded into the warehouse tables, Football\_Leagues\_DW, which contains the below tables,

1. dbo.DimDate
2. dbo.Dimgames
3. dbo.Dimleagues
4. dbo.Dimplayers
5. dbo.Dimshots
6. dbo.Dimteamstats
7. dbo.Factappearances

**Used Transformation Tasks**

1. Lookups

Mainly lookups used for Appearances fact table transform and loading process. These lookups are Date lookup, game lookup, players lookup, leagues ookup, teamstats lookup.

1. Derived Columns

* Replace Null gender Values and title values in players Dim Table.
* Calculate txn\_process\_time in appearances Fact Table.
* Modified date field in appearances Fact Table.

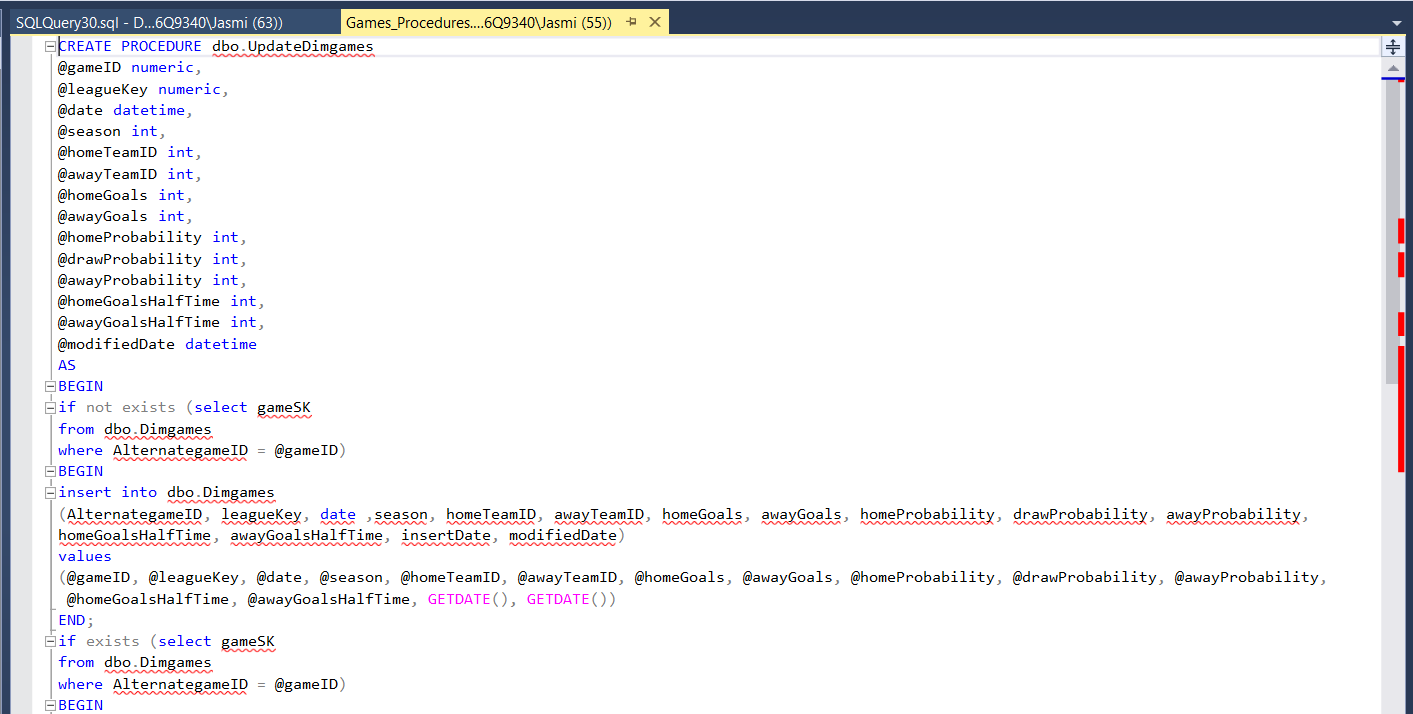
1. Union

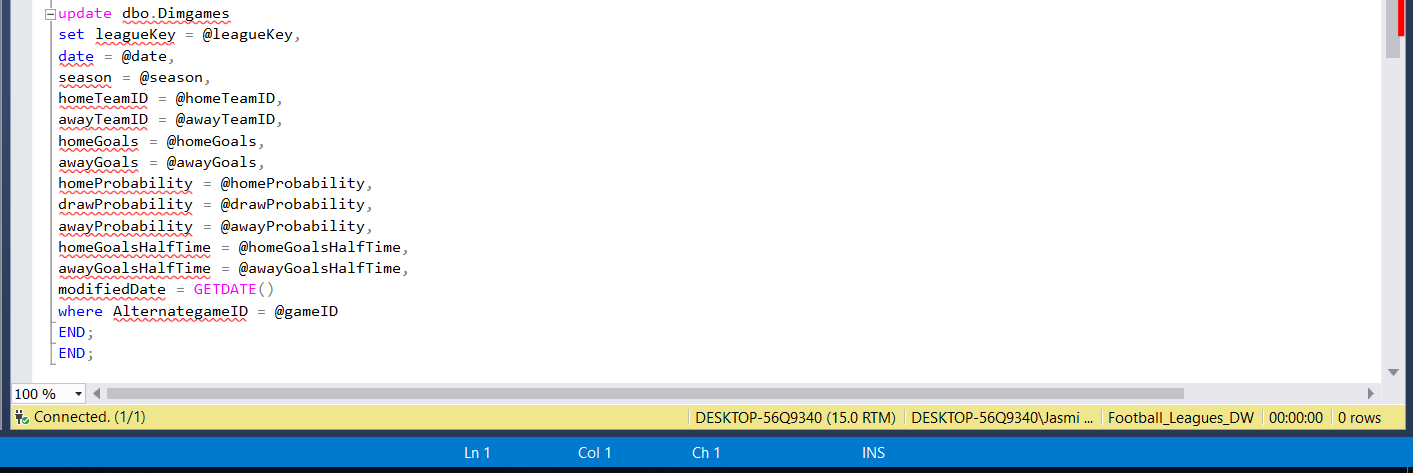
* Union is used in the Extract step to combine and get all the data from both players and players address data files.

1. Sort and Merge

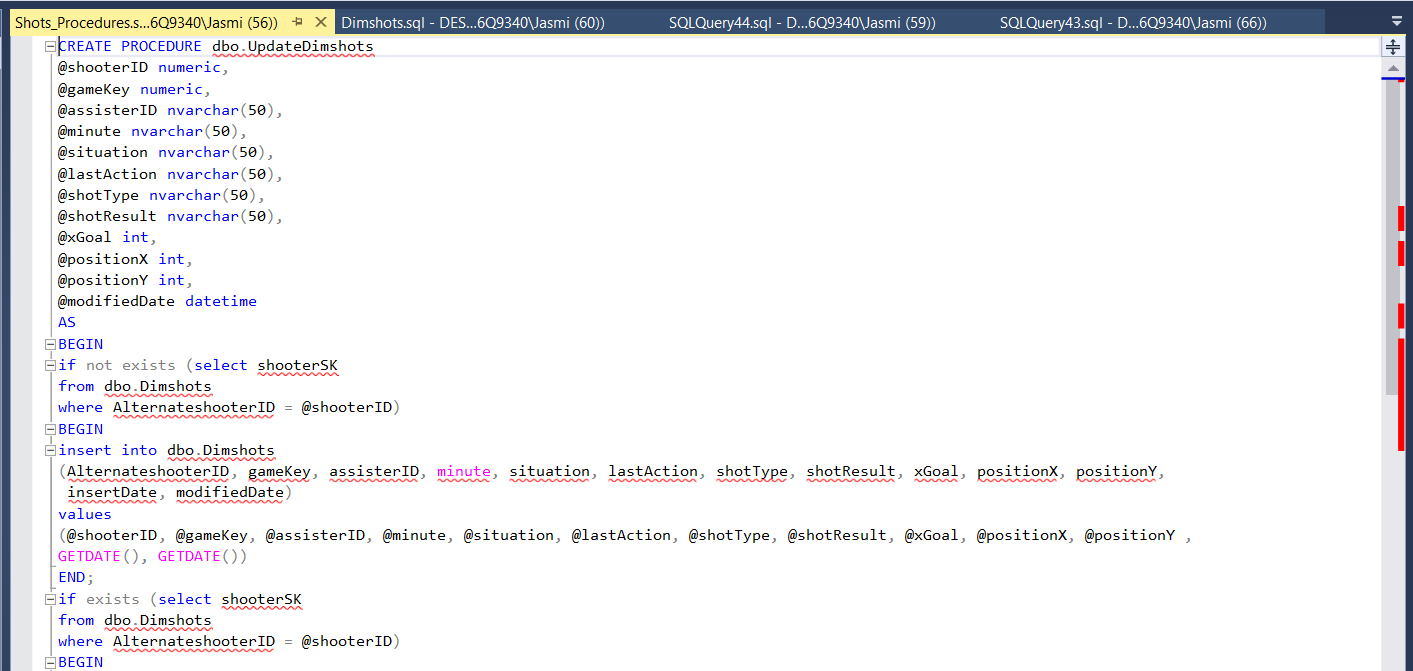
* ‘Sort’ is used sort out the players and players address data and they are merged ‘Merge’ using playerID.
* ‘Sort’ is used sort out the teams and teamstats data and they are merged ‘Merge’ using teamID.
* ‘Sort’ is used sort out the shots and games dimesion data and they are merged ‘Merge’ using gameSK and AlternategameID.
* ‘Sort’ is used sort out the appearances and appearance\_time data and they are merged ‘Merge’ using appearncesID.
* ‘Sort’ is used sort out the games and leagues dimesion data and they are merged ‘Merge’ using leagueSK and AlternateleagueID.
* **Update Functions**

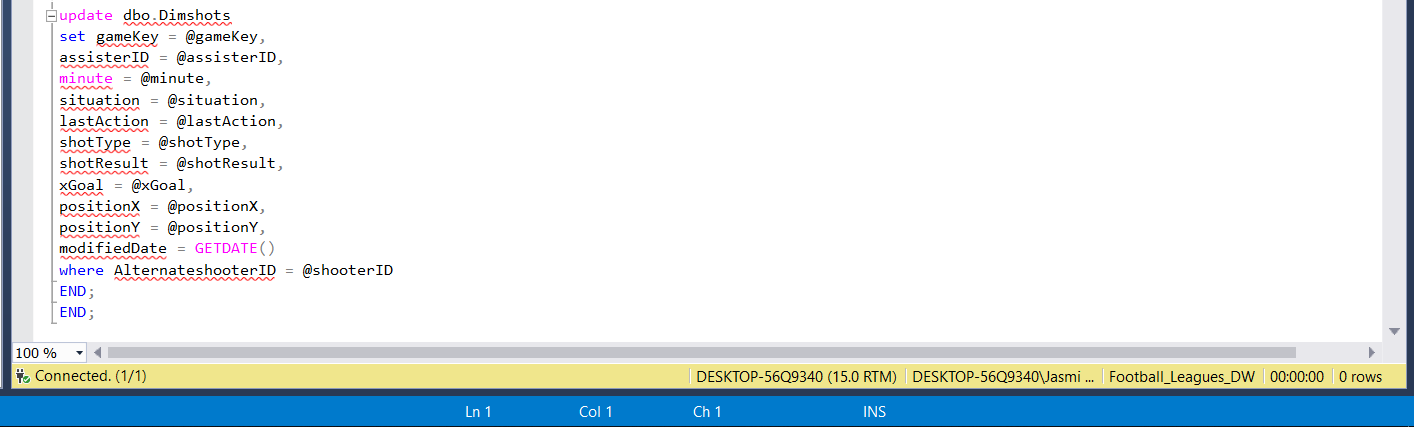
**dbo.UpdateDimgames**

****

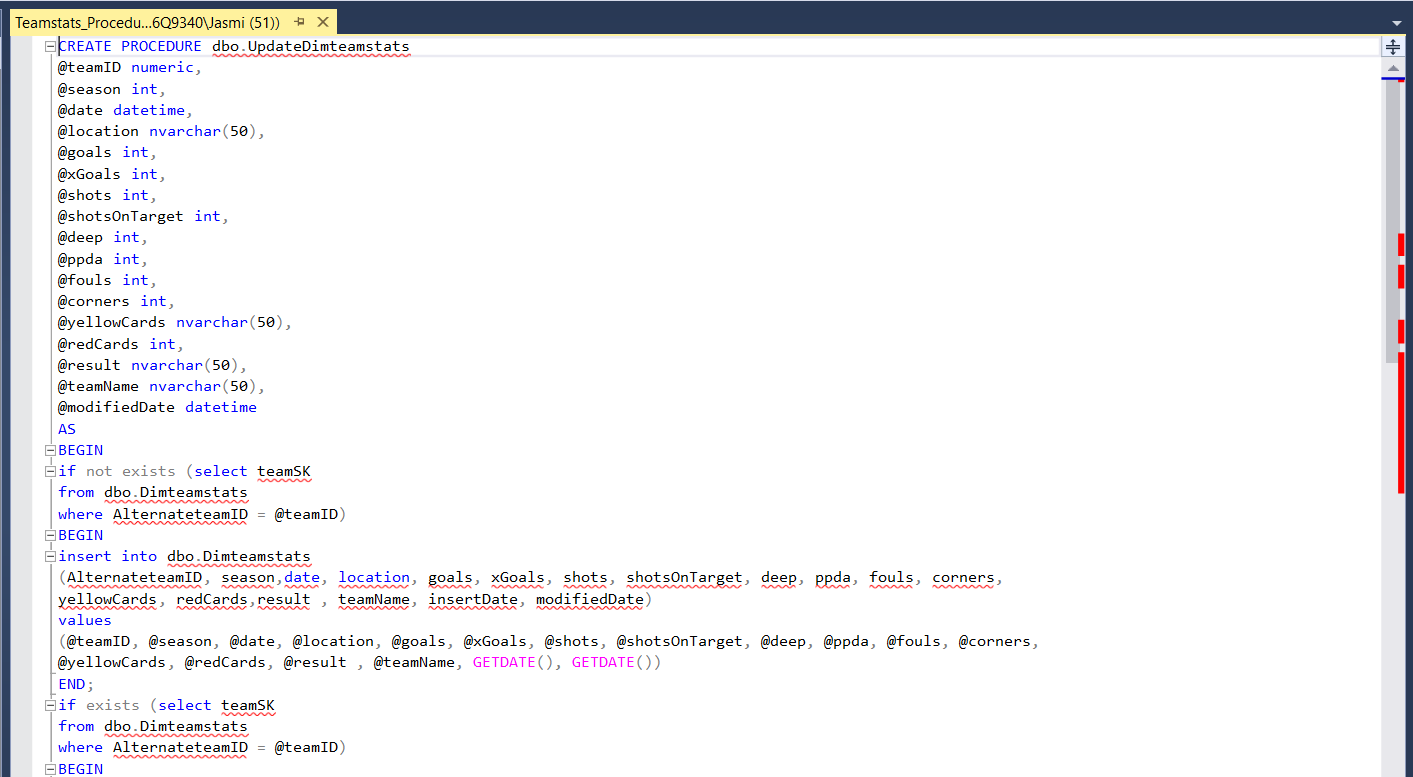
****

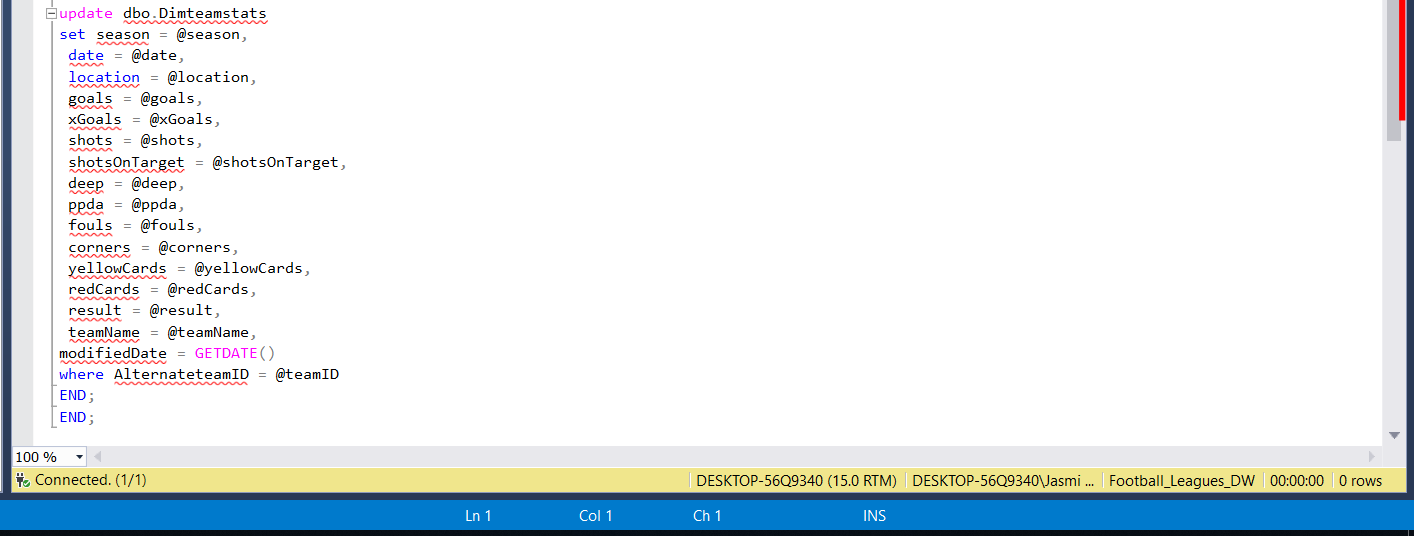
**dbo.UpdateDimshots**

****

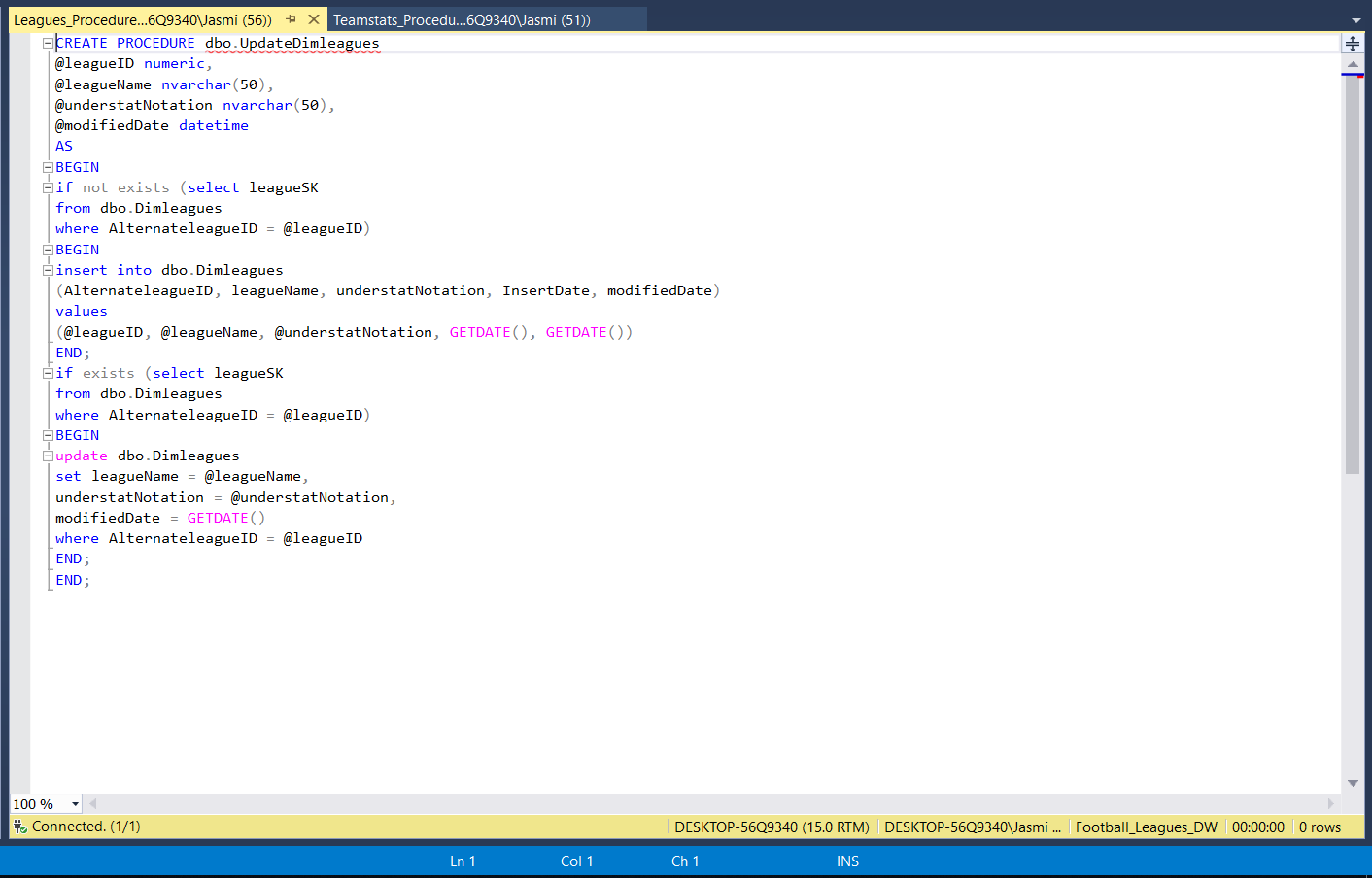
****

**dbo.UpadteDimteamstats**

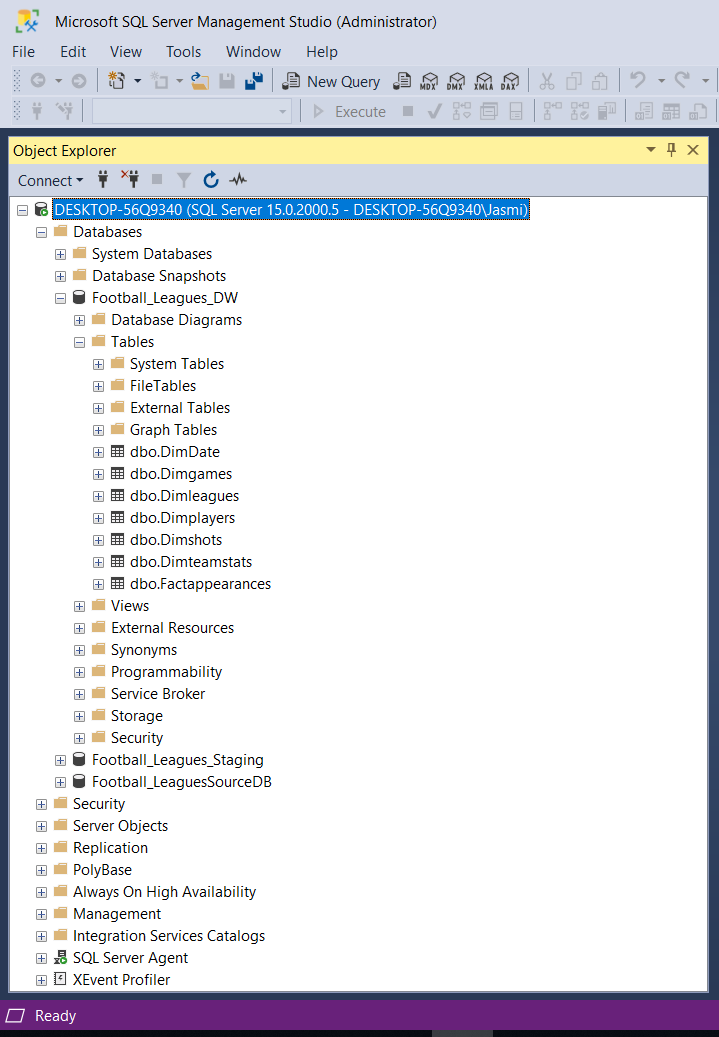
****

****

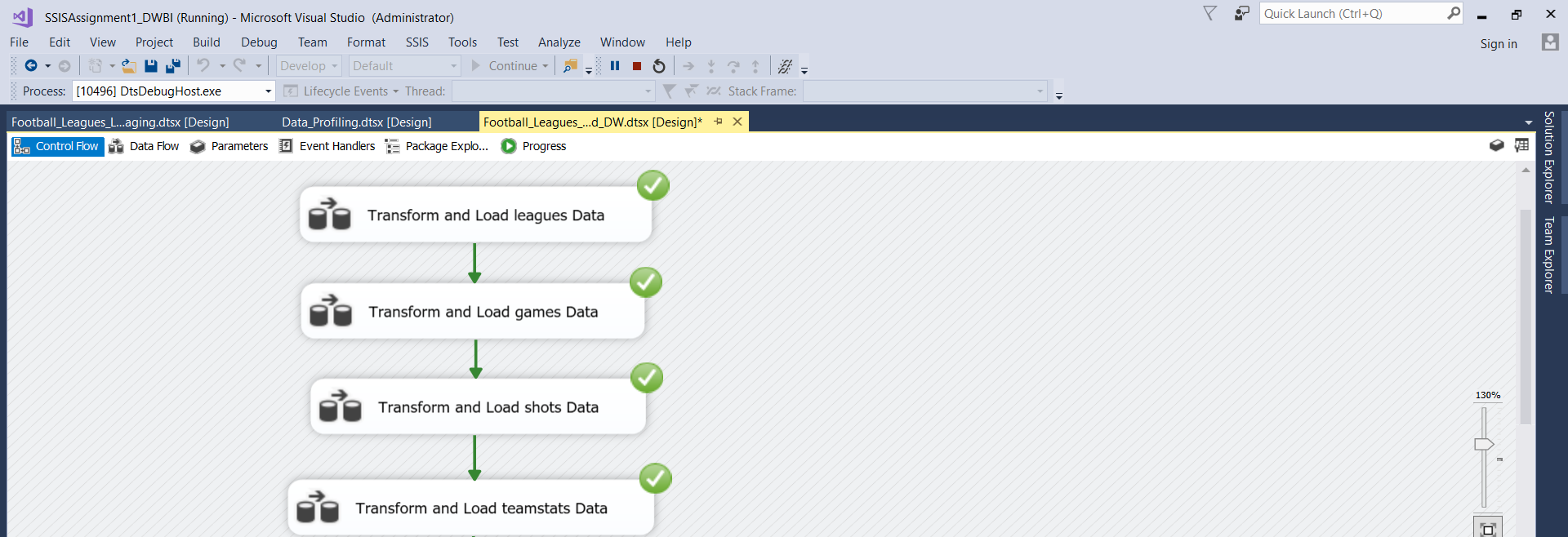
**dbo.UpdateDimleagues**

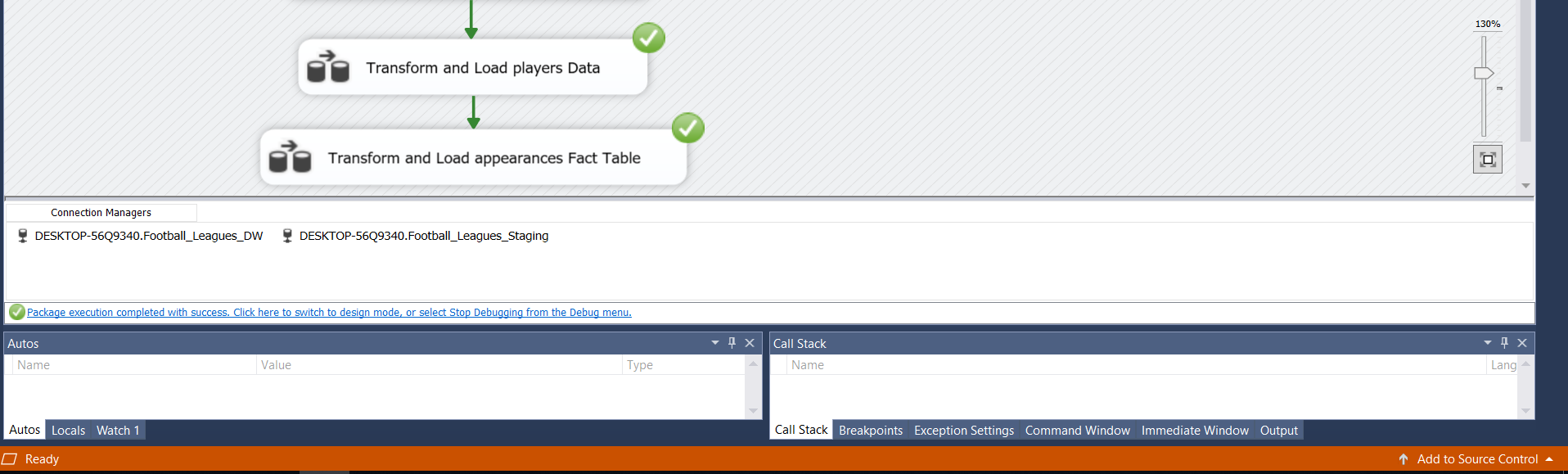
****

**• Snapshot of SQL server Data warehouse Database**

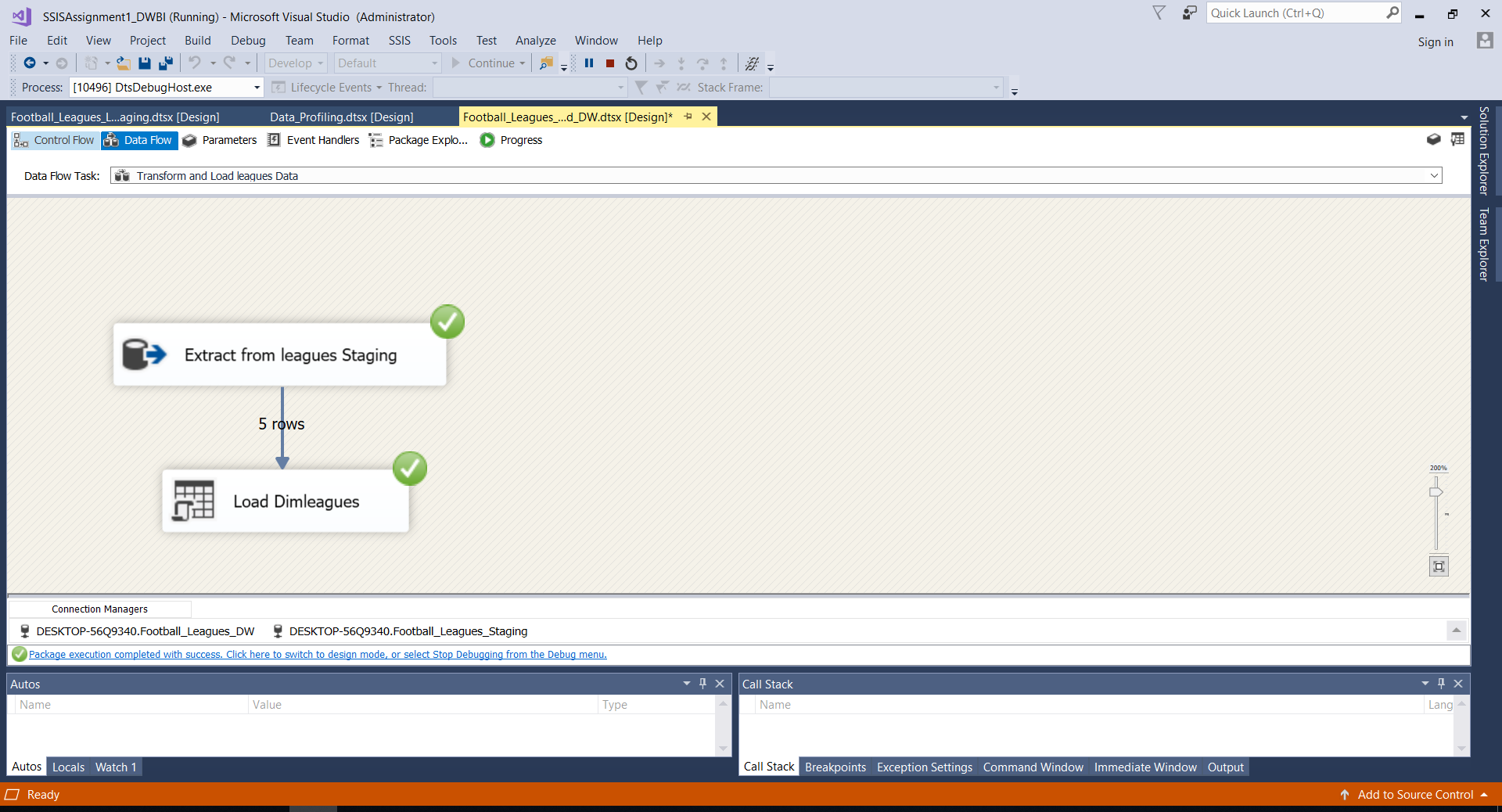
****

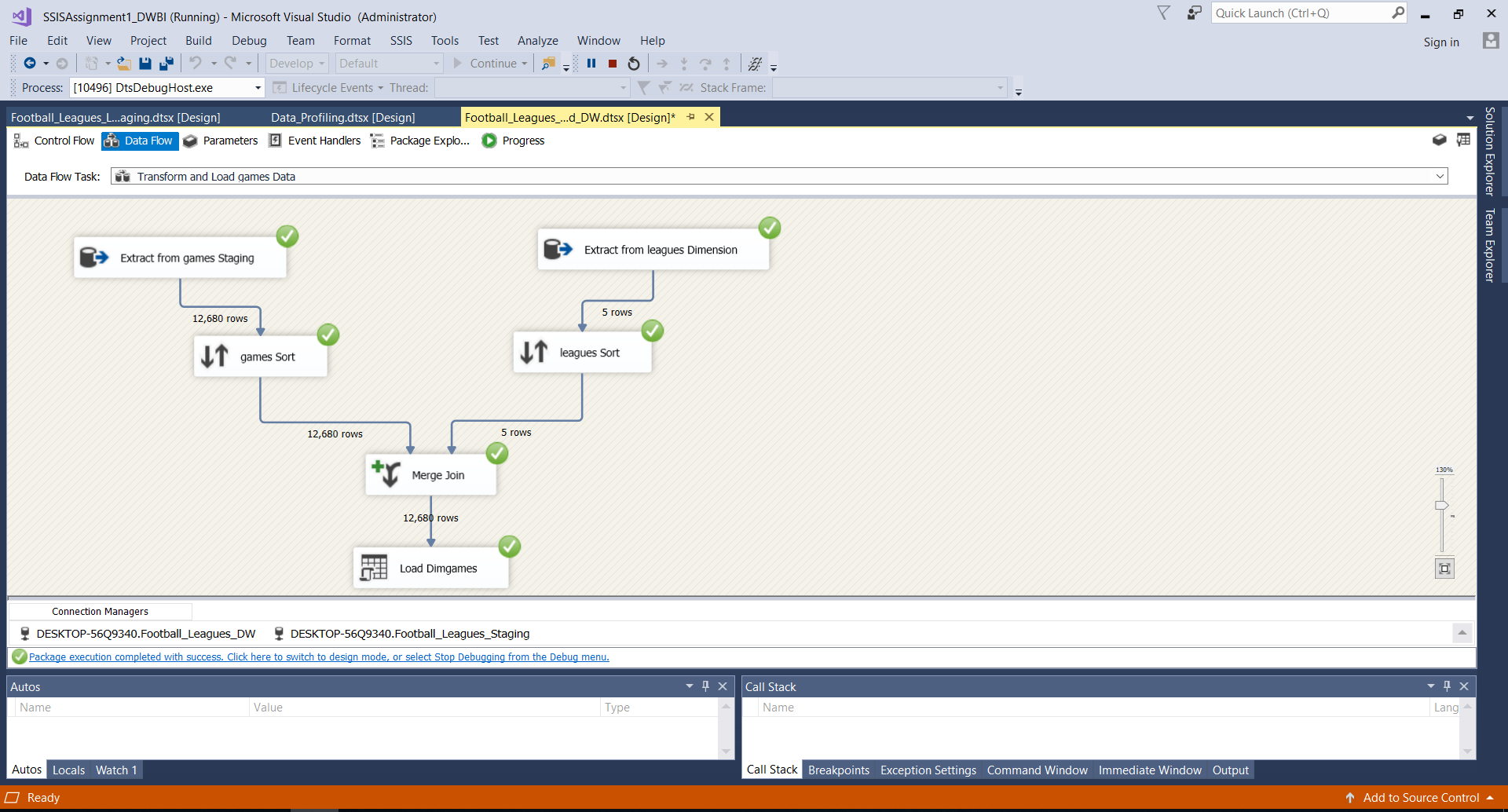
* **Snapshot of Visual Studio Control Flow of Extraction**

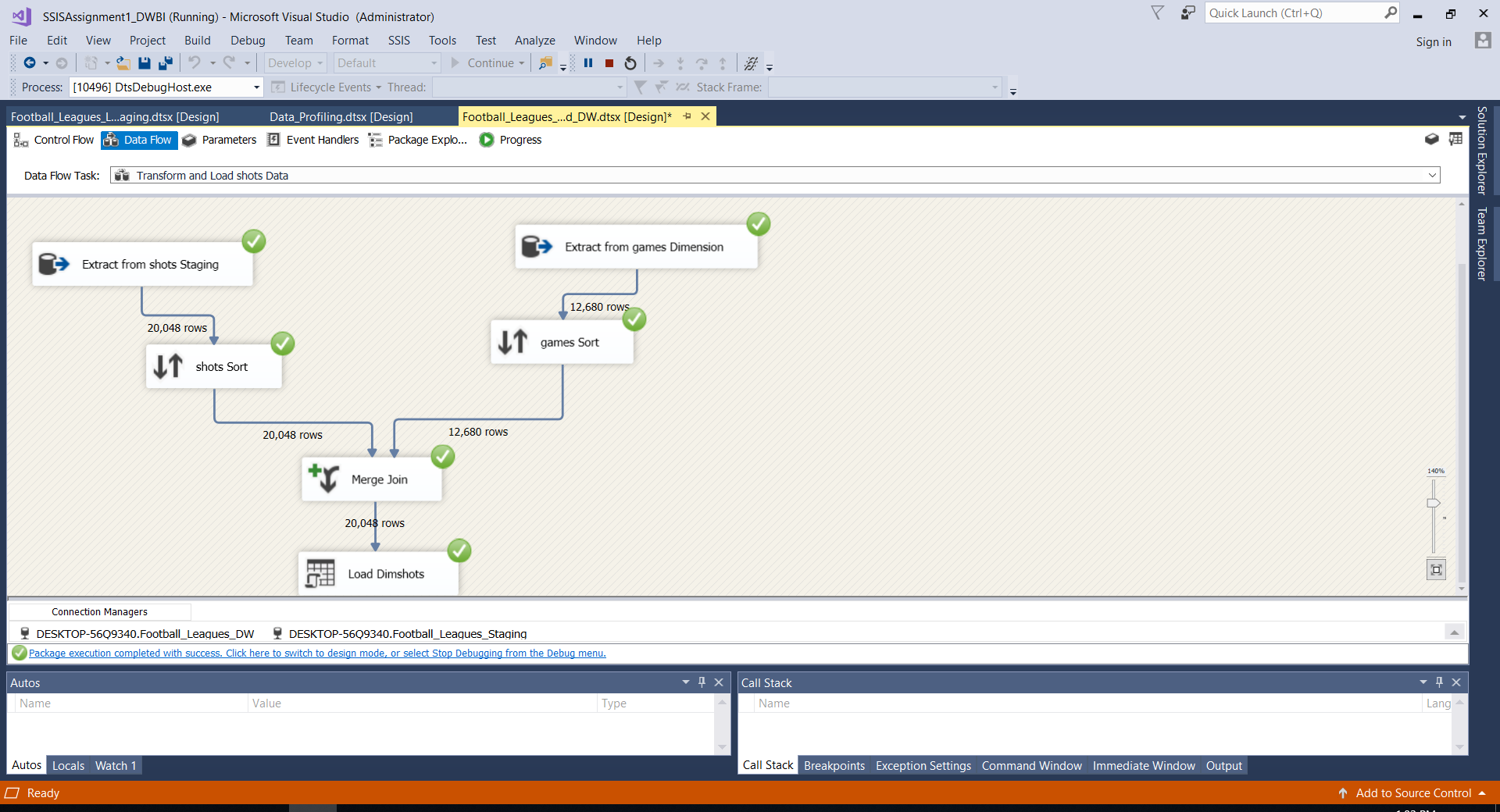
****

****

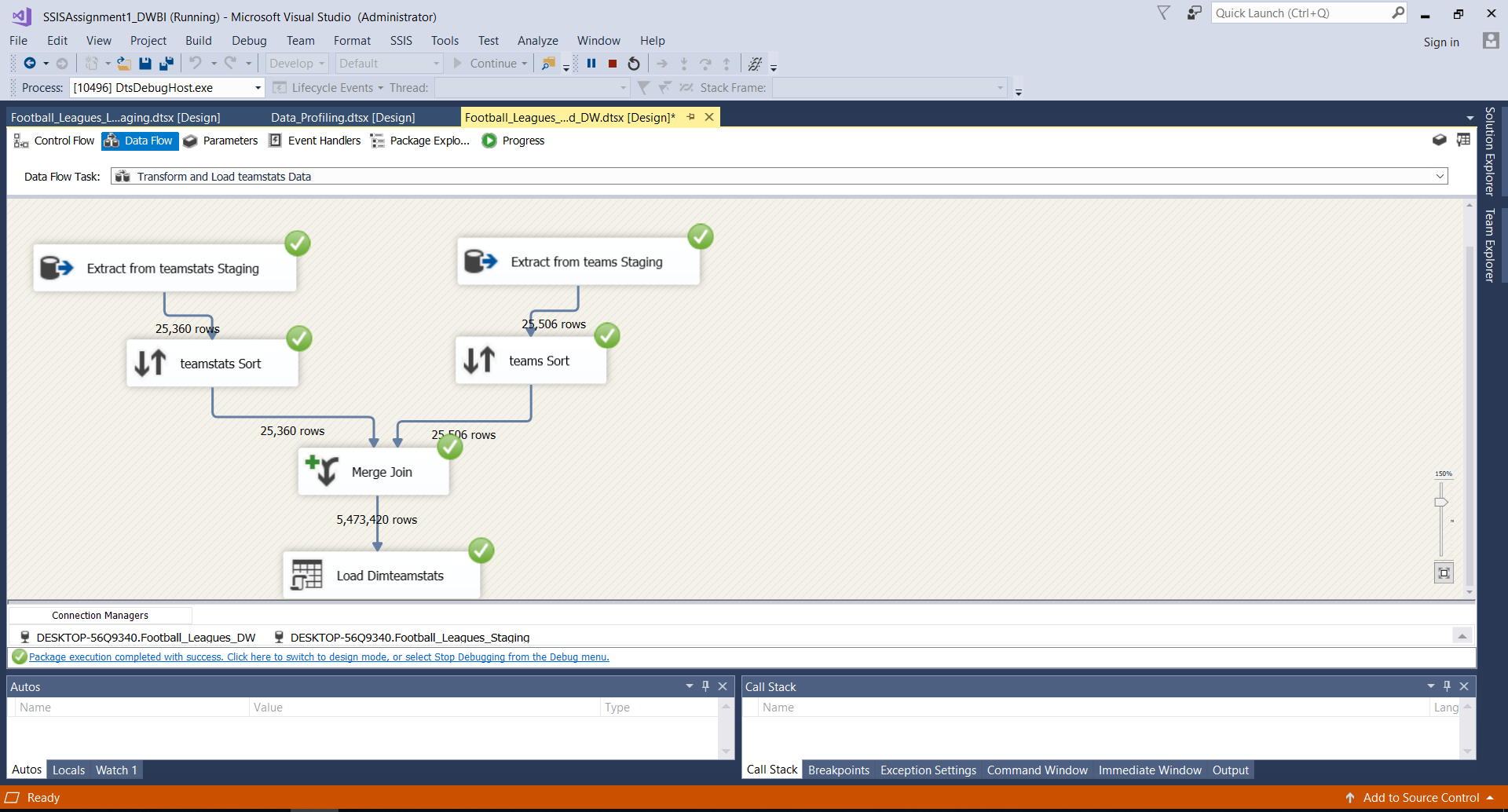
* **LEAGUES DATA TRANSFORM AND LOADING**

****

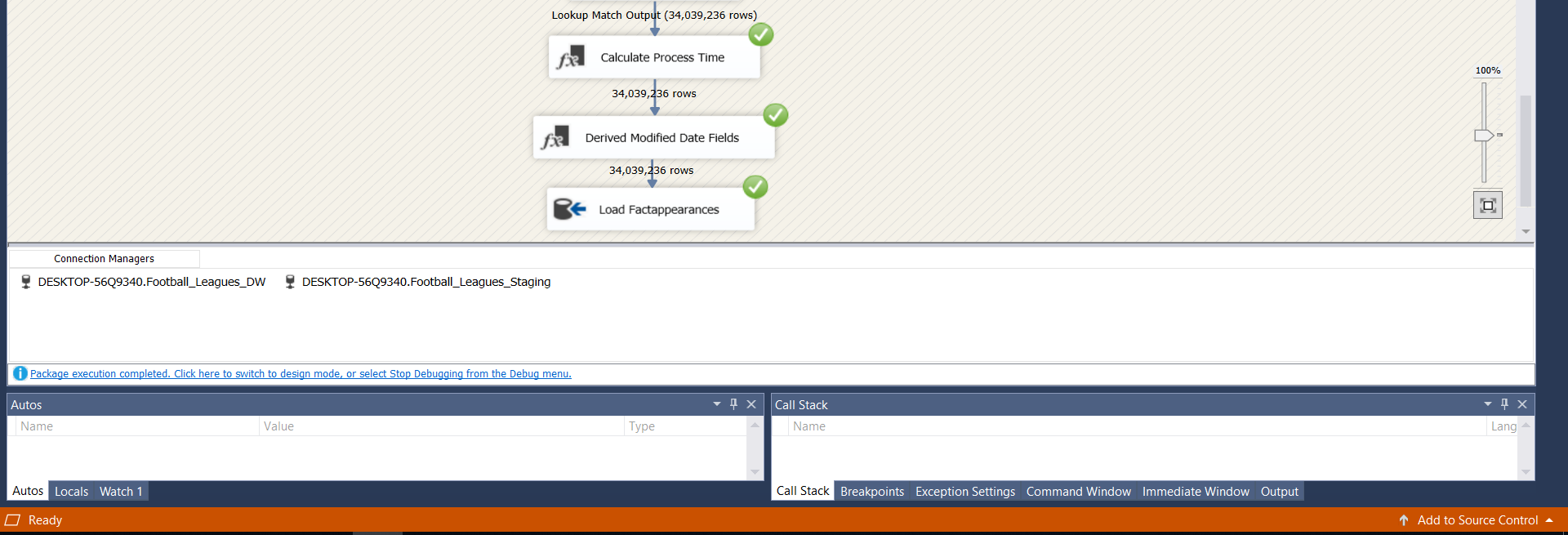
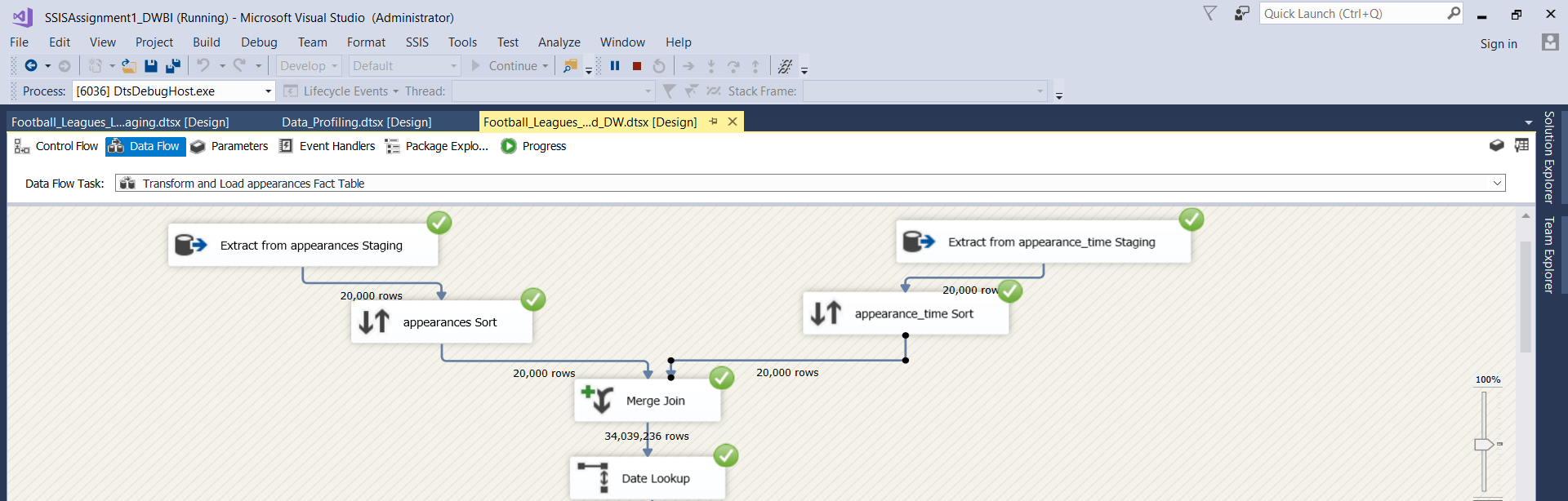
* **GAMES DATA TRANAFORM AND LOADING**
* **SHOTS DATA TRANSFORM AND LOADING**

****

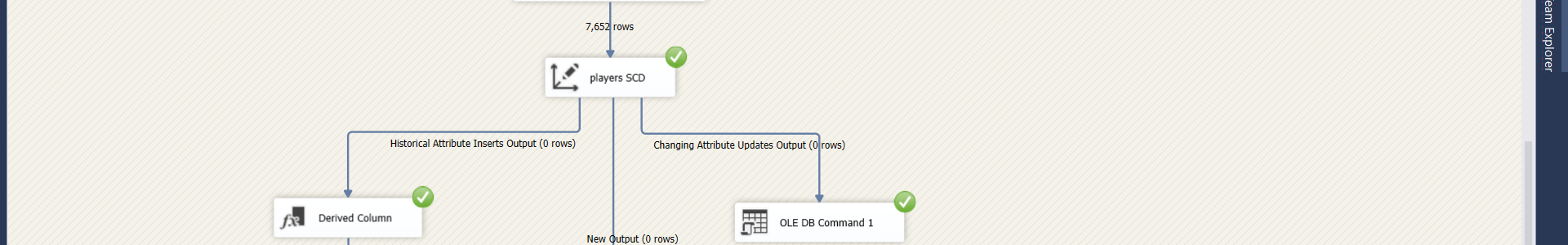
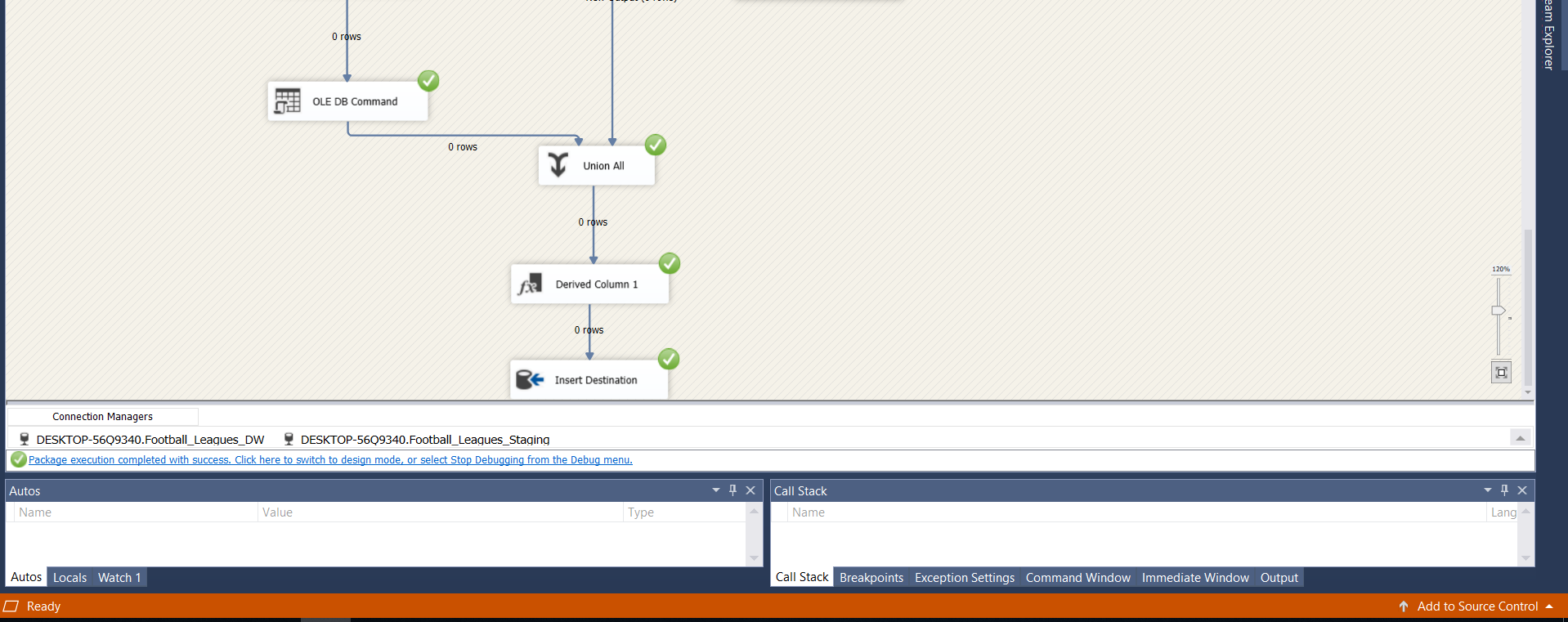
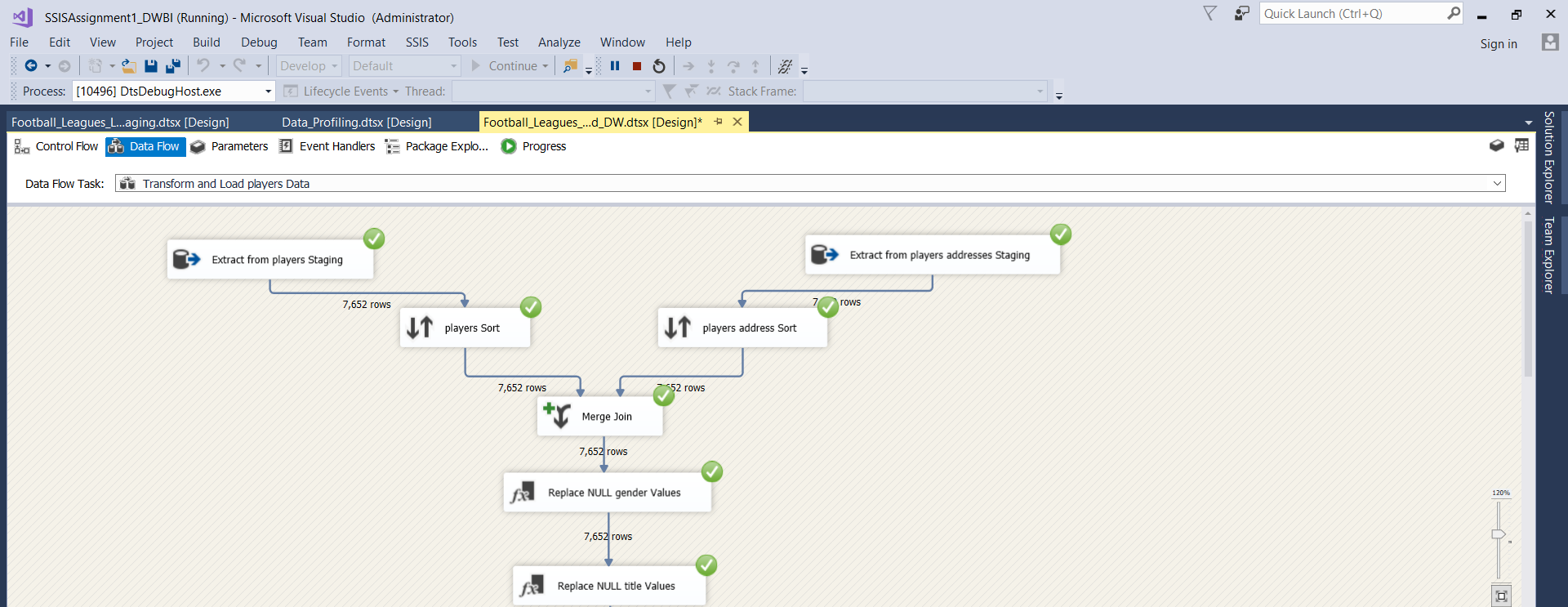
* **TEAMSTATS DATA TRANSFORM AND LOADING**

****

* **APPEARANCES DATA TRANSFORM AND LOADING**

****

* **PLAYERS DATA TRANSFORM AND LOADING**

****