DATA SCIENCE PROJECT TO PREDICT FOOTBALL WINS FOR 2003 TO 2019 FOR TENNESSEE STATE UNIVERSITY USING PYTHON AND DATA FROM THE SCHOOL WEBSITES

by Erika Harrell

In [1]:
```
#importing libraries for project
#python's lxml library parses xml and html files
from lxml import html
#python requests library gets data from web pages
import requests
#libraries for data wrangling & cleaning
import pandas as pd
import numpy as np
import datetime as dt
#libraries for data visualization
import matplotlib.pyplot as plt
import seaborn as sns
#libraries for machine learning
from sklearn.preprocessing import StandardScaler # data normalization
from sklearn.tree import DecisionTreeClassifier # Decision tree algorithm
from sklearn.model_selection import train_test_split # data split
from sklearn.metrics import accuracy_score # evaluation metric
from sklearn import tree
```

Web Scraping-getting data on location of games and scores from TSU websites

Data Wrangling-putting data from TSU websites into lists (one list per year)

In [2]:
```
#2019 data
#use requests.get() to get web page with 2019 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/EC18A8CE-A1B8-46E7-81B7-46F
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2019 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2019)
```

```
['Nov 23, 2019\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee State 37, Tennessee Tech 27\xa0\xa0', '\xa0', 'Nov 16, 2019\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'UT Martin 28, Tennessee State 17\xa0\xa0', '\xa0', 'Nov 09, 2019\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'EIU 49, Tennessee State 38\xa0\xa0', '\xa0', 'Nov 02, 2019\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'SEMO 32, Tennessee State 13\xa0\xa0', '\xa0', 'Oct 19, 2019\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 26, Austin Peay 24\xa0\xa0', '\xa0', 'Oct 12, 2019\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Murray State 19, Tennessee State 17\xa0\xa0', '\xa0', 'Oct 05, 2019\xa0\xa0', 'Jacksonville, Ala.  \xa0\xa0', 'Jacksonville State 31, Tennessee State 23\xa0\xa0', '\xa0', 'Sep 28, 2019\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Eastern Kentucky 42, Tennessee State 31\xa0\xa0', '\xa0', 'Sep 14, 2019\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'UAPB 37, Tennessee State 31\xa0\xa0', '\xa0', 'Sep 07, 2019\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Jackson State 49, Tennessee State 44\xa0\xa0', '\xa0', 'Sep 07, 2019\xa0\xa0', 'Murfreesboro, TN    \xa0\xa0', 'Middle Tennessee 45, Tennessee State 26\xa0\xa0', '\xa0', 'Aug 31, 2019\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 26, Mississippi Valley 20\xa0\xa0', '\xa0']
```

In [3]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2019=[w.replace("UAPB","Pine Bluff") for w in schedule2019]
schedule2019=[x.replace("SEMO","Southeast Missouri") for x in schedule2019]
schedule2019=[z.replace("EIU","Eastern Illinois") for z in schedule2019]
schedule2019=[e.replace("Tenn.","Tenn") for e in schedule2019]
schedule2019=[p.replace(".","") for p in schedule2019]
schedule2019=[q.replace("\xa0","") for q in schedule2019]
schedule2019=[v.replace(",","") for v in schedule2019]
schedule2019=[s.strip() for s in schedule2019]
print(schedule2019)
```

```
['Nov 23 2019', 'Cookeville Tenn', 'Tennessee State 37 Tennessee Tech 27', '', 'Nov 16 2019', 'Martin Tenn', 'UT Martin 28 Tennessee State 17', '', 'Nov 09 2019', 'Nashville Tenn', 'Eastern Illinois 49 Tennessee State 38', '', 'Nov 02 2019', 'Nashville Tenn', 'Southeast Missouri 32 Tennessee State 13', '', 'Oct 19 2019', 'Nashville Tenn', 'Tennessee State 26 Austin Peay 24', '', 'Oct 12 2019', 'Nashville Tenn', 'Murray State 19 Tennessee State 17', '', 'Oct 05 2019', 'Jacksonville Ala', 'Jacksonville State 31 Tennessee State 23', '', 'Sep 28 2019', 'Richmond Ky', 'Eastern Kentucky 42 Tennessee State 31', '', 'Sep 21 2019', 'Nashville Tenn', 'Pine Bluff 37 Tennessee State 31', '', 'Sep 14 2019', 'Memphis Tenn', 'Jackson State 49 Tennessee State 44', '', 'Sep 7 2019', 'Murfreesboro TN', 'Middle Tennessee 45 Tennessee State 26', '', 'Aug 31 2019', 'Nashville Tenn', 'Tennessee State 26 Mississippi Valley 20', '']
```

In [4]:
```
#2018 data
#use requests.get() to get web page with 2018 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/12580388-C113-4C2F-890C-445
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2018 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2018)
```

```
['Nov 17, 2018\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 31, UT Martin 28\xa0\xa0', '\xa0', 'Nov 10, 2018\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 41, Tennessee Tech 14\xa0\xa0', '\xa0', 'Nov 03-18    \xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'Southeast Missouri 38, Tennessee State 21\xa0\xa0', '\xa0', 'Oct 20, 2018\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 21\xa0\xa0', '\xa0', 'Oct 13, 2018\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray St. 45, Tennessee State 21\xa0\xa0', '\xa0', 'Oct 06, 2018\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Austin Peay 49, Tennessee State 34\xa0\xa0', '\xa0', 'Sep 29, 2018\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Vanderbilt 31, Tennessee State 27\xa0\xa0', '\xa0', 'Sep 22, 2018\xa0\xa0', 'Charleston, Ill.    \xa0\xa0', 'Tennessee State 41, Eastern Illinois 40\xa0\xa0', '\xa0', 'Sep 01, 2018\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 34, Bethune-Cookman 3\xa0\xa0', '\xa0']
```

In [5]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2018=[q.replace("Vanderbilt","Vanderbilt University") for q in schedule2018]
schedule2018=[r.replace("Cape Girardeau","CapeGirardeau") for r in schedule2018]
schedule2018=[t.replace("11-03-18","Nov 03 2018") for t in schedule2018]
schedule2018=[v.replace("TN","Tennessee Martin") for v in schedule2018]
schedule2018=[x.replace("SEMO","Southeast Missouri") for x in schedule2018]
schedule2018=[t.replace("Bethune-Cookman","Bethune Cookman") for t in schedule2018]
schedule2018=[a.replace(".","") for a in schedule2018]
schedule2018=[b.replace("\xa0","") for b in schedule2018]
schedule2018=[c.replace(",","") for c in schedule2018]
schedule2018=[t.strip() for t in schedule2018]
print(schedule2018)
```

```
['Nov 17 2018', 'Nashville Tenn', 'Tennessee State 31 Tennessee Martin 28', '', 'Nov 10 2018', 'Nashville Tenn', 'Tennessee State 41 Tennessee Tech 14', '', 'Nov 03 2018', 'CapeGirardeau Mo', 'Southeast Missouri 38 Tennessee State 21', '', 'Oct 20 2018', 'Nashville Tenn', 'Tennessee State 34', '', 'Oct 13 2018', 'Murray Ky', 'Murray St 45 Tennessee State 21', '', 'Oct 06 2018', 'Clarksville Tenn', 'Austin Peay 49 Tennessee State 34', '', 'Sep 29 2018', 'Nashville Tenn', 'Vanderbilt University 31 Tennessee State 27', '', 'Sep 22 2018', 'Charleston Ill', 'Tennessee State 41 Eastern Illinois 40', '', 'Sep 01 2018', 'Nashville Tenn', 'Tennessee State 34 Bethune Cookman 3', '']
```

In [6]:
```
#2017 data
#use requests.get() to get web page with 2017 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/1EEBBCA9-B018-43AB-B6EB-AFD
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2017 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2017)
```

```
['Nov 16, 2017\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 36, Tennessee State 6\xa0\xa0', '\xa0', 'Nov 11, 2017\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 23, SEMO 20\xa0\xa0', '\xa0', 'Nov 04, 2017\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 60, UVA 0\xa0\xa0', '\xa0', 'Oct 28, 2017\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee Tech 30, Tennessee State 26\xa0\xa0', '\xa0', 'Oct 14, 2017\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Austin Peay 21, Tennessee State 17\xa0\xa0', '\xa0', 'Oct 07, 2017\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 45, Eastern Kentucky 21\xa0\xa0', '\xa0', 'Sep 30, 2017\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Eastern Illinois 15, Tennessee State 16\xa0\xa0', '\xa0', 'Sep 23, 2017\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'UT Martin 31, Tennessee State 16\xa0\xa0', '\xa0', 'Sep 16, 2017\xa0\xa0', 'Tampa, Fla.    \xa0\xa0', 'Tennessee State 26, Florida A&M 13\xa0\xa0', '\xa0', 'Sep 09, 2017\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 17, Jackson State 15\xa0\xa0', '\xa0', 'Aug 31, 2017\xa0\xa0', 'Atlanta, Ga.    \xa0\xa0', 'Tennessee State 17, Georgia State 10\xa0\xa0', '\xa0']
```

In [7]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2017=[w.replace("UVA","Virginia Lynchburg") for w in schedule2017]
schedule2017=[x.replace("SEMO","Southeast Missouri") for x in schedule2017]
schedule2017=[y.replace("UT","Tennessee") for y in schedule2017]
schedule2017=[p.replace(".","") for p in schedule2017]
schedule2017=[q.replace("\xa0","") for q in schedule2017]
schedule2017=[v.replace(",","") for v in schedule2017]
schedule2017=[s.strip() for s in schedule2017]
print(schedule2017)
```

```
['Nov 16 2017', 'Jacksonville Ala', 'Jacksonville State 36 Tennessee State 6', '', 'Nov 11 2017', 'Nashville Tenn', 'Tennessee State 23 Southeast Missouri 20', '', 'Nov 04 2017', 'Nashville Tenn', 'Tennessee State 60 Virginia Lynchburg 0', '', 'Oct 28 2017', 'Cookeville Tenn', 'Tennessee Tech 30 Tennessee State 26', '', 'Oct 14 2017', 'Nashville Tenn', 'Austin Peay 21 Tennessee State 17', '', 'Oct 07 2017', 'Richmond Ky', 'Tennessee State 45 Eastern Kentucky 21', '', 'Sep 30 2017', 'Nashville Tenn', 'Tennessee State 16 Eastern Illinois 15', '', 'Sep 23 2017', 'Martin Tenn', 'Tennessee Martin 31 Tennessee State 16', '', 'Sep 16 2017', 'Tampa Fla', 'Tennessee State 26 Florida A&M 13', '', 'Sep 09 2017', 'Memphis Tenn', 'Tennessee State 17 Jackson State 15', '', 'Aug 31 2017', 'Atlanta Ga', 'Tennessee State 17 Georgia State 10', '']
```

In [8]:
```
#2016 data
#use requests.get() to get web page with 2016 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/F8A9B17C-4555-45FB-AAF3-D89
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2016 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2016)
```

```
['11-19-16    \xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'Tennessee State 32, Southeast Missouri 31\xa0\xa0', '\xa0', 'Nov 12, 2016\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 32, Southeast Missouri 31\xa0\xa0', '\xa0', 'Nov 05, 2016\xa0\xa0', 'Clarksville, Tenn.    \xa0\xa0', 'Tennessee State 41, Austin Peay 40\xa0\xa0', '\xa0', 'Oct 29, 2016\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray St. 38, Tennessee State 31\xa0\xa0', '\xa0', 'Oct 22, 2016\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Vanderbilt 35, Tennessee State 17\xa0\xa0', '\xa0', 'Oct 15, 2016\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 35, Tennessee Tech 34\xa0\xa0', '\xa0', 'Oct 08, 2016\xa0\xa0', 'Charleston, Ill.    \xa0\xa0', 'Eastern Illinois 35, Tennessee State 3\xa0\xa0', '\xa0', 'Oct 01, 2016\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 34, UT Martin 30\xa0\xa0', '\xa0', 'Sep 17, 2016\xa0\xa0', 'Daytona Beach, Fla.    \xa0\xa0', 'Tennessee State 31, Bethune-Cookman 24\xa0\xa0', '\xa0', 'Sep 10, 2016\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 40, Jackson State 20\xa0\xa0', '\xa0', 'Sep 03, 2016\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 44, Arkansas-Pine Bluff 0\xa0\xa0', '\xa0']
```

In [9]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2016=[q.replace("Pine Bluff","PineBluff") for q in schedule2016]
schedule2016=[r.replace("11-19-16","Nov 19 2016") for r in schedule2016]
schedule2016=[t.replace("Cape Girardeau","CapeGirardeau") for t in schedule2016]
schedule2016=[u.replace("Vanderbilt","Vanderbilt University") for u in schedule2016]
schedule2016=[v.replace("UT","Tennessee") for v in schedule2016]
schedule2016=[x.replace("Daytona Beach","DaytonaBeach") for x in schedule2016]
schedule2016=[a.replace("-","") for a in schedule2016]
schedule2016=[b.replace(".","") for b in schedule2016]
schedule2016=[c.replace("\xa0","") for c in schedule2016]
schedule2016=[d.replace(",","") for d in schedule2016]
schedule2016=[t.strip() for t in schedule2016]
print(schedule2016)
```

```
['Nov 19 2016', 'CapeGirardeau Mo', 'Tennessee State 32 Southeast Missouri 31', '', 'Nov 12 2016', 'Nashville TN', 'Tennessee State 32 Southeast Missouri 31', '', 'Nov 05 2016', 'Clarksville Tenn', 'Tennessee State 41 Austin Peay 40', '', 'Oct 29 2016', 'Murray Ky', 'Murray St 38 Tennessee State 31', '', 'Oct 22 2016', 'Nashville Tenn', 'Vanderbilt University 35 Tennessee State 17', '', 'Oct 15 2016', 'Nashville Tenn', 'Tennessee State 35 Tennessee Tech 34', '', 'Oct 08 2016', 'Charleston Ill', 'Eastern Illinois 35 Tennessee State 3', '', 'Oct 01 2016', 'Nashville Tenn', 'Tennessee State 34 Tennessee Martin 30', '', 'Sep 17 2016', 'DaytonaBeach Fla', 'Tennessee State 31 Bethune Cookman 24', '', 'Sep 10 2016', 'Memphis Tenn', 'Tennessee State 40 Jackson State 20', '', 'Sep 03 2016', 'Nashville Tenn', 'Tennessee State 44 Arkansas PineBluff 0', '']
```

In [10]:
```
#2015 data
#use requests.get() to get web page with 2015 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/B8713968-6888-4B84-B12E-9BA
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2015 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2015)
```

```
['Nov 21, 2015\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee Tech 30, Tennessee State 24\xa0\xa0', '\xa0', 'Nov 07, 2015\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray State 33, Tennessee State 24\xa0\xa0', '\xa0', 'Oct 31, 2015\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 20, Austin Peay 6\xa0\xa0', '\xa0', 'Oct 24, 2015\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Eastern Kentucky 31, Tennessee State 22\xa0\xa0', '\xa0', 'Oct 17, 2015\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Eastern Illinois 25, Tennessee State 22\xa0\xa0', '\xa0', 'Oct 10, 2015\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'UT Martin 28, Tennessee State 14\xa0\xa0', '\xa0', 'Sep 26, 2015\xa0\xa0', 'Tallahassee, Fla.  \xa0\xa0', 'Tennessee State 27, Florida A&M 14\xa0\xa0', '\xa0', 'Sep 19, 2015\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 48, Tennessee State 13\xa0\xa0', '\xa0', 'Sep 12, 2015\xa0\xa0', 'Memphis, TN    \xa0\xa0', 'Tennessee State 35, Jackson State 25\xa0\xa0', '\xa0', 'Sep 06, 2015\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 24, Alabama State 14\xa0\xa0', '\xa0']
```

In [11]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2015=[r.replace("Tigers","") for r in schedule2015]
schedule2015=[t.replace("UT","Tennessee") for t in schedule2015]
schedule2015=[u.replace("FAMU","Florida A&M") for u in schedule2015]
schedule2015=[a.replace("-","") for a in schedule2015]
schedule2015=[b.replace(".","") for b in schedule2015]
schedule2015=[c.replace("\xa0","") for c in schedule2015]
schedule2015=[d.replace(",","") for d in schedule2015]
schedule2015=[t.strip() for t in schedule2015]
print(schedule2015)
```

```
['Nov 21 2015', 'Cookeville Tenn', 'Tennessee Tech 30 Tennessee State 24', '', 'Nov 07 2015', 'Nashville Tenn', 'Murray State 33 Tennessee State 24', '', 'Oct 31 2015', 'Nashville Tenn', 'Tennessee State 20 Austin Peay 6', '', 'Oct 24 2015', 'Richmond Ky', 'Eastern Kentucky 31 Tennessee State 22', '', 'Oct 17 2015', 'Nashville Tenn', 'Eastern Illinois 25 Tennessee State 22', '', 'Oct 10 2015', 'Martin Tenn', 'Tennessee Martin 28 Tennessee State 14', '', 'Sep 26 2015', 'Tallahassee Fla', 'Tennessee State 27 Florida A&M 14', '', 'Sep 19 2015', 'Jacksonville Ala', 'Jacksonville State 48 Tennessee State 13', '', 'Sep 12 2015', 'Memphis TN', 'Tennessee State 35 Jackson State 25', '', 'Sep 06 2015', 'Nashville Tenn', 'Tennessee State 24 Alabama State 14', '']
```

In [12]:
```
#2014 data
#use requests.get() to get web page with 2014 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/26C45AE2-D036-475D-A4EF-3BE
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2014 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2014)
```

```
['Nov 22, 2014\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Tennessee State 48, Murray St. 33\xa0\xa0', '\xa0', 'Nov 08, 2014\xa0\xa0', 'Clarksville, Tenn.    \xa0\xa0', 'Tennessee State 31, Austin Peay 27\xa0\xa0', '\xa0', 'Nov 01, 2014\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Eastern Kentucky 56, Tennessee State 42\xa0\xa0', '\xa0', 'Oct 25, 2014\xa0\xa0', 'Charleston, Ill.    \xa0\xa0', 'Eastern Illinois 35, Tennessee State 17\xa0\xa0', '\xa0', 'Oct 18, 2014\xa0\xa0', 'Hale Stadium    \xa0\xa0', 'UT Martin 21, Tennessee State 16\xa0\xa0', '\xa0', 'Oct 11, 2014\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Jacksonville State 27, Tennessee State 20\xa0\xa0', '\xa0', '10-04-14    \xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'Southeast Missouri 28, Tennessee State 21\xa0\xa0', '\xa0', 'Sep 27, 2014\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 21, Tennessee Tech 7\xa0\xa0', '\xa0', 'Sep 20, 2014\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Florida A&M 35, Tennessee State 7\xa0\xa0', '\xa0', 'Sep 13, 2014\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 35, Jackson State 7\xa0\xa0', '\xa0', 'Sep 06, 2014\xa0\xa0', 'Montgomery, Ala.    \xa0\xa0', 'Alabama State 27, Tennessee State 2\xa0\xa0', '\xa0', 'Aug 30, 2014\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 58, Edward Waters 6\xa0\xa0', '\xa0']
```

In [13]:
```
#get rid of extra text and spaces in list
#put text for date and city into format that's compatible for data frame
#correct location for a game at TSU
schedule2014=[r.replace("Hale Stadium","Nashville, Tenn.") for r in schedule2014]
schedule2014=[s.replace("10-04-14","Oct 04, 2014") for s in schedule2014]
schedule2014=[t.replace("Cape Girardeau","CapeGirardeau") for t in schedule2014]
schedule2014=[a.replace("-","") for a in schedule2014]
schedule2014=[b.replace(".","") for b in schedule2014]
schedule2014=[c.replace("\xa0","") for c in schedule2014]
schedule2014=[d.replace(",","") for d in schedule2014]
schedule2014=[t.strip() for t in schedule2014]
print(schedule2014)
```

```
['Nov 22 2014', 'Murray Ky', 'Tennessee State 48 Murray St 33', '', 'Nov 08 2014', 'Clarksville Tenn', 'Tennessee State 31 Austin Peay 27', '', 'Nov 01 2014', 'Nashville Tenn', 'Eastern Kentucky 56 Tennessee State 42', '', 'Oct 25 2014', 'Charleston Ill', 'Eastern Illinois 35 Tennessee State 17', '', 'Oct 18 2014', 'Nashville Tenn', 'UT Martin 21 Tennessee State 16', '', 'Oct 11 2014', 'Nashville Tenn', 'Jacksonville State 27 Tennessee State 20', '', 'Oct 04 2014', 'CapeGirardeau Mo', 'Southeast Missouri 28 Tennessee State 21', '', 'Sep 27 2014', 'Nashville Tenn', 'Tennessee State 21 Tennessee Tech 7', '', 'Sep 20 2014', 'Nashville TN', 'Florida A&M 35 Tennessee State 7', '', 'Sep 13 2014', 'Memphis Tenn', 'Tennessee State 35 Jackson State 7', '', 'Sep 06 2014', 'Montgomery Ala', 'Alabama State 27 Tennessee State 21', '', 'Aug 30 2014', 'Nashville Tenn', 'Tennessee State 58 Edward Waters 6', '']
```

In [14]:
```
#2013 data
#use requests.get() to get web page with 2013 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/E3968BB1-3878-4C6A-9B21-E330
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2013 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2013)
```

```
['Dec 07, 2013\xa0\xa0', 'Charleston, Ill.    \xa0\xa0', 'Eastern Illinois 51, Tennessee State 10\xa0\xa0', '\xa0', 'Nov 30, 2013\xa0\xa0', 'Indianapolis, Ind.    \xa0\xa0', 'Tennessee State 31, Butler 0\xa0\xa0', '\xa0', 'Nov 09, 2013\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 31, Austin Peay 6\xa0\xa0', '\xa0', 'Nov 02, 2013\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 17, Murray State 10\xa0\xa0', '\xa0', 'Oct 26, 2013\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Eastern Kentucky 34, Tennessee State 7\xa0\xa0', '\xa0', 'Oct 19, 2013\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'Tennessee State 24, Tennessee State 14\xa0\xa0', '\xa0', 'Oct 12, 2013\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Tennessee State 31, Jacksonville State 15\xa0\xa0', '\xa0', 'Oct 05, 2013\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 40, Southeast Missouri 16\xa0\xa0', '\xa0', 'Sep 28, 2013\xa0\xa0', 'St. Louis, Mo.    \xa0\xa0', 'Tennessee State 73, Central State 6\xa0\xa0', '\xa0', 'Sep 21, 2013\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee State 41, Tennessee Tech 21\xa0\xa0', '\xa0', 'Sep 14, 2013\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 26, Jackson State 7\xa0\xa0', '\xa0', 'Sep 07, 2013\xa0\xa0', 'Tallahassee, Fla.    \xa0\xa0', 'Tennessee State 27, Florida A&M 7\xa0\xa0', '\xa0', 'Sep 01, 2013\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Bethune-Cookman 12, Tennessee State 9\xa0\xa0', '\xa0']
```

In [15]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2013=[r.replace("St. Louis","StLouis") for r in schedule2013]
schedule2013=[s.replace("Butler","Butler University") for s in schedule2013]
schedule2013=[t.replace("UT","Tennessee") for t in schedule2013]
schedule2013=[a.replace("-","") for a in schedule2013]
schedule2013=[b.replace(".","") for b in schedule2013]
schedule2013=[c.replace("\xa0","") for c in schedule2013]
schedule2013=[d.replace(",","") for d in schedule2013]
schedule2013=[t.strip() for t in schedule2013]
print(schedule2013)
```

```
['Dec 07 2013', 'Charleston Ill', 'Eastern Illinois 51 Tennessee State 10', '', 'Nov 30 2013', 'Indianapolis Ind', 'Tennessee State 31 Butler University 0', '', 'Nov 09 2013', 'Nashville Tenn', 'Tennessee State 31 Austin Peay 6', '', 'Nov 02 2013', 'Nashville Tenn', 'Tennessee State 17 Murray State 10', '', 'Oct 26 2013', 'Richmond Ky', 'Eastern Kentucky 34 Tennessee State 7', '', 'Oct 19 2013', 'Martin Tenn', 'Tennessee State 24 Tennessee Martin 14', '', 'Oct 12 2013', 'Jacksonville Ala', 'Tennessee State 31 Jacksonville State 15', '', 'Oct 05 2013', 'Nashville Tenn', 'Tennessee State 40 Southeast Missouri 16', '', 'Sep 28 2013', 'StLouis Mo', 'Tennessee State 73 Central State 6', '', 'Sep 21 2013', 'Cookeville Tenn', 'Tennessee State 41 Tennessee Tech 21', '', 'Sep 14 2013', 'Memphis Tenn', 'Tennessee State 26 Jackson State 7', '', 'Sep 07 2013', 'Tallahassee Fla', 'Tennessee State 27 Florida A&M 7', '', 'Sep 01 2013', 'Nashville Tenn', 'Bethune Cookman 12 Tennessee State 9', '']
```

In [16]:
```
#2012 data
#use requests.get() to get web page with 2012 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/BDF8C4F3-07B8-4D00-847F-9227
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2012 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2012)
```

```
['Nov 17, 2012\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'UT Martin 35, Tennessee State 26\xa0\xa0', '\xa0', 'Nov 03, 2012\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray State 31, Tennessee State 3\xa0\xa0', '\xa0', 'Oct 27, 2012\xa0\xa0', 'Nashville, Tennessee\xa0\xa0', 'Tennessee State 22, Tennessee Tech 22\xa0\xa0', '\xa0', 'Oct 20, 2012\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 31, Tennessee State 3\xa0\xa0', '\xa0', 'Oct 13, 2012\xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'Tennessee State 40, Southeast Missouri 28\xa0\xa0', '\xa0', 'Oct 06, 2012\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 43, Arkansas Pine Bluff 34\xa0\xa0', '\xa0', 'Sep 29, 2012\xa0\xa0', 'Daytona Beach, Fla.    \xa0\xa0', 'Tennessee State 34, Bethune-Cookman 14\xa0\xa0', '\xa0', 'Sep 22, 2012\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 26, Jackson State 24\xa0\xa0', '\xa0', 'Sep 15, 2012\xa0\xa0', 'Clarksville, Tenn.    \xa0\xa0', 'Tennessee State 34, Austin Peay 21\xa0\xa0', '\xa0', 'Sep 08, 2012\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 17, Florida A&M 14\xa0\xa0', '\xa0', 'Sep 01, 2012\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 33, Alabama A&M 14\xa0\xa0', '\xa0']
```

In [17]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2012=[q.replace("Daytona Beach","DaytonaBeach") for q in schedule2012]
schedule2012=[r.replace("Bulldog Stadium","Butler University") for r in schedule2012]
schedule2012=[s.replace("UT","Tennessee") for s in schedule2012]
schedule2012=[t.replace("Pine Bluff","PineBluff") for t in schedule2012]
schedule2012=[a.replace("-","") for a in schedule2012]
schedule2012=[b.replace(".","") for b in schedule2012]
schedule2012=[c.replace("\xa0","") for c in schedule2012]
schedule2012=[d.replace(",","") for d in schedule2012]
schedule2012=[t.strip() for t in schedule2012]
print(schedule2012)
```

```
['Nov 17 2012', 'Martin Tenn', 'Tennessee Martin 35 Tennessee State 26', '', 'Nov 03 2012', 'Murray Ky', 'Murray State 31 Tennessee State 3', '', 'Oct 27 2012', 'Nashville Tennessee', 'Tennessee State 22 Tennessee Tech 22', '', 'Oct 20 2012', 'Jacksonville Ala', 'Jacksonville State 31 Tennessee State 3', '', 'Oct 13 2012', 'CapeGirardeau Mo', 'Tennessee State 40 Southeast Missouri 28', '', 'Oct 06 2012', 'Nashville Tenn', 'Tennessee State 43 Arkansas PineBluff 34', '', 'Sep 29 2012', 'DaytonaBeach Fla', 'Tennessee State 34 Bethune Cookman 14', '', 'Sep 22 2012', 'Nashville Tenn', 'Tennessee State 26 Jackson State 24', '', 'Sep 15 2012', 'Clarksville Tenn', 'Tennessee State 34 Austin Peay 21', '', 'Sep 08 2012', 'Memphis Tenn', 'Tennessee State 17 Florida A&M 14', '', 'Sep 01 2012', 'Nashville Tenn', 'Tennessee State 33 Alabama A&M 14', '']
```

In [18]:
```
#2011 data
#use requests.get() to get web page with 2011 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/CAE35442-6529-4337-854B-C06
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2011 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2011)
```

```
['Nov 19, 2011\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Jacksonville State 38, Tennessee State 16\xa0\xa0', '\xa0', 'Nov 12, 2011\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 35, UT Martin 30\xa0\xa0', '\xa0', 'Nov 05, 2011\xa0\xa0', 'Charleston, Ill.    \xa0\xa0', 'Tennessee State 28, Eastern Illinois 17\xa0\xa0', '\xa0', 'Oct 22, 2011\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Eastern Kentucky 31, Tennessee State 22\xa0\xa0', '\xa0', 'Oct 15, 2011\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 55, Southeast Missouri 31\xa0\xa0', '\xa0', 'Oct 08, 2011\xa0\xa0', 'USAFA, Colo.    \xa0\xa0', 'Air Force 63, Tennessee State 24\xa0\xa0', '\xa0', 'Oct 01, 2011\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 42, Tennessee Tech 34\xa0\xa0', '\xa0', 'Sep 24, 2011\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray State 34, Tennessee State 29\xa0\xa0', '\xa0', 'Sep 17, 2011\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Jackson State 35, Tennessee State 29\xa0\xa0', '\xa0', 'Sep 10, 2011\xa0\xa0', 'Memphis, TN    \xa0\xa0', 'Tennessee State 33, Southern U. 7\xa0\xa0', '\xa0']
```

In [19]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2011=[q.replace("USAFA","AirForceAcademy") for q in schedule2011]
schedule2011=[r.replace("Southern U.","Southern University") for r in schedule2011]
schedule2011=[s.replace("UT","Tennessee") for s in schedule2011]
schedule2011=[a.replace("-","") for a in schedule2011]
schedule2011=[b.replace(".","") for b in schedule2011]
schedule2011=[c.replace("\xa0","") for c in schedule2011]
schedule2011=[d.replace(",","") for d in schedule2011]
schedule2011=[t.strip() for t in schedule2011]
print(schedule2011)
```

```
['Nov 19 2011', 'Nashville TN', 'Jacksonville State 38 Tennessee State 16', '', 'Nov 12 2011', 'Nashville Tenn', 'Tennessee State 35 Tennessee Martin 30', '', 'Nov 05 2011', 'Charleston Ill', 'Tennessee State 28 Eastern Illinois 17', '', 'Oct 22 2011', 'Richmond Ky', 'Eastern Kentucky 31 Tennessee State 22', '', 'Oct 15 2011', 'Nashville Tenn', 'Tennessee State 55 Southeast Missouri 31', '', 'Oct 08 2011', 'AirForceAcademy Colo', 'Air Force 63 Tennessee State 24', '', 'Oct 01 2011', 'Nashville Tenn', 'Tennessee State 42 Tennessee Tech 34', '', 'Sep 24 2011', 'Murray Ky', 'Murray State 34 Tennessee State 29', '', 'Sep 17 2011', 'Nashville Tenn', 'Jackson State 35 Tennessee State 29', '', 'Sep 10 2011', 'Memphis Tenn', 'Tennessee State 33 Southern University 7', '']
```

In [20]:
```
#2010 data
#use requests.get() to get web page with 2010 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/1087038-E3AB-4C3D-95BB-76C
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2010 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2010)
```

```
['Nov 20, 2010\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray State 28, Tennessee State 23\xa0\xa0', '\xa0', 'Nov 13, 2010\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'UT Martin 31, Tennessee State 0\xa0\xa0', '\xa0', 'Nov 06, 2010\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 28, Tennessee Tech 21\xa0\xa0', '\xa0', 'Oct 30, 2010\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Eastern Illinois 21, Tennessee State 11\xa0\xa0', '\xa0', 'Oct 23, 2010\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 28, Tennessee State 14\xa0\xa0', '\xa0', 'Oct 16, 2010\xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'Southeast Missouri 19, Tennessee State 14\xa0\xa0', '\xa0', 'Oct 09, 2010\xa0\xa0', 'Indianapolis, Ind.    \xa0\xa0', 'Tennessee State 21, North Carolina A&T 10\xa0\xa0', '\xa0', 'Oct 02, 2010\xa0\xa0', 'Atlanta, GA    \xa0\xa0', 'Florida A&M 29, Tennessee State 13\xa0\xa0', '\xa0', 'Sep 18, 2010\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Austin Peay 24, Tennessee State 23\xa0\xa0', '\xa0', 'Sep 11, 2010\xa0\xa0', 'Memphis, TN    \xa0\xa0', 'Tennessee State 27, Jackson State 26\xa0\xa0', '\xa0', 'Sep 04, 2010\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 27, Alabama A&M 16\xa0\xa0', '\xa0']
```

In [21]:
```
#get rid of extra text and spaces in list
#put text for date, school names, and city into format that's compatible for data frame
schedule2010=[q.replace("Oct. 2, 2010","Oct 02, 2010") for q in schedule2010]
schedule2010=[r.replace("Oct. 9, 2010","Oct. 09, 2010") for r in schedule2010]
schedule2010=[s.replace("N.C.","North Carolina") for s in schedule2010]
schedule2010=[t.replace("North Carolina ","NorthCarolina") for t in schedule2010]
schedule2010=[u.replace("UT","Tennessee") for u in schedule2010]
schedule2010=[v.replace("Cape Girardeau","CapeGirardeau") for v in schedule2010]
schedule2010=[a.replace("-","") for a in schedule2010]
schedule2010=[b.replace(".","") for b in schedule2010]
schedule2010=[c.replace("\xa0","") for c in schedule2010]
schedule2010=[d.replace(",","") for d in schedule2010]
schedule2010=[t.strip() for t in schedule2010]
print(schedule2010)
```

```
['Nov 20 2010', 'Murray Ky', 'Murray State 28 Tennessee State 23', '', 'Nov 13 2010', 'Martin Tenn', 'Tennessee Martin 31 Tennessee State 0', '', 'Nov 06 2010', 'Nashville Tenn', 'Tennessee State 28 Tennessee Tech 21', '', 'Oct 30 2010', 'Nashville Tenn', 'Eastern Illinois 21 Tennessee State 11', '', 'Oct 23 2010', 'Jacksonville Ala', 'Jacksonville State 28 Tennessee State 14', '', 'Oct 16 2010', 'CapeGirardeau Mo', 'Southeast Missouri 19 Tennessee State 14', '', 'Oct 09 2010', 'Indianapolis Ind', 'Tennessee State 21 NorthCarolina A&T 10', '', 'Oct 02 2010', 'Atlanta GA', 'Florida A&M 29 Tennessee State 13', '', 'Sep 18 2010', 'Nashville TN', 'Austin Peay 24 Tennessee State 23', '', 'Sep 11 2010', 'Memphis TN', 'Tennessee State 27 Jackson State 26', '', 'Sep 04 2010', 'Nashville TN', 'Tennessee State 27 Alabama A&M 16', '']
```

In [22]:
```
#2009 data
#use requests.get() to get web page with 2009 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/D83B06F7-031A-4E37-94F7-ACE
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2009 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2009)
```

```
['Nov 19, 2009\xa0\xa0', 'Charleston, Ill.    \xa0\xa0', 'Tennessee State 21, Eastern Illinois 10\xa0\xa0', '\xa0', 'Nov 14, 2009\xa0\xa0', 'Clarksville, Tenn.    \xa0\xa0', 'Tennessee State 24, Austin Peay 24\xa0\xa0', '\xa0', 'Nov 07, 2009\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 26, Tennessee Tech 21\xa0\xa0', '\xa0', 'Oct 31, 2009\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee State 26, Tennessee Tech 21\xa0\xa0', '\xa0', 'Oct 17, 2009\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 34, Tennessee Tech 7\xa0\xa0', '\xa0', 'Oct 10, 2009\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Tennessee State 21, Eastern Kentucky 21\xa0\xa0', '\xa0', 'Oct 03, 2009\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 23, Southeast Missouri 16\xa0\xa0', '\xa0', 'Sep 26, 2009\xa0\xa0', 'Baton Rouge, LA    \xa0\xa0', 'Southern University 23, Tennessee State 14\xa0\xa0', '\xa0', 'Sep 12, 2009\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Jackson State 24, Tennessee State 14\xa0\xa0', '\xa0', 'Sep 05, 2009\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Alabama A&M 28, Tennessee State 26\xa0\xa0', '\xa0']
```

In [23]:
```
#get rid of extra text and spaces in list
#put text for date and city into format that's compatible for data frame
schedule2009=[q.replace("Tigers","") for q in schedule2009]
schedule2009=[r.replace("Baton Rouge","BatonRouge") for r in schedule2009]
schedule2009=[a.replace("-","") for a in schedule2009]
schedule2009=[b.replace(".","") for b in schedule2009]
schedule2009=[c.replace("\xa0","") for c in schedule2009]
schedule2009=[d.replace(",","") for d in schedule2009]
schedule2009=[t.strip() for t in schedule2009]
print(schedule2009)
```

```
['Nov 19 2009', 'Charleston Ill', 'Tennessee State 21 Eastern Illinois 10', '', 'Nov 14 2009', 'Clarksville Tenn', 'Tennessee State 24 Austin Peay 24', '', 'Nov 07 2009', 'Nashville Tenn', 'Tennessee State 26 Tennessee Tech 21', '', 'Oct 31 2009', 'Cookeville Tenn', 'Tennessee State 26 Tennessee Tech 21', '', 'Oct 17 2009', 'Nashville TN', 'Tennessee State 34 Tennessee Tech 7', '', 'Oct 10 2009', 'Richmond Ky', 'Tennessee State 21 Eastern Kentucky 21', '', 'Oct 03 2009', 'Nashville Tenn', 'Tennessee State 23 Southeast Missouri 16', '', 'Sep 26 2009', 'BatonRouge LA', 'Southern University 23 Tennessee State 14', '', 'Sep 12 2009', 'Memphis TN', 'Jackson State 24 Tennessee State 14', '', 'Sep 05 2009', 'Nashville TN', 'Alabama A&M 28 Tennessee State 26', '']
```

In [24]:
```
#2008 data
#use requests.get() to get web page with 2008 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/E0564AE9-37A0-4B65-8811-B7B
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2008 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2008)
```

```
['Nov 22, 2008\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Murray State 24, Tennessee State 17\xa0\xa0', '\xa0', 'Nov 15, 2008\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 24, Tennessee State 21\xa0\xa0', '\xa0', 'Nov 08, 2008\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 45, Eastern Illinois 24\xa0\xa0', '\xa0', 'Nov 01, 2008\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 41, Tennessee Tech 14\xa0\xa0', '\xa0', 'Oct 25, 2008\xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'Southeast Missouri 27, Tennessee State 20\xa0\xa0', '\xa0', 'Oct 18, 2008\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 37, Austin Peay 3\xa0\xa0', '\xa0', 'Oct 11, 2008\xa0\xa0', 'Martin, TN    \xa0\xa0', 'Tennessee State 30, UT Martin 27\xa0\xa0', '\xa0', 'Oct 04, 2008\xa0\xa0', 'Atlanta, GA    \xa0\xa0', 'Florida A&M 29, Tennessee State 21\xa0\xa0', '\xa0', 'Sep 20, 2008\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 34, Eastern Kentucky 2\xa0\xa0', '\xa0', 'Sep 13, 2008\xa0\xa0', 'Memphis, TN    \xa0\xa0', 'Tennessee State 41, Jackson State 18\xa0\xa0', '\xa0', 'Sep 06, 2008\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 34, Southern U 32\xa0\xa0', '\xa0', 'Aug 30, 2008\xa0\xa0', 'Huntsville, AL    \xa0\xa0', 'Samford 24, Tennessee State 34, Alabama A&M 13\xa0\xa0', '\xa0']
```

In [25]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2008=[r.replace("Southern U","Southern University") for r in schedule2008]
schedule2008=[s.replace("UT","Tennessee") for s in schedule2008]
schedule2008=[t.replace("Cape Girardeau","CapeGirardeau") for t in schedule2008]
schedule2008=[a.replace("-","") for a in schedule2008]
schedule2008=[b.replace(".","") for b in schedule2008]
schedule2008=[c.replace("\xa0","") for c in schedule2008]
schedule2008=[d.replace(",","") for d in schedule2008]
schedule2008=[t.strip() for t in schedule2008]
print(schedule2008)
```

```
['Nov 22 2008', 'Murray Ky', 'Murray State 24 Tennessee State 17', '', 'Nov 15 2008', 'Jacksonville Ala', 'Jacksonville State 24 Tennessee State 21', '', 'Nov 08 2008', 'Nashville TN', 'Tennessee State 45 Eastern Illinois 24', '', 'Nov 01 2008', 'Nashville TN', 'Tennessee State 41 Tennessee Tech 14', '', 'Oct 25 2008', 'CapeGirardeau Mo', 'Southeast Missouri 27 Tennessee State 20', '', 'Oct 18 2008', 'Nashville TN', 'Tennessee State 37 Austin Peay 3', '', 'Oct 11 2008', 'Martin TN', 'Tennessee State 30 Tennessee Martin 27', '', 'Oct 04 2008', 'Atlanta GA', 'Florida A&M 29 Tennessee State 21', '', 'Sep 20 2008', 'Nashville TN', 'Tennessee State 34 Eastern Kentucky 2', '', 'Sep 13 2008', 'Memphis TN', 'Tennessee State 41 Jackson State 18', '', 'Sep 06 2008', 'Nashville TN', 'Tennessee State 34 Southern University 32', '', 'Aug 30 2008', 'Huntsville AL', 'Tennessee State 34 Alabama A&M 13', '']
```

In [26]:
```
#2007 data
#use requests.get() to get web page with 2007 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/4C480A5B-76BE-4378-8E95-9BB
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2007 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2007)
```

```
['Nov 17, 2007\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee-Martin 43, Tennessee State 38\xa0\xa0', '\xa0', 'Nov 08, 2007\xa0\xa0', 'Birmingham, AL    \xa0\xa0', 'Tennessee State 36, Samford 28\xa0\xa0', '\xa0', 'Nov 03, 2007\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 42, Murray State 28\xa0\xa0', '\xa0', 'Oct 27, 2007\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Eastern Illinois 38, Tennessee State 35\xa0\xa0', '\xa0', 'Oct 20, 2007\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Eastern Kentucky 45, Tennessee State 7\xa0\xa0', '\xa0', 'Oct 13, 2007\xa0\xa0', 'Cookeville, TN    \xa0\xa0', 'Tennessee State 45, Tennessee Tech 26\xa0\xa0', '\xa0', 'Sep 29, 2007\xa0\xa0', 'Atlanta, Ga    \xa0\xa0', 'Florida A&M 18, Tennessee State 17\xa0\xa0', '\xa0', 'Sep 22, 2007\xa0\xa0', 'Baton Rouge, LA    \xa0\xa0', 'Southern U 41, Tennessee State 34\xa0\xa0', '\xa0', 'Sep 15, 2007\xa0\xa0', 'Clarksville, Tenn.    \xa0\xa0', 'Tennessee State 33, Austin Peay 16\xa0\xa0', '\xa0', 'Sep 08, 2007\xa0\xa0', 'Memphis, TN    \xa0\xa0', 'Tennessee State 16, Jackson State 13\xa0\xa0', '\xa0', 'Sep 01, 2007\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Alabama A&M 49, Tennessee State 23\xa0\xa0', '\xa0']
```

In [27]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2007=[r.replace("Southern","Southern University") for r in schedule2007]
schedule2007=[s.replace("Samford","Samford University") for s in schedule2007]
schedule2007=[t.replace("Baton Rouge","BatonRouge") for t in schedule2007]
schedule2007=[a.replace("-"," ") for a in schedule2007]
schedule2007=[b.replace(".","") for b in schedule2007]
schedule2007=[c.replace("\xa0","") for c in schedule2007]
schedule2007=[d.replace(",","") for d in schedule2007]
schedule2007=[t.strip() for t in schedule2007]
print(schedule2007)
```

```
['Nov 17 2007', 'Nashville TN', 'Tennessee Martin 43 Tennessee State 38', '', 'Nov 08 2007', 'Birmingham AL', 'Tennessee State 36 Samford University 28', '', 'Nov 03 2007', 'Nashville TN', 'Tennessee State 42 Murray State 28', '', 'Oct 27 2007', 'Nashville TN', 'Eastern Illinois 38 Tennessee State 35', '', 'Oct 20 2007', 'Richmond Ky', 'Eastern Kentucky 45 Tennessee State 7', '', 'Oct 13 2007', 'Cookeville TN', 'Tennessee State 45 Tennessee Tech 26', '', 'Sep 29 2007', 'Atlanta Ga', 'Florida A&M 18 Tennessee State 17', '', 'Sep 22 2007', 'BatonRouge LA', 'Southern University 41 Tennessee State 34', '', 'Sep 15 2007', 'Clarksville Tenn', 'Tennessee State 33 Austin Peay 16', '', 'Sep 08 2007', 'Memphis TN', 'Tennessee State 16 Jackson State 13', '', 'Sep 01 2007', 'Nashville TN', 'Alabama A&M 49 Tennessee State 23', '']
```

In [28]:
```
#2006 data
#use requests.get() to get web page with 2006 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/902C2D72-E881-4836-87E2-2BA
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2006 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2006)
```

```
['Nov 18, 2006\xa0\xa0', 'Richmond, Ky.    \xa0\xa0', 'Eastern Kentucky 20, Tennessee State 3\xa0\xa0', '\xa0', 'Nov 11, 2006\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 31, Southeast Missouri 0\xa0\xa0', '\xa0', 'Nov 04, 2006\xa0\xa0', 'Charleston, IL    \xa0\xa0', 'EASTERN ILLINOIS 29, Tennessee State 3\xa0\xa0', '\xa0', 'Oct 28, 2006\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 29, Samford 7\xa0\xa0', '\xa0', 'Oct 21, 2006\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Tennessee State 38, Jacksonville State 17\xa0\xa0', '\xa0', 'Oct 14, 2006\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 30, Tennessee Tech 20\xa0\xa0', '\xa0', 'Oct 07, 2006\xa0\xa0', 'Atlanta, GA    \xa0\xa0', 'Florida A&M 28, Tennessee State 17\xa0\xa0', '\xa0', 'Sep 23, 2006\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Vanderbilt 38, Tennessee State 9\xa0\xa0', '\xa0', 'Sep 16, 2006\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 31, Jackson State 28\xa0\xa0', '\xa0', 'Sep 09, 2006\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 25, Murray State 20\xa0\xa0', '\xa0', 'Sep 02, 2006\xa0\xa0', 'Nashville,TN    \xa0\xa0', 'Alabama A&M 27, Tennessee State 20\xa0\xa0', '\xa0']
```

In [29]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2006=[q.replace("Nashville,TN","Nashville, TN") for q in schedule2006]
schedule2006=[q.replace("Murray State","Murray State") for q in schedule2006]
schedule2006=[r.replace("Vanderbilt","Vanderbilt University") for r in schedule2006]
schedule2006=[s.replace("Samford","Samford University") for s in schedule2006]
schedule2006=[t.replace("EASTERN ILLINOIS","Eastern Illinois") for t in schedule2006]
schedule2006=[a.replace("-","") for a in schedule2006]
schedule2006=[b.replace(".","") for b in schedule2006]
schedule2006=[c.replace("\xa0","") for c in schedule2006]
schedule2006=[d.replace(",","") for d in schedule2006]
schedule2006=[t.strip() for t in schedule2006]
print(schedule2006)
```

```
['Nov 18 2006', 'Richmond Ky', 'Eastern Kentucky 20 Tennessee State 3', '', 'Nov 11 2006', 'Nashville TN', 'Tennessee State 31 Southeast Missouri 0', '', 'Nov 04 2006', 'Charleston IL', 'Eastern Illinois 29 Tennessee State 3', '', 'Oct 28 2006', 'Nashville TN', 'Tennessee State 29 Samford University 7', '', 'Oct 21 2006', 'Jacksonville Ala', 'Tennessee State 38 Jacksonville State 17', '', 'Oct 14 2006', 'Nashville TN', 'Tennessee State 30 Tennessee Tech 20', '', 'Oct 07 2006', 'Atlanta GA', 'Florida A&M 28 Tennessee State 17', '', 'Sep 23 2006', 'Nashville TN', 'Vanderbilt University 38 Tennessee State 9', '', 'Sep 16 2006', 'Nashville TN', 'Tennessee State 31 Jackson State 28', '', 'Sep 09 2006', 'Nashville TN', 'Tennessee State 25 Murray State 20', '', 'Sep 02 2006', 'Nashville TN', 'Alabama A&M 27 Tennessee State 20', '']
```

In [30]:
```
#2005 data
#use requests.get() to get web page with 2005 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/99B72BE2-12D4-4887-A990-2490
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2005 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2005)
```

```
['Nov 19, 2005\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'EKU 49, Tennessee State 0\xa0\xa0', '\xa0', 'Nov 12, 2005\xa0\xa0', 'Cape Girardeau, Mo. \xa0\xa0', 'SEMO 32, Tennessee State 24\xa0\xa0', '\xa0', 'Nov 05, 2005\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Eastern Illinois 27, Tennessee State 3\xa0\xa0', '\xa0', 'Oct 29, 2005\xa0\xa0', 'Birmingham, Ala.    \xa0\xa0', 'Samford 38, Tennessee State 0\xa0\xa0', '\xa0', 'Oct 22, 2005\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 33, Tennessee State 3\xa0\xa0', '\xa0', 'Oct 13, 2005\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee State 31, Tennessee Tech 28\xa0\xa0', '\xa0', 'Oct 01, 2005\xa0\xa0', 'Indianapolis, Ind.    \xa0\xa0', 'No. Carolina A&T St. 16, Tennessee State 3\xa0\xa0', '\xa0', 'Sep 24, 2005\xa0\xa0', 'Atlanta, GA    \xa0\xa0', 'Florida A&M 26, Tennessee State 23\xa0\xa0', '\xa0', 'Sep 17, 2005\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'Tennessee-Martin 42, Tennessee State 20\xa0\xa0', '\xa0', 'Sep 10, 2005\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 26, Jackson State 14\xa0\xa0', '\xa0', 'Sep 03, 2005\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Alabama A&M Univ 27, Tennessee State 14\xa0\xa0', '\xa0']
```

In [31]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2005=[q.replace("Eastern Illinois","Eastern Illinois") for q in schedule2005]
schedule2005=[r.replace("No. Carolina A&T St.","NorthCarolina A&T") for r in schedule2005]
schedule2005=[s.replace("Alabama A&M Univ","Alabama A&M") for s in schedule2005]
schedule2005=[t.replace("Nashville, TN","Nashville, TN") for t in schedule2005]
schedule2005=[u.replace("Samford","Samford University") for u in schedule2005]
schedule2005=[v.replace("SEMO","Southeast Missouri") for v in schedule2005]
schedule2005=[w.replace("EKU","Eastern Kentucky") for w in schedule2005]
schedule2005=[x.replace("Cape Girardeau","CapeGirardeau") for x in schedule2005]
schedule2005=[a.replace("-"," ") for a in schedule2005]
schedule2005=[b.replace(".","") for b in schedule2005]
schedule2005=[c.replace("\xa0","") for c in schedule2005]
schedule2005=[d.replace(",","") for d in schedule2005]
schedule2005=[t.strip() for t in schedule2005]
print(schedule2005)
```

```
['Nov 19 2005', 'Nashville TN', 'Eastern Kentucky 49 Tennessee State 0', '', 'Nov 12 2005', 'CapeGirardeau Mo', 'Southeast Missouri 32 Tennessee State 24', '', 'Nov 05 2005', 'Nashville TN', 'Eastern Illinois 27 Tennessee State 3', '', 'Oct 29 2005', 'Birmingham Ala', 'Samford University 38 Tennessee State 0', '', 'Oct 22 2005', 'Jacksonville Ala', 'Jacksonville State 33 Tennessee State 3', '', 'Oct 13 2005', 'Cookeville Tenn', 'Tennessee State 31 Tennessee Tech 28', '', 'Oct 01 2005', 'Indianapolis Ind', 'NorthCarolina A&T 16 Tennessee State 3', '', 'Sep 24 2005', 'Atlanta GA', 'Florida A&M 26 Tennessee State 23', '', 'Sep 17 2005', 'Martin Tenn', 'Tennessee Martin 42 Tennessee State 20', '', 'Sep 10 2005', 'Memphis Tenn', 'Tennessee State 26 Jackson State 14', '', 'Sep 03 2005', 'Nashville TN', 'Alabama A&M 27 Tennessee State 14', '']
```

In [32]:
```
#2004 data
#use requests.get() to get web page with 2004 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/57C54C44-0B63-4F94-99EB-2BB
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2004 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2004)
```

```
['Nov 20, 2004\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Murray State 30, Tennessee State 13\xa0\xa0', '\xa0', 'Nov 13, 2004\xa0\xa0', 'Richmond KY    \xa0\xa0', 'Eastern Kentucky 29, Tennessee State 14\xa0\xa0', '\xa0', 'Nov 06, 2004\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Southeast Missouri 38, Tennessee State 30\xa0\xa0', '\xa0', 'Oct 30, 2004\xa0\xa0', 'Charleston, Illinois\xa0\xa0', 'Eastern Illinois 34, Tennessee State 24\xa0\xa0', '\xa0', 'Oct 23, 2004\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Samford University 34, Tennessee State 24\xa0\xa0', '\xa0', 'Oct 16, 2004\xa0\xa0', 'Jacksonville, Ala.    \xa0\xa0', 'Jacksonville State 35, Tennessee State 24\xa0\xa0', '\xa0', 'Oct 02, 2004\xa0\xa0', 'RCA Dome    \xa0\xa0', 'South Carolina State 30, Tennessee State 27\xa0\xa0', '\xa0', 'Sep 25, 2004\xa0\xa0', 'Atlanta, Georgia\xa0\xa0', 'Florida A&M 21, Tennessee State 15\xa0\xa0', '\xa0', 'Sep 18, 2004\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Jackson State 20, Tennessee State 14\xa0\xa0', '\xa0', 'Sep. 4, 2003\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 27, Tennessee-Martin 13\xa0\xa0', '\xa0', 'Sep 04, 2004\xa0\xa0', 'Nashville, TN    \xa0\xa0', 'Tennessee State 42, Alabama A&M 7\xa0\xa0', '\xa0']
```

In [33]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2004=[p.replace('Sep. 4, 2003','Sep 04 2004') for p in schedule2004]
schedule2004=[q.replace("Nashville, TN","Nashville, TN") for q in schedule2004]
schedule2004=[r.replace("South Carolina State","SouthCarolina State") for r in schedule2004]
schedule2004=[s.replace("RCA Dome","Indianapolis, IN") for s in schedule2004]
schedule2004=[t.replace("Richmond KY","Richmond, KY") for t in schedule2004]
schedule2004=[a.replace("-"," ") for a in schedule2004]
schedule2004=[b.replace(".","") for b in schedule2004]
schedule2004=[c.replace("\xa0","") for c in schedule2004]
schedule2004=[d.replace(",","") for d in schedule2004]
schedule2004=[t.strip() for t in schedule2004]

#adding blank elements to list to make it similar to other lists being concatenated
#can't assign it back to itself or I'll get a Nonetype error
schedule2004.insert(3, "")
schedule2004.insert(7, "")
schedule2004.insert(11, "")
schedule2004.insert(15, "")
schedule2004.insert(19, "")
schedule2004.insert(23, "")
schedule2004.insert(27, "")
schedule2004.insert(31, "")
schedule2004.insert(35, "")
schedule2004.insert(39, "")
schedule2004.insert(43, "")
print(schedule2004)
```

```
['Nov 20 2004', 'Nashville TN', 'Murray State 30 Tennessee State 13', '', 'Nov 13 2004', 'Richmond KY', 'Eastern Kentucky 29 Tennessee State 14', '', 'Nov 06 2004', 'Nashville TN', 'Southeast Missouri 38 Tennessee State 30', '', 'Oct 30 2004', 'Charleston Illinois', 'Eastern Illinois 34 Tennessee State 24', '', 'Oct 23 2004', 'Nashville TN', 'Samford University 34 Tennessee State 24', '', 'Oct 16 2004', 'Jacksonville Ala', 'Jacksonville State 35 Tennessee State 24', '', 'Oct 02 2004', 'Indianapolis IN', 'SouthCarolina State 30 Tennessee State 27', '', 'Sep 25 2004', 'Atlanta Georgia', 'Florida A&M 21 Tennessee State 15', '', 'Sep 18 2004', 'Memphis Tenn', 'Jackson State 20 Tennessee State 14', '', 'Sep 04 2004', 'Nashville TN', 'Tennessee State 27 Tennessee Martin 13', '', 'Sep 04 2004', 'Nashville TN', 'Tennessee State 42 Alabama A&M 7', '']
```

In [34]:
```
#2003 data
#use requests.get() to get web page with 2003 data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/A5FB3AA1-8151-496A-A022-7548
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)
#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get list of data from web page
schedule2003 = mytree.xpath('body/tr/td/font[@color="#000000"]/text()')
#look at list
print(schedule2003)
```

```
['Nov 22, 2003\xa0\xa0', 'Murray, Ky.    \xa0\xa0', 'Tennessee State 35, Murray State 10\xa0\xa0', '\xa0', 'Nov 15, 2003\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Eastern Kentucky 35, Tennessee State 38\xa0\xa0', '\xa0', 'Nov 08, 2003\xa0\xa0', 'Cape Girardeau, Mo.\xa0\xa0', 'Southeast Missouri 32, Tennessee State 35\xa0\xa0', '\xa0', 'Nov 01, 2003\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Tennessee State 24, Eastern Illinois 14\xa0\xa0', '\xa0', 'Oct 25, 2003\xa0\xa0', 'Birmingham, Ala.    \xa0\xa0', 'Tennessee State 29, Samford 24\xa0\xa0', '\xa0', 'Oct 18, 2003\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'Jacksonville State 34, Tennessee State 26\xa0\xa0', '\xa0', 'Oct 11, 2003\xa0\xa0', 'Cookeville, Tenn.    \xa0\xa0', 'Tennessee State 27, Tennessee Tech 23\xa0\xa0', '\xa0', 'Oct 04, 2003\xa0\xa0', 'Indianapolis, Ind.    \xa0\xa0', 'South Carolina State 28, Tennessee State 27\xa0\xa0', '\xa0', 'Sep 27, 2003\xa0\xa0', 'Atlanta, Ga.    \xa0\xa0', 'Florida A&M 20, Tennessee State 10\xa0\xa0', '\xa0', 'Sep 20, 2003\xa0\xa0', 'Martin, Tenn.    \xa0\xa0', 'Tennessee State 27, Tennessee-Martin 10\xa0\xa0', '\xa0', 'Sep 13, 2003\xa0\xa0', 'Memphis, Tenn.    \xa0\xa0', 'Tennessee State 26, Jackson State 24\xa0\xa0', '\xa0', 'Sep. 6, 2003\xa0\xa0', 'Huntsville, Alabama\xa0\xa0', 'Alabama A&M 31, Tennessee State 24\xa0\xa0', '\xa0', 'Aug 30, 2003\xa0\xa0', 'Nashville, Tenn.    \xa0\xa0', 'South Carolina State 27, Tennessee State 20\xa0\xa0', '\xa0']
```

In [35]:
```
#get rid of extra text and spaces in list
#put text for school names and city into format that's compatible for data frame
schedule2003=[p.replace("Univ.","") for p in schedule2003]
schedule2003=[q.replace("South Carolina State","SouthCarolina State") for q in schedule2003]
schedule2003=[r.replace("Samford","Samford University") for r in schedule2003]
schedule2003=[s.replace("Tigers","") for s in schedule2003]
schedule2003=[t.replace("Cape Girardeau","CapeGirardeau") for t in schedule2003]
schedule2003=[a.replace("-"," ") for a in schedule2003]
schedule2003=[b.replace(".","") for b in schedule2003]
schedule2003=[c.replace("\xa0","") for c in schedule2003]
schedule2003=[d.replace(",","") for d in schedule2003]
schedule2003=[t.strip() for t in schedule2003]
schedule2003

#adding blank elements to list to make it similar to other lists being concatenated
#can't assign it back to itself or I'll get a Nonetype error
schedule2003.insert(3, "")
schedule2003.insert(7, "")
schedule2003.insert(11, "")
schedule2003.insert(15, "")
schedule2003.insert(19, "")
schedule2003.insert(23, "")
schedule2003.insert(27, "")
schedule2003.insert(31, "")
schedule2003.insert(35, "")
schedule2003.insert(39, "")
schedule2003.insert(43, "")
schedule2003.insert(47, "")
print(schedule2003)
```

```
['Nov 22 2003', 'Murray Ky', 'Tennessee State 35 Murray State 10', '', 'Nov 15 2003', 'Nashville Tenn', 'Eastern Kentucky 35 Tennessee State 38', '', 'Nov 08 2003', 'CapeGirardeau Mo', 'Southeast Missouri 32 Tennessee State 35', '', 'Nov 01 2003', 'Nashville Tenn', 'Tennessee State 24 Eastern Illinois 14', '', 'Oct 25 2003', 'Birmingham Ala', 'Tennessee State 29 Samford University 24', '', 'Oct 18 2003', 'Nashville Tenn', 'Jacksonville State 34 Tennessee State 26', '', 'Oct 11 2003', 'Cookeville Tenn', 'Tennessee State 27 Tennessee Tech 23', '', 'Oct 04 2003', 'Indianapolis Ind', 'SouthCarolina State 28 Tennessee State 27', '', 'Sep 27 2003', 'Atlanta Ga', 'Florida A&M 20 Tennessee State 10', '', 'Sep 20 2003', 'Martin Tenn', 'Tennessee State 27 Tennessee Martin 10', '', 'Sep 13 2003', 'Memphis Tenn', 'Tennessee State 26 Jackson State 24', '', 'Sep 6 2003', 'Huntsville Alabama', 'Alabama A&M 31 Tennessee State 24', '', 'Aug 30 2003', 'Nashville Tenn', 'SouthCarolina State 27 Tennessee State 20', '']
```

In [36]:
```
#get number of elements for 2003 list
len(schedule2003)
```

Out[36]: 48

In [37]:
```
#get number of elements for 2004 list
len(schedule2004)
```

Out[37]: 44

In [38]:
```
#get number of elements for 2005 list
len(schedule2005)
```

Out[38]: 44

In [39]:
```
#get number of elements for 2006 list
len(schedule2006)
```
Out[39]: 44

In [40]:
```
#get number of elements for 2007 list
len(schedule2007)
```
Out[40]: 44

In [41]:
```
#get number of elements for 2008 list
len(schedule2008)
```
Out[41]: 48

In [42]:
```
#get number of elements for 2009 list
len(schedule2009)
```
Out[42]: 44

In [43]:
```
#get number of elements for 2010 list
len(schedule2010)
```
Out[43]: 44

In [44]:
```
#get number of elements for 2011 list
len(schedule2011)
```
Out[44]: 44

In [45]:
```
#get number of elements for 2012 list
len(schedule2012)
```
Out[45]: 44

In [46]:
```
#get number of elements for 2013 list
len(schedule2013)
```
Out[46]: 56

In [47]:
```
# get number of elements for 2014 list
len(schedule2014)
```
Out[47]: 48

In [48]:
```
# get number of elements for 2015 list
len(schedule2015)
```
Out[48]: 40

In [49]:
```
# get number of elements for 2016 list
len(schedule2016)
```
Out[49]: 44

In [50]:
```
# get number of elements for 2017 list
len(schedule2017)
```
Out[50]: 44

In [51]:
```
# get number of elements for 2018 list
len(schedule2018)
```
Out[51]: 36

In [52]:
```
# get number of elements for 2019 list
len(schedule2019)
```
Out[52]: 48

In [53]:
```
#create a single list for all years
fulldata=schedule2003+schedule2004+schedule2005+schedule2006+schedule2007+schedule2008+schedule2009+schedule201
#get number of elements for full data
len(fulldata)
```
Out[53]: 764

In [54]:
```
#create list of sublists from fulldata
#each sublist has 4 elements and represents a single game
#using list comprehension that takes every 4 elements of fulldata in order and creates a sublist inside larger
list_of_sublists=[fulldata[i:i+4] for i in range(0,len(fulldata),4)]
#print 1st 5 elements of list
print(list_of_sublists[:5])
```
```
[['Nov 22 2003', 'Murray Ky', 'Tennessee State 35 Murray State 10', ''], ['Nov 15 2003', 'Nashville Tenn', 'Eastern Kentucky 43 Tennessee State 38', ''], ['Nov 08 2003', 'CapeGirardeau Mo', 'Southeast Missouri 52 Tennessee State 35', ''], ['Nov 01 2003', 'Nashville Tenn', 'Tennessee State 24 Eastern Illinois 14', ''], ['Oct 25 2003', 'Birmingham Ala', 'Tennessee State 29 Samford University 24', '']]
```

In [55]:
```
#create dataframe from list of sublists
#adding columns names(reserve is to see if any text that didn't fit in other 3 columns)
df=pd.DataFrame(list_of_sublists, columns=['date','location','scores', 'reserve'])
df.head()
```
Out[55]:

	date	location	scores	reserve
0	Nov 22 2003	Murray Ky	Tennessee State 35 Murray State 10	
1	Nov 15 2003	Nashville Tenn	Eastern Kentucky 43 Tennessee State 38	
2	Nov 08 2003	CapeGirardeau Mo	Southeast Missouri 52 Tennessee State 35	
3	Nov 01 2003	Nashville Tenn	Tennessee State 24 Eastern Illinois 14	
4	Oct 25 2003	Birmingham Ala	Tennessee State 29 Samford University 24	

In [56]:
```
#check reserve column to see if there any values in there to see if any text extended out of the scores column
df['reserve'].value_counts()
```
Out[56]:
```
    191
Name: reserve, dtype: int64
```

Data Cleaning- dropping unneeded variables and creating needed variables

In [57]:
```
#drop reserve column since it's empty
df=df.drop('reserve',axis=1)
df.head()
```
Out[57]:

	date	location	scores
0	Nov 22 2003	Murray Ky	Tennessee State 35 Murray State 10
1	Nov 15 2003	Nashville Tenn	Eastern Kentucky 43 Tennessee State 38
2	Nov 08 2003	CapeGirardeau Mo	Southeast Missouri 52 Tennessee State 35
3	Nov 01 2003	Nashville Tenn	Tennessee State 24 Eastern Illinois 14
4	Oct 25 2003	Birmingham Ala	Tennessee State 29 Samford University 24

In [58]:
```
#get number of rows and columns in file
df.shape
```
Out[58]: (191, 3)

In [59]:
```
#making date a datetime column
df['date']=pd.to_datetime(df['date'],format='%b %d %Y')
#split location column into city and state columns
df[['city','state']]=df.location.str.split(expand=True)
#expand scores column into win1 (1st part of winner name), win2 (2nd part of winner name), winscore (winner sco
#los1 (1st part of loser name), los2 (2nd part of loser name), losscore (loser score) columns
df[['win1','win2','winscore','los1','los2','losscore']]=df.scores.str.split(expand=True)
#create combined school names (names of 1 school in a single cell with a space between 1st and 2nd prts of name
df['loser']=df['los1']+ ' ' +df['los2']
df['winner']=df['win1']+ ' ' +df['win2']
#add home/away column to dataframe
df['locale'] = np.where(df['city']== 'Nashville', 'Home', 'Away')
#add column for TSU score and make it numeric
df['TSU score']=pd.to_numeric(np.where(df['winner']=='Tennessee State',df['winscore'],df['losscore']))
#add opponent score column and make it numeric
df['opponent score']=pd.to_numeric(np.where(df['winner']=='Tennessee State',df['losscore'],df['winscore']))
#create winscore and losscore numeric
df['winscore']=pd.to_numeric(df['winscore'])
df['losscore']=pd.to_numeric(df['losscore'])
#create score difference column (negative numbers means TSU had a lower score than opponent)
df['scorediff']=df['TSU score']-df['opponent score']
#create absolute value of score difference column
df['scorediff_abs']=abs(df['scorediff'])
#create column for win and loss
df['winloss']=np.where(df['winner']=='Tennessee State','Win','Loss')
df.head()
```
Out[59]:

	date	location	scores	city	state	win1	win2	winscore	los1	los2	losscore	loser	winner	lo
0	2003-11-22	Murray Ky	Tennessee State 35 Murray State 10	Murray	Ky	Tennessee	State	35	Murray	State	10	Murray State	Tennessee State	A
1	2003-11-15	Nashville Tenn	Eastern Kentucky 43 Tennessee State 38	Nashville	Tenn	Eastern	Kentucky	43	Tennessee	State	38	Tennessee State	Eastern Kentucky	H
2	2003-11-08	CapeGirardeau Mo	Southeast Missouri 52 Tennessee State 35	CapeGirardeau	Mo	Southeast	Missouri	52	Tennessee	State	35	Tennessee State	Southeast Missouri	A
3	2003-11-01	Nashville Tenn	Tennessee State 24 Eastern Illinois 14	Nashville	Tenn	Tennessee	State	24	Eastern	Illinois	14	Eastern Illinois	Tennessee State	H
4	2003-10-25	Birmingham Ala	Tennessee State 29 Samford University 24	Birmingham	Ala	Tennessee	State	29	Samford	University	24	Samford University	Tennessee State	A

In [60]:
```
#dropping unneccesary columns
df=df.drop(['location','scores','win1','win2','los1','los2'], axis=1)
df.head()
```
Out[60]:

	date	city	state	winscore	losscore	loser	winner	locale	TSU score	opponent score	scorediff	scorediff_abs	winloss
0	2003-11-22	Murray	Ky	35	10	Murray State	Tennessee State	Away	35	10	25	25	Win
1	2003-11-15	Nashville	Tenn	43	38	Tennessee State	Eastern Kentucky	Home	38	43	-5	5	Loss
2	2003-11-08	CapeGirardeau	Mo	52	35	Tennessee State	Southeast Missouri	Away	35	52	-17	17	Loss
3	2003-11-01	Nashville	Tenn	24	14	Eastern Illinois	Tennessee State	Home	24	14	10	10	Win
4	2003-10-25	Birmingham	Ala	29	24	Samford University	Tennessee State	Away	29	24	5	5	Win

In [61]:
```
#create a year variable
df['year']=df.date.dt.year
df.head()
```
Out[61]:

	date	city	state	winscore	losscore	loser	winner	locale	TSU score	opponent score	scorediff	scorediff_abs	winloss	year
0	2003-11-22	Murray	Ky	35	10	Murray State	Tennessee State	Away	35	10	25	25	Win	2003
1	2003-11-15	Nashville	Tenn	43	38	Tennessee State	Eastern Kentucky	Home	38	43	-5	5	Loss	2003
2	2003-11-08	CapeGirardeau	Mo	52	35	Tennessee State	Southeast Missouri	Away	35	52	-17	17	Loss	2003
3	2003-11-01	Nashville	Tenn	24	14	Eastern Illinois	Tennessee State	Home	24	14	10	10	Win	2003
4	2003-10-25	Birmingham	Ala	29	24	Samford University	Tennessee State	Away	29	24	5	5	Win	2003

In [62]:
```
#clean up city columns (some names were collasped because they had multiple word names)
#get names in city column
df['city'].unique()
```
Out[62]:
```
array(['Murray', 'Nashville', 'CapeGirardeau', 'Birmingham', 'Cookeville',
       'Atlanta', 'Memphis', 'Huntsville', 'Richmond', 'Charleston',
       'Jacksonville', 'Indianapolis', 'Martin', 'BatonRouge',
       'Clarksville', 'AirForceAcademy', 'DaytonaBeach', 'StLouis',
       'Tallahassee', 'Montgomery', 'Tampa', 'Murfreesboro'], dtype=object)
```

In [63]:
```
#put appropriate spaces and period in city names
df['city']=df['city'].replace("CapeGirardeau","Cape Girardeau").replace("AirForceAcademy","Air Force Academy")
df['city']=df['city'].replace("DaytonaBeach","Daytona Beach").replace("StLouis", "St. Louis")
df['city']=df['city'].replace("BatonRouge","Baton Rouge")
#check names in city column
df['city'].unique()
```
Out[63]:
```
array(['Murray', 'Nashville', 'Cape Girardeau', 'Birmingham',
       'Cookeville', 'Atlanta', 'Memphis', 'Huntsville', 'Richmond',
       'Charleston', 'Jacksonville', 'Indianapolis', 'Martin',
       'Baton Rouge', 'Clarksville', 'Air Force Academy', 'Daytona Beach',
       'St. Louis', 'Tallahassee', 'Montgomery', 'Tampa', 'Murfreesboro'],
      dtype=object)
```

In [64]:
```
#making values in state column data into post office state abbreviations
#get names from column
df['state'].unique()
```
Out[64]:
```
array(['Ky', 'Tenn', 'Mo', 'Ala', 'Tennessee', 'Ga', 'Alabama', 'TN',
       'KY', 'Illinois', 'IN', 'Georgia', 'Ind', 'GA', 'IL', 'AL', 'LA',
       'Ill', 'Colo', 'Fla', 'Missouri'], dtype=object)
```

In [65]:
```
#make state abbreviations consistent
df['state']=df['state'].replace(["Ga","Georgia"],"GA").replace(["Tenn","Tennessee"],"TN").replace("Ind","IN")
df['state']=df['state'].replace(["Mo","Missouri"],"MO").replace(["Ala","Alabama"],"AL").replace("Ky","KY")
df['state']=df['state'].replace(["Ill","Illinois"],"IL").replace("Fla","FL").replace("Colo","CO").replace("Ind'
df['state'].unique()
```
Out[65]:
```
array(['KY', 'TN', 'MO', 'AL', 'GA', 'IL', 'IN', 'LA', 'CO', 'FL'],
      dtype=object)
```

In [66]:
```
#checking loser column values
df['loser'].unique()
```
Out[66]:
```
array(['Murray State', 'Tennessee State', 'Eastern Illinois',
       'Samford University', 'Tennessee Tech', 'Tennessee Martin',
       'Jackson State', 'SouthCarolina State', 'Southeast Missouri',
       'Alabama A&M', 'Jacksonville State', 'Austin Peay',
       'Eastern Kentucky', 'Southern University', 'NorthCarolina A&T',
       'Florida A&M', 'Arkansas PineBluff', 'Bethune Cookman',
       'Butler University', 'Central State', 'Murray St', 'Edward Waters',
       'Alabama State', 'VirginiaU Lynchburg', 'Georgia State',
       'Mississippi Valley'], dtype=object)
```

In [67]:
```
#correcting spelling of school names in loser column
df['loser']=df['loser'].replace(["Murray St","Murray State"], "Murray State University")
df['loser']=df['loser'].replace("Tennessee State","Tennessee State University")
df['loser']=df['loser'].replace("Eastern Illinois","Eastern Illinois University")
df['loser']=df['loser'].replace("Tennessee Tech","Tennessee Tech University")
df['loser']=df['loser'].replace("Tennessee Martin", "University of Tennessee Martin")
df['loser']=df['loser'].replace("Jackson State","Jackson State University")
df['loser']=df['loser'].replace("SouthCarolina State","South Carolina State University")
df['loser']=df['loser'].replace('Southeast Missouri','Southeast Missouri State University')
df['loser']=df['loser'].replace('Alabama A&M','Alabama A&M University')
df['loser']=df['loser'].replace('Jacksonville State','Jacksonville State University')
df['loser']=df['loser'].replace('Austin Peay','Austin Peay State University')
df['loser']=df['loser'].replace('Eastern Kentucky','Eastern Kentucky University')
df['loser']=df['loser'].replace('Southern University','Southern University & A&M College')
df['loser']=df['loser'].replace("NorthCarolina A&T", "North Carolina A&T State University")
df['loser']=df['loser'].replace("Florida A&M", "Florida A&M University")
df['loser']=df['loser'].replace("Arkansas PineBluff","University of Arkansas Pine Bluff")
df['loser']=df['loser'].replace('Bethune Cookman','Bethune Cookman University')
df['loser']=df['loser'].replace('Central State','Central State University')
df['loser']=df['loser'].replace('Edward Waters','Edward Waters College')
df['loser']=df['loser'].replace('Alabama State','Alabama State University')
df['loser']=df['loser'].replace("VirginiaU Lynchburg","Virginia University Lynchburg")
df['loser']=df['loser'].replace('Georgia State','Georgia State University')
df['loser']=df['loser'].replace('Mississippi Valley','Mississippi Valley State University')
#check values
df['loser'].unique()
```
Out[67]:
```
array(['Murray State University', 'Tennessee State University',
       'Eastern Illinois University', 'Samford University',
       'Tennessee Tech University', 'University of Tennessee Martin',
       'Jackson State University', 'South Carolina State University',
       'Southeast Missouri State University', 'Alabama A&M University',
       'Jacksonville State University', 'Austin Peay State University',
       'Eastern Kentucky University', 'Southern University & A&M College',
       'North Carolina A&T State University', 'Florida A&M University',
       'University of Arkansas Pine Bluff', 'Bethune Cookman University',
       'Butler University', 'Central State University',
       'Edward Waters College', 'Alabama State University',
       'Virginia University Lynchburg', 'Georgia State University',
       'Mississippi Valley State University'], dtype=object)
```

In [68]:
```
#checking winner column values
df['winner'].unique()
```
Out[68]:
```
array(['Tennessee State', 'Eastern Kentucky', 'Southeast Missouri',
       'Jacksonville State', 'Florida A&M', 'Alabama A&M', 'Murray State',
       'Eastern Illinois', 'Samford University', 'SouthCarolina State',
       'NorthCarolina A&T', 'Tennessee Martin', 'Vanderbilt University',
       'Southern University', 'Austin Peay', 'Tennessee Tech',
       'Jackson State', 'Air Force', 'Bethune Cookman', 'UT Martin',
       'Alabama State', 'Murray St', 'Pine Bluff', 'Middle Tennessee'],
      dtype=object)
```

In [69]:
```
#correcting school names in winner column
df['winner']=df['winner'].replace("Tennessee State","Tennessee State University")
df['winner']=df['winner'].replace('Eastern Kentucky','Eastern Kentucky University')
df['winner']=df['winner'].replace('Southeast Missouri','Southeast Missouri State University')
df['winner']=df['winner'].replace('Jacksonville State','Jacksonville State University')
df['winner']=df['winner'].replace("Florida A&M", "Florida A&M University")
df['winner']=df['winner'].replace('Alabama A&M','Alabama A&M University')
df['winner']=df['winner'].replace(["Murray St","Murray State"], "Murray State University")
df['winner']=df['winner'].replace("Eastern Illinois","Eastern Illinois University")
df['winner']=df['winner'].replace("SouthCarolina State","South Carolina State University")
df['winner']=df['winner'].replace("NorthCarolina A&T", "North Carolina A&T State University")
df['winner']=df['winner'].replace(["UT Martin","Tennessee Martin"], "University of Tennessee Martin")
df['winner']=df['winner'].replace('Southern University','Southern University & A&M College')
df['winner']=df['winner'].replace('Austin Peay','Austin Peay State University')
df['winner']=df['winner'].replace("Tennessee Tech","Tennessee Tech University")
df['winner']=df['winner'].replace("Jackson State","Jackson State University")
df['winner']=df['winner'].replace("Air Force", "Air Force Academy")
df['winner']=df['winner'].replace('Bethune Cookman','Bethune Cookman University')
df['winner']=df['winner'].replace('Alabama State','Alabama State University')
df['winner']=df['winner'].replace("Pine Bluff","University of Arkansas Pine Bluff")
df['winner']=df['winner'].replace('Middle Tennessee','Middle Tennessee State University')
#check changes to winner column
df['winner'].unique()
```
Out[69]:
```
array(['Tennessee State University', 'Eastern Kentucky University',
       'Southeast Missouri State University',
       'Jacksonville State University', 'Florida A&M University',
       'Alabama A&M University', 'Murray State University',
       'Eastern Illinois University', 'Samford University',
       'South Carolina State University',
       'North Carolina A&T State University',
       'University of Tennessee Martin', 'Vanderbilt University',
       'Southern University & A&M College',
       'Austin Peay State University', 'Tennessee Tech University',
       'Jackson State University', 'Air Force Academy',
       'Bethune Cookman University', 'Alabama State University',
       'University of Arkansas Pine Bluff',
       'Middle Tennessee State University'], dtype=object)
```

In [70]:
```
#sort rows by date
df=df.sort_values('date',ignore_index=True)
df.head()
```
Out[70]:

	date	city	state	winscore	losscore	loser	winner	locale	TSU score	opponent score	scorediff	scorediff_abs	winloss	year
0	2003-08-30	Nashville	TN	37	20	South Carolina State University	Tennessee State University	Home	37	20	17	17	Win	2003
1	2003-09-06	Huntsville	AL	31	24	Tennessee State University	Alabama A&M University	Away	24	31	-7	7	Loss	2003
2	2003-09-13	Memphis	TN	44	14	Jackson State University	Tennessee State University	Away	44	14	30	30	Win	2003
3	2003-09-20	Atlanta	GA	10	7	Tennessee State University	Florida A&M University	Away	7	10	-3	3	Loss	2003
4	2003-09-27	Nashville	TN	41	10	University of Tennessee Martin	Tennessee State University	Home	41	10	31	31	Win	2003

In [71]:
```
#check data types of columns
df.dtypes
```
Out[71]:
```
date              datetime64[ns]
city                      object
state                     object
winscore                   int64
losscore                   int64
loser                     object
winner                    object
locale                    object
TSU score                  int64
opponent score             int64
scorediff                  int64
scorediff_abs              int64
winloss                   object
year                       int64
dtype: object
```

In [72]:
```
#get number of rows and columns in dataframe
df.shape
```
Out[72]: (191, 14)

In [73]:
```
#save data frame with scores to folder
df.to_csv('scores.csv',encoding='utf-8')
```

Web Scraping-getting individual years of stats data from TSU stats websites

Data Wrangling-putting data into lists and data frame

In [74]:
```
#2003 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/A5FB3AA1-8151-496A-A022-754F

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:120:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace('Sep. 6, 2003','Sep 06 2003') for a in date]
date=[a.replace(",","") for a in date]
print(date)
```
```
['Aug 30 2003', 'Sep 06 2003', 'Sep 13 2003', 'Sep 20 2003', 'Sep 27 2003', 'Oct 11 2003', 'Oct 18 2003', 'Oct 25 2003', 'Nov 01 2003', 'Nov 08 2003', 'Nov 15 2003', 'Nov 22 2003']
```

In [75]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:120:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[75]:
```
array([18124, 18085, 52603, 70185,  8434,  8127,  8023, 10360, 25037,
        5375,  3875,  2814], dtype=int64)
```

In [76]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[124:380:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[76]:
```
array([113, 141, 209, 153, 142, 182,  47, 132, 223, 149, 144, 191],
      dtype=int64)
```

In [77]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[128:400:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[77]:
```
array([365, 146, 132, 138, 285, 297, 175, 271, 184, 196, 371, 195],
      dtype=int64)
```

In [78]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[136:400:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[78]:
```
array([ 69, 147, 138,  86,  11,  33,  90,  50,   8, 139, 112,  63],
      dtype=int64)
```

In [79]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[140:400:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[79]:
```
array([ 88,   8, 107,   2,  29,  52, -17,  45,   9,  18,   0,  15],
      dtype=int64)
```

In [80]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[447:700:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[80]:
```
array([69, 68, 84, 50, 76, 74, 79, 83, 65, 65, 75, 67], dtype=int64)
```

In [81]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[449:700:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[81]:
```
array([33, 22, 26, 22, 46, 44,  9, 34, 23, 23, 32, 37], dtype=int64)
```

In [82]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[450:700:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[82]:
```
array([3., 0., 1., 0., 4., 4., 1., 2., 2., 2., 4., 5.])
```

In [83]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[451:700:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[83]:
```
array([21,  0,  4,  0, 22, 30,  9, 24, 12, 14, 20, 28], dtype=int64)
```

In [84]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[753:969:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[84]:
```
array([ 5,  7,  5,  6,  3,  2,  6,  8,  5,  5, 10,  7], dtype=int64)
```

In [85]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2003=pd.DataFrame(list_of_dicts)
df2003.head()
```
Out[85]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Aug 30 2003	18124	113	365	69	88	69	33	3.0	21
1	Sep 06 2003	18085	141	146	147	8	68	22	0.0	0
2	Sep 13 2003	52603	209	132	138	107	84	26	1.0	4
3	Sep 20 2003	70185	153	138	86	2	50	22	0.0	0
4	Sep 27 2003	8434	142	285	11	29	76	46	4.0	22

In [86]:
```
#creating date column as a datetime column
df2003['date']=pd.to_datetime(df2003['date'],format='%b %d %Y')
#creating year column
df2003['year']=df2003.date.dt.year
df2003.head()
```
Out[86]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2003-08-30	18124	113	365	69	88	69	33	3.0	21
1	2003-09-06	18085	141	146	147	8	68	22	0.0	0
2	2003-09-13	52603	209	132	138	107	84	26	1.0	4
3	2003-09-20	70185	153	138	86	2	50	22	0.0	0
4	2003-09-27	8434	142	285	11	29	76	46	4.0	22

In [87]:
```
#get info on dataframe
df2003.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12 entries, 0 to 11
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             12 non-null     datetime64[ns]
 1   attendance       12 non-null     int64
 2   TSUrushyards     12 non-null     int64
 3   TSUreceiveyards  12 non-null     int64
 4   TSUkreturnyards  12 non-null     int64
 5   TSUpreturnyards  12 non-null     int64
 6   TSUtackles       12 non-null     int64
 7   TSUtackleyd      12 non-null     int64
 8   TSUsacks         12 non-null     float64
 9   TSUsackyd        12 non-null     int64
 10  TSUpunt          12 non-null     int64
 11  year             12 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [88]:
```
#sort rows by date
df2003=df2003.sort_values('date',ignore_index=True)
df2003.head()
```
Out[88]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2003-08-30	18124	113	365	69	88	69	33	3.0	21
1	2003-09-06	18085	141	146	147	8	68	22	0.0	0
2	2003-09-13	52603	209	132	138	107	84	26	1.0	4
3	2003-09-20	70185	153	138	86	2	50	22	0.0	0
4	2003-09-27	8434	142	285	11	29	76	46	4.0	22

In [89]:
```
#save data frame to folder
df2003.to_csv('2003.csv',encoding='utf-8')
```

In [90]:
```
#2004 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/57C54C44-DE63-4F94-99EB-2BB3

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace('2003','2004') for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
date=[a.replace(" 4"," 04") for a in date]
print(date)
```
```
['Sep 04 2004', 'Sep 09 2004', 'Sep 18 2004', 'Sep 25 2004', 'Oct 02 2004', 'Oct 16 2004', 'Oct 23 2004', 'Oct 30 2004', 'Nov 06 2004', 'Nov 13 2004', 'Nov 20 2004']
```

In [91]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[91]:
```
array([25117,  7019, 55015, 67712, 51082, 14722,  5041,  5422, 23421,
        7100,  4864], dtype=int64)
```

In [92]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[92]:
```
array([152, 156, 164, 161,  22, 106, 290,  22, 243, 194,  83], dtype=int64)
```

In [93]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[93]:
```
array([203, 158, 149, 163, 173, 381, 211, 192, 215, 162, 142], dtype=int64)
```

In [94]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[94]:
```
array([ 24,  34,  39,  49,  57, 120, 127, 100,  76,  95, 130], dtype=int64)
```

In [95]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:380:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[95]:
```
array([26, 32, 10, 32,  0,  0, 36, 77, 22, 63,  0], dtype=int64)
```

In [96]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:650:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[96]:
```
array([74, 65, 63, 78, 91, 77, 88, 66, 63, 67, 66], dtype=int64)
```

In [97]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:650:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[97]:
```
array([35, 60, 19, 70, 17,  0, 23,  6, 31, 25, 16], dtype=int64)
```

In [98]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:650:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[98]:
```
array([2., 4., 2., 5., 1., 0., 1., 0., 2., 1., 0.])
```

In [99]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:650:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[99]:
```
array([12, 30,  7, 49,  4,  0,  5,  0, 12,  5,  0], dtype=int64)
```

In [100…
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[100…
```
array([4, 7, 7, 9, 9, 4, 4, 8, 7, 6, 8], dtype=int64)
```

In [101…
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2004=pd.DataFrame(list_of_dicts)
df2004.head()
```
Out[101…

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 04 2004	25117	152	203	24	26	74	35	2.0	12
1	Sep 09 2004	7019	156	158	34	32	65	60	4.0	30
2	Sep 18 2004	55015	164	149	39	10	63	19	2.0	7
3	Sep 25 2004	67712	161	163	49	32	78	70	5.0	49
4	Oct 02 2004	51082	22	173	57	0	91	17	1.0	4

In [102…
```
#creating date column as a datetime column
df2004['date']=pd.to_datetime(df2004['date'],format='%b %d %Y')
#creating year column
df2004['year']=df2004.date.dt.year
#make year variable
df2004.head()
```
Out[102…

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2004-09-04	25117	152	203	24	26	74	35	2.0	12
1	2004-09-09	7019	156	158	34	32	65	60	4.0	30
2	2004-09-18	55015	164	149	39	10	63	19	2.0	7
3	2004-09-25	67712	161	163	49	32	78	70	5.0	49
4	2004-10-02	51082	22	173	57	0	91	17	1.0	4

In [103]:
```
#get info on dataframe
df2004.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   date              11 non-null     datetime64[ns]
 1   attendance        11 non-null     int64
 2   TSUrushyards      11 non-null     int64
 3   TSUreceiveyards   11 non-null     int64
 4   TSUkreturnyards   11 non-null     int64
 5   TSUpreturnyards   11 non-null     int64
 6   TSUtackles        11 non-null     int64
 7   TSUtackleyd       11 non-null     int64
 8   TSUsacks          11 non-null     float64
 9   TSUsackyd         11 non-null     int64
 10  TSUpunt           11 non-null     int64
 11  year              11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [104]:
```
#sort values
df2004=df2004.sort_values('date',ignore_index=True)
df2004.head()
```
Out[104]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2004-09-04	25117	152	203	24	26	74	35	2.0	12
1	2004-09-09	7019	156	158	34	32	65	60	4.0	30
2	2004-09-18	55015	164	149	39	10	63	19	2.0	7
3	2004-09-25	67712	161	163	49	32	78	70	5.0	49
4	2004-10-02	51082	22	173	57	0	91	17	1.0	4

In [105]:
```
#save data frame to folder
df2004.to_csv('2004.csv',encoding='utf-8')
```

In [106]:
```
#2005 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/998728E2-12D4-4887-A990-249

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```
```
['Sep 03 2005', 'Sep 10 2005', 'Sep 17 2005', 'Sep 24 2005', 'Oct 01 2005', 'Oct 13 2005', 'Oct 22 2005', 'Oct
29 2005', 'Nov 05 2005', 'Nov 12 2005', 'Nov 19 2005']
```

In [107]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[107]:
```
array([25342, 48300, 5263, 56297, 42310, 10226, 6490, 8278, 23481,
       2512, 4779], dtype=int64)
```

In [108]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[108]:
```
array([ 72, 286, 77, 26, 117, 233, 157, 87, 50, 141, -18], dtype=int64)
```

In [109]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[109]:
```
array([154, 103, 170, 238, 115, 78, 165, 160, 196, 286, 150], dtype=int64)
```

In [110]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[110]:
```
array([ 62, 11, 110, 10, 99, 71, 114, 78, 90, 128, 138], dtype=int64)
```

In [111]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:380:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[111]:
```
array([32, 47, -2, 67, 49, 53, 14, 3, 0, 1, 0], dtype=int64)
```

In [112]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:650:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[112]:
```
array([47, 59, 95, 70, 75, 89, 62, 72, 61, 53, 64], dtype=int64)
```

In [113]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:650:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[113]:
```
array([20, 29, 17, 56, 56, 41, 9, 20, 26, 20, 6], dtype=int64)
```

In [114]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:650:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[114]:
```
array([2., 3., 1., 7., 6., 2., 0., 2., 1., 1., 0.])
```

In [115]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:650:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[115]:
```
array([13, 16, 6, 50, 37, 11, 0, 8, 14, 7, 0], dtype=int64)
```

In [116]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[116]:
```
array([4, 3, 6, 4, 6, 8, 5, 7, 5, 5, 8], dtype=int64)
```

In [117]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
      'attendance':attendance,
      'TSUrushyards':TSUrushyards,
      'TSUreceiveyards':TSUreceiveyards,
      'TSUkreturnyards':TSUkreturnyards,
      'TSUpreturnyards':TSUpreturnyards,
      'TSUtackles':TSUtackles,
      'TSUtackleyd':TSUtackleyd,
      'TSUsacks':TSUsacks,
      'TSUsackyd':TSUsackyd,
      'TSUpunt':TSUpunt}
df2005=pd.DataFrame(list_of_dicts)
df2005.head()
```
Out[117]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 03 2005	25342	72	154	62	32	47	20	2.0	13
1	Sep 10 2005	48300	286	103	11	47	59	29	3.0	16
2	Sep 17 2005	5263	77	170	110	-2	95	17	1.0	6
3	Sep 24 2005	56297	26	238	10	67	70	56	7.0	50
4	Oct 01 2005	42310	117	115	99	49	75	56	6.0	37

In [118]:
```
#creating date column as a datetime column
df2005['date']=pd.to_datetime(df2005['date'],format='%b %d %Y')
#creating year column
df2005['year']=df2005.date.dt.year
df2005.head()
```
Out[118]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2005-09-03	25342	72	154	62	32	47	20	2.0	13
1	2005-09-10	48300	286	103	11	47	59	29	3.0	16
2	2005-09-17	5263	77	170	110	-2	95	17	1.0	6
3	2005-09-24	56297	26	238	10	67	70	56	7.0	50
4	2005-10-01	42310	117	115	99	49	75	56	6.0	37

In [119]:
```
#get info about dataframe
df2005.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   date              11 non-null     datetime64[ns]
 1   attendance        11 non-null     int64
 2   TSUrushyards      11 non-null     int64
 3   TSUreceiveyards   11 non-null     int64
 4   TSUkreturnyards   11 non-null     int64
 5   TSUpreturnyards   11 non-null     int64
 6   TSUtackles        11 non-null     int64
 7   TSUtackleyd       11 non-null     int64
 8   TSUsacks          11 non-null     float64
 9   TSUsackyd         11 non-null     int64
 10  TSUpunt           11 non-null     int64
 11  year              11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [120]:
```
#sort values
df2005=df2005.sort_values('date',ignore_index=True)
df2005.head()
```
Out[120]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2005-09-03	25342	72	154	62	32	47	20	2.0	13
1	2005-09-10	48300	286	103	11	47	59	29	3.0	16
2	2005-09-17	5263	77	170	110	-2	95	17	1.0	6
3	2005-09-24	56297	26	238	10	67	70	56	7.0	50
4	2005-10-01	42310	117	115	99	49	75	56	6.0	37

In [121]:
```
#save data
df2005.to_csv('2005.csv',encoding='utf-8')
```

In [122]:
```
#2006 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/902C2D72-E881-4838-87F2-28A

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```
```
['Sep 02 2006', 'Sep 09 2006', 'Sep 16 2006', 'Sep 23 2006', 'Sep 30 2006', 'Oct 14 2006', 'Oct 21 2006', 'Oct
28 2006', 'Nov 04 2006', 'Nov 11 2006', 'Nov 18 2006']
```

In [123]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[123]:
```
array([19487, 10613, 53441, 27460, 57885, 9720, 11800, 18758, 5912,
       4271, 5500], dtype=int64)
```

In [124]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[124]:
```
array([156, 111, 146, 71, 227, 310, 103, 247, 49, 133, 95], dtype=int64)
```

In [125]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[125]:
```
array([206, 230, 167, 181, 210, 169, 260, 105, 162, 152, 162], dtype=int64)
```

In [126]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[126]:
```
array([105, 102, 77, 70, 98, 45, 119, 23, 115, 0, 126], dtype=int64)
```

In [127]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:380:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[127]:
```
array([ 0, 0, 24, 0, 16, 4, 20, 13, 0, 17, 7], dtype=int64)
```

In [128]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:650:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[128]:
```
array([63, 76, 63, 59, 54, 64, 77, 60, 68, 44, 70], dtype=int64)
```

In [129]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:650:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[129]:
```
array([ 3, 38, 18, 25, 14, 15, 26, 28, 22, 27, 30], dtype=int64)
```

In [130]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:650:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[130]:
```
array([0., 3., 1., 1., 1., 1., 1., 3., 1., 4., 2.])
```

In [131]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:650:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[131]:
```
array([ 0, 23, 6, 7, 4, 9, 20, 16, 10, 23, 14], dtype=int64)
```

In [132]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[132]:
```
array([2, 5, 2, 3, 4, 4, 6, 3, 4, 4, 1], dtype=int64)
```

In [133]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
      'attendance':attendance,
      'TSUrushyards':TSUrushyards,
      'TSUreceiveyards':TSUreceiveyards,
      'TSUkreturnyards':TSUkreturnyards,
      'TSUpreturnyards':TSUpreturnyards,
      'TSUtackles':TSUtackles,
      'TSUtackleyd':TSUtackleyd,
      'TSUsacks':TSUsacks,
      'TSUsackyd':TSUsackyd,
      'TSUpunt':TSUpunt}
df2006=pd.DataFrame(list_of_dicts)
df2006.head()
```
Out[133]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 02 2006	19487	156	206	105	0	63	3	0.0	0
1	Sep 09 2006	10613	111	230	102	0	76	38	3.0	23
2	Sep 16 2006	53441	146	167	77	24	63	18	1.0	6
3	Sep 23 2006	27460	71	181	70	0	59	25	1.0	7
4	Sep 30 2006	57885	227	210	98	16	54	14	1.0	4

In [134]:
```
#creating date column as a datetime column
df2006['date']=pd.to_datetime(df2006['date'],format='%b %d %Y')
#creating year column
df2006['year']=df2006.date.dt.year
df2006.head()
```
Out[134]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2006-09-02	19487	156	206	105	0	63	3	0.0	0
1	2006-09-09	10613	111	230	102	0	76	38	3.0	23
2	2006-09-16	53441	146	167	77	24	63	18	1.0	6
3	2006-09-23	27460	71	181	70	0	59	25	1.0	7
4	2006-09-30	57885	227	210	98	16	54	14	1.0	4

In [135]:
```
#get info on data frame
df2006.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   date              11 non-null     datetime64[ns]
 1   attendance        11 non-null     int64
 2   TSUrushyards      11 non-null     int64
 3   TSUreceiveyards   11 non-null     int64
 4   TSUkreturnyards   11 non-null     int64
 5   TSUpreturnyards   11 non-null     int64
 6   TSUtackles        11 non-null     int64
 7   TSUtackleyd       11 non-null     int64
 8   TSUsacks          11 non-null     float64
 9   TSUsackyd         11 non-null     int64
 10  TSUpunt           11 non-null     int64
 11  year              11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [136]:
```
#sort values
df2006=df2006.sort_values('date',ignore_index=True)
df2006.head()
```
Out[136]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2006-09-02	19487	156	206	105	0	63	3	0.0	0
1	2006-09-09	10613	111	230	102	0	76	38	3.0	23
2	2006-09-16	53441	146	167	77	24	63	18	1.0	6
3	2006-09-23	27460	71	181	70	0	59	25	1.0	7
4	2006-09-30	57885	227	210	98	16	54	14	1.0	4

In [137]:
```
#save data
df2006.to_csv('2006.csv',encoding='utf-8')
```

In [138]:
```
#2007 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/8C48DA5A-76BB-4378-BE95-5BA

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```
```
['Sep 01 2007', 'Sep 08 2007', 'Sep 15 2007', 'Sep 22 2007', 'Sep 29 2007', 'Oct 11 2007', 'Oct 20 2007', 'Oct
27 2007', 'Nov 03 2007', 'Nov 08 2007', 'Nov 17 2007']
```

In [139]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[139]:
```
array([23440, 50879, 8359, 15371, 56990, 9369, 11500, 8935, 24878,
       4193, 7859], dtype=int64)
```

In [140]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[140]:
```
array([206, 153, 238, 133, 177, 329, 169, 166, 280, 187, 290], dtype=int64)
```

In [141]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[141]:
```
array([176, 232, 233, 316, 309, 138, 127, 189, 69, 125, 173], dtype=int64)
```

In [142]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[142]:
```
array([154, 84, 83, 135, 111, 64, 122, 84, 6, 94, 222], dtype=int64)
```

In [143]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:380:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[143]:
```
array([35, 24, 14, 15, 33, 8, 0, 20, 77, -1, 31], dtype=int64)
```

In [144]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:650:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[144]:
```
array([56, 63, 64, 80, 78, 70, 77, 88, 93, 75, 92], dtype=int64)
```

In [145]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:650:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[145]:
```
array([17, 30, 44, 56, 28, 23, 19, 12, 25, 35, 14], dtype=int64)
```

In [146]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:650:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[146]:
```
array([2, 2, 3, 3, 3, 0, 1, 0, 3, 1, 1], dtype=int64)
```

In [147]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:650:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[147]:
```
array([15, 16, 26, 36, 23, 0, 10, 0, 15, 11, 8], dtype=int64)
```

In [148]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[148]:
```
array([ 6, 5, 3, 7, 4, 4, 10, 5, 3, 5, 3], dtype=int64)
```

In [149]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
      'attendance':attendance,
      'TSUrushyards':TSUrushyards,
      'TSUreceiveyards':TSUreceiveyards,
      'TSUkreturnyards':TSUkreturnyards,
      'TSUpreturnyards':TSUpreturnyards,
      'TSUtackles':TSUtackles,
      'TSUtackleyd':TSUtackleyd,
      'TSUsacks':TSUsacks,
      'TSUsackyd':TSUsackyd,
      'TSUpunt':TSUpunt}
df2007=pd.DataFrame(list_of_dicts)
df2007.head()
```
Out[149]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 01 2007	23440	206	176	154	35	56	17	2	15
1	Sep 08 2007	50879	153	232	84	24	63	30	2	16
2	Sep 15 2007	8359	238	233	83	14	64	44	3	26
3	Sep 22 2007	15371	133	316	135	15	80	56	3	36
4	Sep 29 2007	56990	177	309	111	33	78	28	3	23

In [150]:
```
#creating date column as a datetime column
df2007['date']=pd.to_datetime(df2007['date'],format='%b %d %Y')
#creating year column
df2007['year']=df2007.date.dt.year
df2007.head()
```
Out[150]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2007-09-01	23440	206	176	154	35	56	17	2	15
1	2007-09-08	50879	153	232	84	24	63	30	2	16
2	2007-09-15	8359	238	233	83	14	64	44	3	26
3	2007-09-22	15371	133	316	135	15	80	56	3	36
4	2007-09-29	56990	177	309	111	33	78	28	3	23

In [151]:
```
#get info about dataframe
df2007.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   date              11 non-null     datetime64[ns]
 1   attendance        11 non-null     int64
 2   TSUrushyards      11 non-null     int64
 3   TSUreceiveyards   11 non-null     int64
 4   TSUkreturnyards   11 non-null     int64
 5   TSUpreturnyards   11 non-null     int64
 6   TSUtackles        11 non-null     int64
 7   TSUtackleyd       11 non-null     int64
 8   TSUsacks          11 non-null     int64
 9   TSUsackyd         11 non-null     int64
 10  TSUpunt           11 non-null     int64
 11  year              11 non-null     int64
dtypes: datetime64[ns](1), int64(11)
memory usage: 1.2 KB
```

In [152]:
```
#sort values
df2007=df2007.sort_values('date',ignore_index=True)
df2007.head()
```
Out[152]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2007-09-01	23440	206	176	154	35	56	17	2	15
1	2007-09-08	50879	153	232	84	24	63	30	2	16
2	2007-09-15	8359	238	233	83	14	64	44	3	26
3	2007-09-22	15371	133	316	135	15	80	56	3	36
4	2007-09-29	56990	177	309	111	33	78	28	3	23

In [153]:
```
#save data
df2007.to_csv('2007.csv',encoding='utf-8')
```

In [154]:
```
#2008 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/E0564A83-37A0-4885-8811-87B

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:120:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```
```
['Aug 30 2008', 'Sep 06 2008', 'Sep 13 2008', 'Sep 20 2008', 'Sep 27 2008', 'Oct 04 2008', 'Oct 18 2008', 'Oct
25 2008', 'Nov 01 2008', 'Nov 08 2008', 'Nov 15 2008', 'Nov 22 2008']
```

In [155]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:120:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[155]:
```
array([10072, 28830, 50794, 8276, 50428, 7549, 9358, 9750, 24361,
       6393, 7956, 2292], dtype=int64)
```

In [156]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[124:380:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[156]:
```
array([107, 148, 148, 229, 136, 177, 377, 217, 212, 309, 177, 113],
      dtype=int64)
```

In [157]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[128:383:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[157]:
```
array([292, 315, 137, 314, 241, 155, 108, 233, 138, 162, 170, 210],
      dtype=int64)
```

In [158]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[136:406:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[158]:
```
array([160, 66, 34, 84, 95, 93, 101, 72, 32, 59, 57, 57],
      dtype=int64)
```

In [159]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[140:403:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[159]:
```
array([ 2, 37, 38, 6, 2, 21, 0, 18, 37, 4, 42, -2], dtype=int64)
```

In [160]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[447:694:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[160]:
```
array([46, 76, 58, 51, 61, 74, 62, 54, 67, 68, 69, 60], dtype=int64)
```

In [161]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[449:694:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[161]:
```
array([20, 29, 9, 48, 37, 13, 25, 27, 30, 19, 10, 26], dtype=int64)
```

In [162]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[450:694:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[162]:
```
array([1, 1, 3, 3, 3, 0, 2, 2, 2, 1, 2, 1], dtype=int64)
```

In [163]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[451:694:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[163]:
```
array([10, 4, 0, 27, 23, 0, 22, 15, 16, 13, 8, 11], dtype=int64)
```

In [164]:
```
#TSU punts
TSUpunt=[]
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[753][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[772][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[791][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[810][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[829][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[848][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[867][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[886][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[905][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[924][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[943][0])
TSUpunt.append(mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[962][0])
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[164]:
```
array([3, 3, 3, 3, 5, 6, 1, 1, 4, 4, 7, 5], dtype=int64)
```

In [165]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
      'attendance':attendance,
      'TSUrushyards':TSUrushyards,
      'TSUreceiveyards':TSUreceiveyards,
      'TSUkreturnyards':TSUkreturnyards,
      'TSUpreturnyards':TSUpreturnyards,
      'TSUtackles':TSUtackles,
      'TSUtackleyd':TSUtackleyd,
      'TSUsacks':TSUsacks,
      'TSUsackyd':TSUsackyd,
      'TSUpunt':TSUpunt}
df2008=pd.DataFrame(list_of_dicts)
df2008.head()
```

Out[165]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Aug 30 2008	10072	107	292	160	2	46	20	1	10
1	Sep 06 2008	28830	148	315	66	37	76	29	1	4
2	Sep 13 2008	50794	148	137	34	38	58	9	0	0
3	Sep 20 2008	8276	229	314	84	6	51	48	3	27
4	Sep 27 2008	50428	136	241	95	2	61	37	3	23

In [166]:
```
#creating date column as a datetime column
df2008['date']=pd.to_datetime(df2008['date'],format='%b %d %Y')
#creating year column
df2008['year']=df2008.date.dt.year
df2008.head()
```

Out[166]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2008-08-30	10072	107	292	160	2	46	20	1	10
1	2008-09-06	28830	148	315	66	37	76	29	1	4
2	2008-09-13	50794	148	137	34	38	58	9	0	0
3	2008-09-20	8276	229	314	84	6	51	48	3	27
4	2008-09-27	50428	136	241	95	2	61	37	3	23

In [167]:
```
#get info on data frame
df2008.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12 entries, 0 to 11
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             12 non-null     datetime64[ns]
 1   attendance       12 non-null     int64
 2   TSUrushyards     12 non-null     int64
 3   TSUreceiveyards  12 non-null     int64
 4   TSUkreturnyards  12 non-null     int64
 5   TSUpreturnyards  12 non-null     int64
 6   TSUtackles       12 non-null     int64
 7   TSUtackleyd      12 non-null     int64
 8   TSUsacks         12 non-null     int64
 9   TSUsackyd        12 non-null     int64
 10  TSUpunt          12 non-null     int64
 11  year             12 non-null     int64
dtypes: datetime64[ns](1), int64(11)
memory usage: 1.2 KB
```

In [168]:
```
#sort data
df2008=df2008.sort_values('date',ignore_index=True)
df2008.head()
```

Out[168]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2008-08-30	10072	107	292	160	2	46	20	1	10
1	2008-09-06	28830	148	315	66	37	76	29	1	4
2	2008-09-13	50794	148	137	34	38	58	9	0	0
3	2008-09-20	8276	229	314	84	6	51	48	3	27
4	2008-09-27	50428	136	241	95	2	61	37	3	23

In [169]:
```
#save data
df2008.to_csv('2008.csv',encoding='utf-8')
```

In [170]:
```
#2009 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/DEDBD687-051A-4E37-98F7-ACE

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```

```
['Sep 05 2009', 'Sep 12 2009', 'Sep 19 2009', 'Sep 26 2009', 'Oct 03 2009', 'Oct 10 2009', 'Oct 17 2009', 'Oct
31 2009', 'Nov 07 2009', 'Nov 14 2009', 'Nov 19 2009']
```

In [171]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```

Out[171]:
```
array([23871, 43306, 12247, 51950,  6314,  7100,  5572,  7999, 22092,
        6968,  3509], dtype=int64)
```

In [172]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```

Out[172]:
```
array([ 13, 217, 169, 167, 259, 159, 169, 152,  62, 104, 176], dtype=int64)
```

In [173]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```

Out[173]:
```
array([178,  43,  86, 107,  95, 124,  68, 128, 207, 189, 135], dtype=int64)
```

In [174]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```

Out[174]:
```
array([46, 19, 75, 95, 72, 81, 64, 78, 97, 60, 55], dtype=int64)
```

In [175]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:383:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```

Out[175]:
```
array([ 0, 20, 13,  2, 30, 24,  5,  7,  7, -4], dtype=int64)
```

In [176]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:644:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```

Out[176]:
```
array([71, 60, 81, 63, 58, 52, 69, 56, 83, 73, 47], dtype=int64)
```

In [177]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:646:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```

Out[177]:
```
array([18, 42, 28, 14, 20, 24, 21, 35,  9,  6, 19], dtype=int64)
```

In [178]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:646:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```

Out[178]:
```
array([1, 3, 3, 1, 3, 3, 2, 2, 0, 0, 2], dtype=int64)
```

In [179]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:646:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```

Out[179]:
```
array([ 7, 26, 15,  5, 15, 14, 12, 18,  0,  0, 11], dtype=int64)
```

In [180]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```

Out[180]:
```
array([6, 9, 8, 6, 5, 6, 7, 7, 8, 8, 6], dtype=int64)
```

In [181]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2009=pd.DataFrame(list_of_dicts)
df2009.head()
```

Out[181]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 05 2009	23871	13	178	46	0	71	18	1	7
1	Sep 12 2009	43306	217	43	19	20	60	42	3	26
2	Sep 19 2009	12247	169	86	75	13	81	28	3	15
3	Sep 26 2009	51950	167	107	95	2	63	14	1	5
4	Oct 03 2009	6314	259	95	72	30	58	20	3	15

In [182]:
```
#creating date column as a datetime column
df2009['date']=pd.to_datetime(df2009['date'],format='%b %d %Y')
#creating year column
df2009['year']=df2009.date.dt.year
df2009.head()
```

Out[182]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2009-09-05	23871	13	178	46	0	71	18	1	7
1	2009-09-12	43306	217	43	19	20	60	42	3	26
2	2009-09-19	12247	169	86	75	13	81	28	3	15
3	2009-09-26	51950	167	107	95	2	63	14	1	5
4	2009-10-03	6314	259	95	72	30	58	20	3	15

In [183]:
```
#get info on data frame
df2009.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             11 non-null     datetime64[ns]
 1   attendance       11 non-null     int64
 2   TSUrushyards     11 non-null     int64
 3   TSUreceiveyards  11 non-null     int64
 4   TSUkreturnyards  11 non-null     int64
 5   TSUpreturnyards  11 non-null     int64
 6   TSUtackles       11 non-null     int64
 7   TSUtackleyd      11 non-null     int64
 8   TSUsacks         11 non-null     int64
 9   TSUsackyd        11 non-null     int64
 10  TSUpunt          11 non-null     int64
 11  year             11 non-null     int64
dtypes: datetime64[ns](1), int64(11)
memory usage: 1.2 KB
```

In [184]:
```
#sort data
df2009=df2009.sort_values('date',ignore_index=True)
df2009.head()
```

Out[184]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2009-09-05	23871	13	178	46	0	71	18	1	7
1	2009-09-12	43306	217	43	19	20	60	42	3	26
2	2009-09-19	12247	169	86	75	13	81	28	3	15
3	2009-09-26	51950	167	107	95	2	63	14	1	5
4	2009-10-03	6314	259	95	72	30	58	20	3	15

In [185]:
```
#save data
df2009.to_csv('2009.csv',encoding='utf-8')
```

In [186]:
```
#2010 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/108733F9-E3AE-4C3D-95EE-76C0

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
date=[a.replace("2 ","02 ") for a in date]
date=[a.replace("9 ","09 ") for a in date]
print(date)
```

```
['Sep 04 2010', 'Sep 11 2010', 'Sep 18 2010', 'Sep 25 2010', 'Oct 02 2010', 'Oct 09 2010', 'Oct 16 2010', 'Oct
23 2010', 'Nov 06 2010', 'Nov 13 2010', 'Nov 20 2010']
```

In [187]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```

Out[187]:
```
array([22607, 44688,  8502, 54202, 35217, 10318, 15218,  6739, 21596,
        3141, 29061], dtype=int64)
```

In [188]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```

Out[188]:
```
array([107, 224, 162, 304, 379, 152, 154, 158, 180,  72, 198], dtype=int64)
```

In [189]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```

Out[189]:
```
array([127, 185, 225, 109, 142, 147,  65, 203, 277,  51, 289], dtype=int64)
```

In [190]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```

Out[190]:
```
array([127, 133, 134,  33,  20,  90,  61,  56,  29, 116, 134], dtype=int64)
```

In [191]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:383:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```

Out[191]:
```
array([ 36,  28,  25, 142,  13,   3, -19,   1,  21,   2,   1], dtype=int64)
```

In [192]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:644:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```

Out[192]:
```
array([64, 60, 57, 58, 66, 86, 66, 67, 64, 55, 65], dtype=int64)
```

In [193]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:646:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```

Out[193]:
```
array([42, 26, 24, 60, 34, 17, 11, 24, 19, 32, 30], dtype=int64)
```

In [194]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:646:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```

Out[194]:
```
array([7., 0., 2., 8., 5., 1., 0., 1., 1., 1., 1.])
```

In [195]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:646:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```

Out[195]:
```
array([40,  0, 13, 48, 31,  9,  0,  4,  7,  7,  8], dtype=int64)
```

In [196]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```

Out[196]:
```
array([6, 6, 3, 5, 4, 5, 8, 3, 7, 9, 5], dtype=int64)
```

In [197]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2010=pd.DataFrame(list_of_dicts)
df2010.head()
```

Out[197]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 04 2010	22607	107	127	127	36	64	42	7.0	40
1	Sep 11 2010	44688	224	185	133	28	60	26	0.0	0
2	Sep 18 2010	8502	162	225	134	25	57	24	2.0	13
3	Sep 25 2010	54202	304	109	33	142	58	60	8.0	48
4	Oct 02 2010	35217	379	142	20	13	66	34	5.0	31

In [198]:
```
#creating date column as a datetime column
df2010['date']=pd.to_datetime(df2010['date'],format='%b %d %Y')
#creating year column
df2010['year']=df2010.date.dt.year
df2010.head()
```

Out[198]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2010-09-04	22607	107	127	127	36	64	42	7.0	40
1	2010-09-11	44688	224	185	133	28	60	26	0.0	0
2	2010-09-18	8502	162	225	134	25	57	24	2.0	13
3	2010-09-25	54202	304	109	33	142	58	60	8.0	48
4	2010-10-02	35217	379	142	20	13	66	34	5.0	31

In [199]:
```
#get info for data frame
df2010.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             11 non-null     datetime64[ns]
 1   attendance       11 non-null     int64
 2   TSUrushyards     11 non-null     int64
 3   TSUreceiveyards  11 non-null     int64
 4   TSUkreturnyards  11 non-null     int64
 5   TSUpreturnyards  11 non-null     int64
 6   TSUtackles       11 non-null     int64
 7   TSUtackleyd      11 non-null     int64
 8   TSUsacks         11 non-null     float64
 9   TSUsackyd        11 non-null     int64
 10  TSUpunt          11 non-null     int64
 11  year             11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [200]:
```
#sort data
df2010=df2010.sort_values('date',ignore_index=True)
df2010.head()
```

Out[200]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2010-09-04	22607	107	127	127	36	64	42	7.0	40
1	2010-09-11	44688	224	185	133	28	60	26	0.0	0
2	2010-09-18	8502	162	225	134	25	57	24	2.0	13
3	2010-09-25	54202	304	109	33	142	58	60	8.0	48
4	2010-10-02	35217	379	142	20	13	66	34	5.0	31

In [201]:
```
#save data
df2010.to_csv('2010.csv',encoding='utf-8')
```

In [202]:
```
#2011 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/CA835442-6528-4937-854B-D96

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```

```
['Sep 03 2011', 'Sep 10 2011', 'Sep 17 2011', 'Sep 24 2011', 'Oct 01 2011', 'Oct 08 2011', 'Oct 15 2011', 'Oct
22 2011', 'Nov 05 2011', 'Nov 12 2011', 'Nov 19 2011']
```

In [203]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```

Out[203]:
```
array([25209, 43532, 10031, 33487,  8614,  6234,  8676, 10800,  6774,
       19537,  6137], dtype=int64)
```

In [204]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```

Out[204]:
```
array([342,  78, 168, 188, 162, 258, 166,  65, 200, 253,  87], dtype=int64)
```

In [205]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```

Out[205]:
```
array([170, 226, 207, 206, 304, 278, 232, 343, 118, 263, 262], dtype=int64)
```

In [206]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```

Out[206]:
```
array([ 48, 146, 197,  97, 158,  15, 118, 112,  64,  46, 129], dtype=int64)
```

In [207]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:383:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```

Out[207]:
```
array([  0,   5,  17,   0,   0, 115,   5,  17,   0,   5,  16], dtype=int64)
```

In [208]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:644:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```

Out[208]:
```
array([45, 64, 94, 88, 61, 54, 88, 80, 59, 53, 50], dtype=int64)
```

In [209]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:646:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```

Out[209]:
```
array([41, 45, 32,  0, 20, 17, 38, 20, 23, 14, 18], dtype=int64)
```

In [210]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:646:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```

Out[210]:
```
array([5., 3., 3., 0., 2., 1., 5., 3., 3., 1., 2.])
```

In [211]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:646:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```

Out[211]:
```
array([30, 31, 21,  0, 10,  9, 32, 14, 17,  1, 10], dtype=int64)
```

In [212]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```

Out[212]:
```
array([3, 6, 9, 8, 2, 2, 5, 4, 7, 5, 5], dtype=int64)
```

In [213]:
```
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2011=pd.DataFrame(list_of_dicts)
df2011.head()
```

Out[213]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	Sep 03 2011	25209	342	170	48	0	45	41	5.0	30
1	Sep 10 2011	43532	78	226	146	5	64	45	3.0	31
2	Sep 17 2011	10031	168	207	197	17	94	32	3.0	21
3	Sep 24 2011	33487	188	206	97	0	88	0	0.0	0
4	Oct 01 2011	8614	162	304	158	0	61	20	2.0	10

In [214]:
```
#creating date column as a datetime column
df2011['date']=pd.to_datetime(df2011['date'],format='%b %d %Y')
#creating year column
df2011['year']=df2011.date.dt.year
df2011.head()
```

Out[214]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2011-09-03	25209	342	170	48	0	45	41	5.0	30
1	2011-09-10	43532	78	226	146	5	64	45	3.0	31
2	2011-09-17	10031	168	207	197	17	94	32	3.0	21
3	2011-09-24	33487	188	206	97	0	88	0	0.0	0
4	2011-10-01	8614	162	304	158	0	61	20	2.0	10

In [215]:
```
#get info from data frame
df2011.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             11 non-null     datetime64[ns]
 1   attendance       11 non-null     int64
 2   TSUrushyards     11 non-null     int64
 3   TSUreceiveyards  11 non-null     int64
 4   TSUkreturnyards  11 non-null     int64
 5   TSUpreturnyards  11 non-null     int64
 6   TSUtackles       11 non-null     int64
 7   TSUtackleyd      11 non-null     int64
 8   TSUsacks         11 non-null     float64
 9   TSUsackyd        11 non-null     int64
 10  TSUpunt          11 non-null     int64
 11  year             11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [216]:
```
#sort data
df2011=df2011.sort_values('date',ignore_index=True)
df2011.head()
```

Out[216]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2011-09-03	25209	342	170	48	0	45	41	5.0	30
1	2011-09-10	43532	78	226	146	5	64	45	3.0	31
2	2011-09-17	10031	168	207	197	17	94	32	3.0	21
3	2011-09-24	33487	188	206	97	0	88	0	0.0	0
4	2011-10-01	8614	162	304	158	0	61	20	2.0	10

In [217]:
```
#save data
df2011.to_csv('2011.csv',encoding='utf-8')
```

In [218]:
```
#2012 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/BDF8C4F3-D7B6-4D50-847F-9227

#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```

```
['Sep 01 2012', 'Sep 08 2012', 'Sep 15 2012', 'Sep 22 2012', 'Sep 29 2012', 'Oct 05 2012', 'Oct 13 2012', 'Oct
20 2012', 'Oct 27 2012', 'Nov 03 2012', 'Nov 17 2012']
```

In [219]:
```
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```

Out[219]:
```
array([15652, 42257, 14264,  9461, 31765,  9878,  4800, 14867, 11373,
        3112,  6322], dtype=int64)
```

In [220]:
```
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:357:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```

Out[220]:
```
array([138, 235, 112, 200, 201, 139, 306, 184, 158, 138, 122], dtype=int64)
```

In [221]:
```
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:360:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```

Out[221]:
```
array([263, 137, 322, 157, 262, 227, 154, 311, 294, 273, 355], dtype=int64)
```

In [222]:
```
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:360:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```

Out[222]:
```
array([ 52,  69,  24,  41,  47,  63,  56,  62,  54, 195, 101], dtype=int64)
```

In [223]:
```
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:383:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```

Out[223]:
```
array([ 8, 42,  7, 37,  4,  0,  6,  0, 44,  0,  0], dtype=int64)
```

In [224]:
```
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:644:22]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```

Out[224]:
```
array([49, 66, 55, 84, 62, 67, 74, 62, 48, 72, 79], dtype=int64)
```

In [225]:
```
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:646:22]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```

Out[225]:
```
array([28, 21, 13, 52, 40, 11, 21, 15, 13, 11, 24], dtype=int64)
```

In [226]:
```
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:646:22]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```

Out[226]:
```
array([1, 3, 1, 4, 3, 3, 2, 1, 1, 1, 1], dtype=int64)
```

In [227]:
```
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:646:22]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```

Out[227]:
```
array([ 5, 20,  9, 30, 29,  6, 13,  9,  8,  7,  6], dtype=int64)
```

In [228]:
```
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[698:900:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```

In [290]:
```python
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:660:23]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[290]:
```
array([5., 3., 1., 2., 1., 0., 1., 2., 2., 1., 3.])
```

In [291]:
```python
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:660:23]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[291]:
```
array([31, 21,  1, 12,  3,  0,  5, 14, 13,  0, 26], dtype=int64)
```

In [292]:
```python
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[711:919:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[292]:
```
array([3, 6, 5, 1, 5, 3, 3, 2, 5, 3, 3], dtype=int64)
```

In [293]:
```python
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2016=pd.DataFrame(list_of_dicts)
df2016.head()
```
Out[293]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	Sep 03 2016	15078	202	259	29	64	52	39	5.0	31	
1	Sep 10 2016	46263	121	273	96	11	61	40	3.0	21	
2	Sep 17 2016	9385	210	184	122	23	52	17	1.0	1	
3	Oct 01 2016	10001	141	223	80	0	63	28	2.0	12	
4	Oct 08 2016	4319	76	303	113	0	75	6	1.0	3	

In [294]:
```python
#creating date column as a datetime column
df2016['date']=pd.to_datetime(df2016['date'],format='%b %d %Y')
#creating year column
df2016['year']=df2016.date.dt.year
df2016.head()
```
Out[294]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2016-09-03	15078	202	259	29	64	52	39	5.0	31	
1	2016-09-10	46263	121	273	96	11	61	40	3.0	21	
2	2016-09-17	9385	210	184	122	23	52	17	1.0	1	
3	2016-10-01	10001	141	223	80	0	63	28	2.0	12	
4	2016-10-08	4319	76	303	113	0	75	6	1.0	3	

In [295]:
```python
#get info on data frame
df2016.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             11 non-null     datetime64[ns]
 1   attendance       11 non-null     int64
 2   TSUrushyards     11 non-null     int64
 3   TSUreceiveyards  11 non-null     int64
 4   TSUkreturnyards  11 non-null     int64
 5   TSUpreturnyards  11 non-null     int64
 6   TSUtackles       11 non-null     int64
 7   TSUtackleyd      11 non-null     int64
 8   TSUsacks         11 non-null     float64
 9   TSUsackyd        11 non-null     int64
 10  TSUpunt          11 non-null     int64
 11  year             11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [296]:
```python
#sort data
df2016=df2016.sort_values('date',ignore_index=True)
df2016.head()
```
Out[296]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2016-09-03	15078	202	259	29	64	52	39	5.0	31	
1	2016-09-10	46263	121	273	96	11	61	40	3.0	21	
2	2016-09-17	9385	210	184	122	23	52	17	1.0	1	
3	2016-10-01	10001	141	223	80	0	63	28	2.0	12	
4	2016-10-08	4319	76	303	113	0	75	6	1.0	3	

In [297]:
```python
#save data
df2016.to_csv('2016.csv',encoding='utf-8')
```

In [298]:
```python
#2017 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/1EEBECA9-B01E-43AB-B6EB-AF5B
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:110:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
print(date)
```
```
['Aug 31 2017', 'Sep 09 2017', 'Sep 17 2017', 'Sep 23 2017', 'Sep 30 2017', 'Oct 07 2017', 'Oct 14 2017', 'Oct 28 2017', 'Nov 04 2017', 'Nov 11 2017', 'Nov 16 2017']
```

In [299]:
```python
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:110:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[299]:
```
array([24333, 47407, 17102,  6484, 11013,  8410, 21127,  5235,  7487,
        8693, 18782], dtype=int64)
```

In [300]:
```python
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[114:346:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[300]:
```
array([238, 160, 241, 100,  83, 193, 106,  83, 174, 147,  15], dtype=int64)
```

In [301]:
```python
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[118:369:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[301]:
```
array([145,  78, 273, 159, 195, 106, 208, 331, 274, 196,  82], dtype=int64)
```

In [302]:
```python
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[126:363:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[302]:
```
array([ 44,  49,  63, 160,  48,  25,  64, 135,  18, 103, 140], dtype=int64)
```

In [303]:
```python
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[130:363:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[303]:
```
array([ 0, 41,  7,  3,  0, -3, 13, -2, 94, 29,  1], dtype=int64)
```

In [304]:
```python
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[414:646:23]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[304]:
```
array([59, 76, 38, 77, 73, 60, 61, 62, 41, 57, 71], dtype=int64)
```

In [305]:
```python
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[416:660:23]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[305]:
```
array([26, 21, 19, 21, 14,  8, 11, 28, 45, 14, 18], dtype=int64)
```

In [306]:
```python
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[417:660:23]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[306]:
```
array([3., 4., 0., 1., 0., 1., 1., 1., 6., 1., 1.])
```

In [307]:
```python
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[418:660:23]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[307]:
```
array([17, 18,  0,  5,  0,  1,  0,  9, 34,  5,  9], dtype=int64)
```

In [308]:
```python
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[711:919:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[308]:
```
array([ 5,  6,  2,  6,  6,  5,  4,  4,  3,  4, 11], dtype=int64)
```

In [309]:
```python
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2017=pd.DataFrame(list_of_dicts)
df2017.head()
```
Out[309]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	Aug 31 2017	24333	238	145	44	0	59	26	3.0	17	
1	Sep 09 2017	47407	160	78	49	41	76	21	4.0	18	
2	Sep 17 2017	17102	241	273	63	7	38	19	0.0	0	
3	Sep 23 2017	6484	100	159	160	3	77	21	1.0	5	
4	Sep 30 2017	11013	83	195	48	0	73	14	0.0	0	

In [310]:
```python
#creating date column as a datetime column
df2017['date']=pd.to_datetime(df2017['date'],format='%b %d %Y')
#creating year column
df2017['year']=df2017.date.dt.year
df2017.head()
```
Out[310]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2017-08-31	24333	238	145	44	0	59	26	3.0	17	
1	2017-09-09	47407	160	78	49	41	76	21	4.0	18	
2	2017-09-17	17102	241	273	63	7	38	19	0.0	0	
3	2017-09-23	6484	100	159	160	3	77	21	1.0	5	
4	2017-09-30	11013	83	195	48	0	73	14	0.0	0	

In [311]:
```python
#get info on data frame
df2017.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11 entries, 0 to 10
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             11 non-null     datetime64[ns]
 1   attendance       11 non-null     int64
 2   TSUrushyards     11 non-null     int64
 3   TSUreceiveyards  11 non-null     int64
 4   TSUkreturnyards  11 non-null     int64
 5   TSUpreturnyards  11 non-null     int64
 6   TSUtackles       11 non-null     int64
 7   TSUtackleyd      11 non-null     int64
 8   TSUsacks         11 non-null     float64
 9   TSUsackyd        11 non-null     int64
 10  TSUpunt          11 non-null     int64
 11  year             11 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [312]:
```python
#sort data
df2017=df2017.sort_values('date',ignore_index=True)
df2017.head()
```
Out[312]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2017-08-31	24333	238	145	44	0	59	26	3.0	17	
1	2017-09-09	47407	160	78	49	41	76	21	4.0	18	
2	2017-09-17	17102	241	273	63	7	38	19	0.0	0	
3	2017-09-23	6484	100	159	160	3	77	21	1.0	5	
4	2017-09-30	11013	83	195	48	0	73	14	0.0	0	

In [313]:
```python
#save data
df2017.to_csv('2017.csv',encoding='utf-8')
```

In [314]:
```python
#2018 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/125803E8-C113-4C2F-890C-645
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:90:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
date=[a.strip() for a in date]
date=[a.replace('11-03-18' ,'Nov 03 2018') for a in date]
print(date)
```
```
['Sep 01 2018', 'Sep 22 2018', 'Sep 29 2018', 'Oct 06 2018', 'Oct 13 2018', 'Oct 20 2018', 'Nov 03 2018', 'Nov 10 2018', 'Nov 17 2018']
```

In [315]:
```python
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:90:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[315]:
```
array([14069,  7670, 27340, 12201,  3318, 17283,  3481,  6718,  3618],
      dtype=int64)
```

In [316]:
```python
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[94:300:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[316]:
```
array([201, 195, 104, 164,  63, 149, 253, 178,  89], dtype=int64)
```

In [317]:
```python
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[98:300:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[317]:
```
array([324, 349, 269, 325, 307, 323, 255, 164, 170], dtype=int64)
```

In [318]:
```python
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[106:300:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[318]:
```
array([ 49, 150, 105, 134, 140,  48,  78,  36,  63], dtype=int64)
```

In [319]:
```python
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[110:300:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[319]:
```
array([64, 51,  3, 34, -1, 14,  0,  0, 29], dtype=int64)
```

In [320]:
```python
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[348:546:23]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[320]:
```
array([54, 63, 69, 77, 74, 77, 65, 60, 64], dtype=int64)
```

In [321]:
```python
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[350:546:23]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[321]:
```
array([43, 30,  5, 19, 19, 28, 28, 27, 38], dtype=int64)
```

In [322]:
```python
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[351:540:23]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[322]:
```
array([5., 2., 0., 0., 1., 4., 2., 2., 2.])
```

In [323]:
```python
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[352:540:23]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[323]:
```
array([35, 15,  0,  0,  7, 21, 14, 19, 18], dtype=int64)
```

In [324]:
```python
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[599:769:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[324]:
```
array([6, 4, 4, 4, 5, 4, 3, 5, 7], dtype=int64)
```

In [325]:
```python
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2018=pd.DataFrame(list_of_dicts)
df2018.head()
```
Out[325]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	Sep 01 2018	14069	201	324	49	64	54	43	5.0	35	
1	Sep 22 2018	7670	195	349	150	51	63	30	2.0	15	
2	Sep 29 2018	27340	104	269	105	3	69	5	0.0	0	
3	Oct 06 2018	12201	164	325	134	34	77	19	0.0	0	
4	Oct 13 2018	3318	63	307	140	-1	74	19	1.0	7	

In [326]:
```python
#creating date column as a datetime column
df2018['date']=pd.to_datetime(df2018['date'],format='%b %d %Y')
#creating year column
df2018['year']=df2018.date.dt.year
df2018.head()
```
Out[326]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2018-09-01	14069	201	324	49	64	54	43	5.0	35	
1	2018-09-22	7670	195	349	150	51	63	30	2.0	15	
2	2018-09-29	27340	104	269	105	3	69	5	0.0	0	
3	2018-10-06	12201	164	325	134	34	77	19	0.0	0	
4	2018-10-13	3318	63	307	140	-1	74	19	1.0	7	

In [327]:
```python
#get info from data frame
df2018.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9 entries, 0 to 8
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             9 non-null      datetime64[ns]
 1   attendance       9 non-null      int64
 2   TSUrushyards     9 non-null      int64
 3   TSUreceiveyards  9 non-null      int64
 4   TSUkreturnyards  9 non-null      int64
 5   TSUpreturnyards  9 non-null      int64
 6   TSUtackles       9 non-null      int64
 7   TSUtackleyd      9 non-null      int64
 8   TSUsacks         9 non-null      float64
 9   TSUsackyd        9 non-null      int64
 10  TSUpunt          9 non-null      int64
 11  year             9 non-null      int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 992.0 bytes
```

In [328]:
```python
#sort values
df2018=df2018.sort_values('date',ignore_index=True)
df2018.head()
```
Out[328]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2018-09-01	14069	201	324	49	64	54	43	5.0	35	
1	2018-09-22	7670	195	349	150	51	63	30	2.0	15	
2	2018-09-29	27340	104	269	105	3	69	5	0.0	0	
3	2018-10-06	12201	164	325	134	34	77	19	0.0	0	
4	2018-10-13	3318	63	307	140	-1	74	19	1.0	7	

In [329]:
```python
#save data
df2018.to_csv('2018.csv',encoding='utf-8')
```

In [330]:
```python
#2019 data
#use requests.get() to get web page with data
page = requests.get('https://tennstate_ftp.sidearmsports.com/custompages/tsutigers/E018AECE-A1B8-46E7-B1B7-46FE
#parse data on web page using html module.fromstring
mytree = html.fromstring(page.content)

#go to web address above , right click on page and select inspect to get HTML code for data from right side of
#create XPath query and use xpath function to get data
#date data
date = mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[1:120:10]
date=[a.replace("\xa0","") for a in date]
date=[a.replace(",","") for a in date]
date=[a.replace(".","") for a in date]
date=[a.replace("7 ","07 ") for a in date]
date=[a.strip() for a in date]
print(date)
```
```
['Aug 31 2019', 'Sep 07 2019', 'Sep 14 2019', 'Sep 21 2019', 'Sep 28 2019', 'Oct 05 2019', 'Oct 12 2019', 'Oct 19 2019', 'Nov 02 2019', 'Nov 09 2019', 'Nov 16 2019', 'Nov 23 2019']
```

In [331]:
```python
#get attendance data
attendance=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[9:120:10]
attendance=[a.replace("\xa0","") for a in attendance]
attendance=[a.strip() for a in attendance]
attendance=pd.to_numeric(attendance)
attendance
```
Out[331]:
```
array([13458, 20912, 48347,  8683,  8861, 16589,  5324, 16389,  4738,
        4131,  1776,  2728], dtype=int64)
```

In [332]:
```python
#get TSU rushing yards
TSUrushyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[124:383:23]
TSUrushyards=[a.replace("\xa0","") for a in TSUrushyards]
TSUrushyards=pd.to_numeric(TSUrushyards)
TSUrushyards
```
Out[332]:
```
array([117, 207, 141,  71, 112, 230, 108, 116,  83,  97, 193, 198],
      dtype=int64)
```

In [333]:
```python
#get TSU receiving yards
TSUreceiveyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[128:383:23]
TSUreceiveyards=[a.replace("\xa0","") for a in TSUreceiveyards]
TSUreceiveyards=pd.to_numeric(TSUreceiveyards)
TSUreceiveyards
```
Out[333]:
```
array([330, 259, 389, 334, 270, 240, 312, 220, 217, 209, 123, 225],
      dtype=int64)
```

In [334]:
```python
#get TSU kick return yards
TSUkreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[136:400:23]
TSUkreturnyards=[a.replace("\xa0","") for a in TSUkreturnyards]
TSUkreturnyards=pd.to_numeric(TSUkreturnyards)
TSUkreturnyards
```
Out[334]:
```
array([ 13, 108, 155,  74,  38,  56,  55,  66, 113, 162, 105,  51],
      dtype=int64)
```

In [335]:
```python
#get TSU punt return yards
TSUpreturnyards=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[140:400:23]
TSUpreturnyards=[a.replace("\xa0","") for a in TSUpreturnyards]
TSUpreturnyards=pd.to_numeric(TSUpreturnyards)
TSUpreturnyards
```
Out[335]:
```
array([61,  0,  0, 15,  7,  0,  8,  0, 13, 62,  6, 25], dtype=int64)
```

In [336]:
```python
#get TSU total tackles
TSUtackles=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[447:723:23]
TSUtackles=[a.replace("\xa0","") for a in TSUtackles]
TSUtackles=pd.to_numeric(TSUtackles)
TSUtackles
```
Out[336]:
```
array([58, 49, 55, 65, 74, 75, 67, 60, 62, 63, 67, 79], dtype=int64)
```

In [337]:
```python
#get TSU tackle yards
TSUtackleyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[449:723:23]
TSUtackleyd=[a.replace("\xa0","") for a in TSUtackleyd]
TSUtackleyd=pd.to_numeric(TSUtackleyd)
TSUtackleyd
```
Out[337]:
```
array([ 3, 13, 19, 13, 11, 13, 21, 11, 17, 46, 26, 22], dtype=int64)
```

In [338]:
```python
#get TSU sacks
TSUsacks=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[450:723:23]
TSUsacks=[a.replace("\xa0","") for a in TSUsacks]
TSUsacks=pd.to_numeric(TSUsacks)
TSUsacks
```
Out[338]:
```
array([0., 1., 1., 1., 1., 1., 1., 1., 2., 2., 2., 2.])
```

In [339]:
```python
#get TSU sack yards
TSUsackyd=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[451:723:23]
TSUsackyd=[a.replace("\xa0","") for a in TSUsackyd]
TSUsackyd=pd.to_numeric(TSUsackyd)
TSUsackyd
```
Out[339]:
```
array([ 0,  7,  8,  3,  5,  7,  7,  0,  8, 21, 12, 14], dtype=int64)
```

In [340]:
```python
#TSU punts
TSUpunt=mytree.xpath('body/center//tr/td/font[@color="#000000"]/text()')[767:980:19]
TSUpunt=[a.replace("\xa0","") for a in TSUpunt]
TSUpunt=pd.to_numeric(TSUpunt)
TSUpunt
```
Out[340]:
```
array([6, 5, 3, 6, 5, 3, 3, 6, 4, 3, 6, 4], dtype=int64)
```

In [341]:
```python
#create data frame
#change dictionary of lists to data frame
list_of_dicts={'date':date,
     'attendance':attendance,
     'TSUrushyards':TSUrushyards,
     'TSUreceiveyards':TSUreceiveyards,
     'TSUkreturnyards':TSUkreturnyards,
     'TSUpreturnyards':TSUpreturnyards,
     'TSUtackles':TSUtackles,
     'TSUtackleyd':TSUtackleyd,
     'TSUsacks':TSUsacks,
     'TSUsackyd':TSUsackyd,
     'TSUpunt':TSUpunt}
df2019=pd.DataFrame(list_of_dicts)
df2019.head()
```
Out[341]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	Aug 31 2019	13458	117	330	13	61	58	3	0.0	0	
1	Sep 07 2019	20912	207	259	108	0	49	13	1.0	7	
2	Sep 14 2019	48347	141	389	155	0	55	19	1.0	8	
3	Sep 21 2019	8683	71	334	74	15	65	13	1.0	3	
4	Sep 28 2019	8861	112	270	38	7	74	11	1.0	5	

In [342]:
```python
#creating date column as a datetime column
df2019['date']=pd.to_datetime(df2019['date'],format='%b %d %Y')
#creating year column
df2019['year']=df2019.date.dt.year
df2019.head()
```
Out[342]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2019-08-31	13458	117	330	13	61	58	3	0.0	0	
1	2019-09-07	20912	207	259	108	0	49	13	1.0	7	
2	2019-09-14	48347	141	389	155	0	55	19	1.0	8	
3	2019-09-21	8683	71	334	74	15	65	13	1.0	3	
4	2019-09-28	8861	112	270	38	7	74	11	1.0	5	

In [343]:
```python
#get info for data frame
df2019.info()
```
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12 entries, 0 to 11
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             12 non-null     datetime64[ns]
 1   attendance       12 non-null     int64
 2   TSUrushyards     12 non-null     int64
 3   TSUreceiveyards  12 non-null     int64
 4   TSUkreturnyards  12 non-null     int64
 5   TSUpreturnyards  12 non-null     int64
 6   TSUtackles       12 non-null     int64
 7   TSUtackleyd      12 non-null     int64
 8   TSUsacks         12 non-null     float64
 9   TSUsackyd        12 non-null     int64
 10  TSUpunt          12 non-null     int64
 11  year             12 non-null     int64
dtypes: datetime64[ns](1), float64(1), int64(10)
memory usage: 1.2 KB
```

In [344]:
```python
#sort data
df2019=df2019.sort_values('date',ignore_index=True)
df2019.head()
```
Out[344]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2019-08-31	13458	117	330	13	61	58	3	0.0	0	
1	2019-09-07	20912	207	259	108	0	49	13	1.0	7	
2	2019-09-14	48347	141	389	155	0	55	19	1.0	8	
3	2019-09-21	8683	71	334	74	15	65	13	1.0	3	
4	2019-09-28	8861	112	270	38	7	74	11	1.0	5	

In [345]:
```python
#save data
df2019.to_csv('2019.csv',encoding='utf-8')
```

Merge data frames with TSU scores and annual stats

In [346]:
```python
#read in annual stats datafiles
#2003
df2003=pd.read_csv('2003.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2003=df2003.iloc[:,1:]
df2003.head()
```
Out[346]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2003-08-30	18124	113	365	69	88	69	33	3.0	21	
1	2003-09-06	18085	141	146	147	8	68	22	0.0	0	
2	2003-09-13	52603	209	132	138	107	84	26	1.0	4	
3	2003-09-20	70185	153	138	86	2	50	22	0.0	0	
4	2003-09-27	8434	142	285	11	29	76	46	4.0	22	

In [347]:
```python
#2004
df2004=pd.read_csv('2004.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2004=df2004.iloc[:,1:]
df2004.head()
```
Out[347]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd	TSUp
0	2004-09-04	25117	152	203	24	26	74	35	2.0	12	
1	2004-09-09	7019	156	158	34	32	65	60	4.0	30	
2	2004-09-18	55015	164	149	39	10	63	19	2.0	7	
3	2004-09-25	67712	161	163	49	32	78	70	5.0	49	
4	2004-10-02	51082	22	173	57	0	91	17	1.0	4	

In [348]:
```python
#2005
df2005=pd.read_csv('2005.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2005=df2005.iloc[:,1:]
df2005.head()
```

Out[348]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2005-09-03	25342	72	154	62	32	47	20	2.0	13
1	2005-09-10	48300	286	103	11	47	59	29	3.0	16
2	2005-09-17	5263	77	170	110	-2	95	17	1.0	6
3	2005-09-24	56297	26	238	10	67	70	56	7.0	50
4	2005-10-01	42310	117	115	99	49	75	56	6.0	37

In [349]:
```
#2006
df2006=pd.read_csv('2006.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2006=df2006.iloc[:,1:]
df2006.head()
```

Out[349]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2006-09-02	19487	156	206	105	0	63	3	0.0	0
1	2006-09-09	10613	111	230	102	0	76	38	3.0	23
2	2006-09-16	53441	146	167	77	24	63	18	1.0	6
3	2006-09-23	27460	71	181	70	0	59	25	1.0	7
4	2006-09-30	57885	227	210	98	16	54	14	1.0	4

In [350]:
```
#2007
df2007=pd.read_csv('2007.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2007=df2007.iloc[:,1:]
df2007.head()
```

Out[350]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2007-09-01	23440	206	176	154	35	56	17	2	15
1	2007-09-08	50879	153	232	84	24	63	30	2	16
2	2007-09-15	8359	238	233	83	14	64	44	3	26
3	2007-09-22	15371	133	316	135	15	80	56	3	36
4	2007-09-29	56990	177	309	111	33	78	28	3	23

In [351]:
```
#2008
df2008=pd.read_csv('2008.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2008=df2008.iloc[:,1:]
df2008.head()
```

Out[351]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2008-08-30	10072	107	292	160	2	46	20	1	10
1	2008-09-06	28830	148	315	66	37	76	29	1	4
2	2008-09-13	50794	148	137	34	38	58	9	0	0
3	2008-09-20	8276	229	314	84	6	51	48	3	27
4	2008-09-27	50428	136	241	95	2	61	37	3	23

In [352]:
```
#2009
df2009=pd.read_csv('2009.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2009=df2009.iloc[:,1:]
df2009.head()
```

Out[352]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2009-09-05	23871	13	178	46	0	71	18	1	7
1	2009-09-12	43306	217	43	19	20	60	42	3	26
2	2009-09-19	12247	169	86	75	13	81	28	3	15
3	2009-09-26	51950	167	107	95	2	63	14	1	5
4	2009-10-03	6314	259	95	72	30	58	20	3	15

In [353]:
```
#2010
df2010=pd.read_csv('2010.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2010=df2010.iloc[:,1:]
df2010.head()
```

Out[353]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2010-09-04	22607	107	127	127	36	64	42	7.0	40
1	2010-09-11	44688	224	185	133	28	60	26	0.0	0
2	2010-09-18	8502	162	225	134	25	57	24	2.0	13
3	2010-09-25	54202	304	109	33	142	58	60	8.0	48
4	2010-10-02	35217	379	142	20	13	66	34	5.0	31

In [354]:
```
#2011
df2011=pd.read_csv('2011.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2011=df2011.iloc[:,1:]
df2011.head()
```

Out[354]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2011-09-03	25209	342	170	48	0	45	41	5.0	30
1	2011-09-10	43532	78	226	146	5	64	45	3.0	31
2	2011-09-17	10031	168	207	197	17	94	32	3.0	21
3	2011-09-24	33487	188	206	97	0	88	0	0.0	0
4	2011-10-01	8614	162	304	158	0	61	20	2.0	10

In [355]:
```
#2012
df2012=pd.read_csv('2012.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2012=df2012.iloc[:,1:]
df2012.head()
```

Out[355]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2012-09-01	15652	138	263	52	8	49	28	1	5
1	2012-09-08	42257	235	137	69	42	66	21	3	20
2	2012-09-15	14264	112	322	24	7	55	13	1	9
3	2012-09-22	9461	200	157	41	37	84	52	4	30
4	2012-09-29	31765	201	262	47	4	62	40	3	29

In [356]:
```
#2013
df2013=pd.read_csv('2013.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2013=df2013.iloc[:,1:]
df2013.head()
```

Out[356]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2013-09-01	16108	116	132	84	51	62	7	1.0	4
1	2013-09-07	14237	268	131	111	17	41	35	3.0	18
2	2013-09-14	42400	174	111	64	11	69	26	2.0	13
3	2013-09-21	10044	95	343	63	23	76	51	3.0	28
4	2013-09-28	22000	311	228	46	75	70	44	3.0	17

In [357]:
```
#2014
df2014=pd.read_csv('2014.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2014=df2014.iloc[:,1:]
df2014.head()
```

Out[357]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2014-08-30	10541	439	71	77	57	53	38	2.0	13
1	2014-09-06	15725	92	340	111	22	89	28	3.0	13
2	2014-09-13	46914	137	187	29	40	54	40	7.0	33
3	2014-09-20	9217	92	113	0	15	48	31	6.0	28
4	2014-09-27	29225	125	153	24	56	64	51	7.0	44

In [358]:
```
#2015
df2015=pd.read_csv('2015.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2015=df2015.iloc[:,1:]
df2015.head()
```

Out[358]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2015-09-06	22455	142	188	58	38	47	21	1.0	12
1	2015-09-12	48385	110	304	152	48	83	38	5.0	31
2	2015-09-19	23413	24	184	102	0	83	4	0.0	0
3	2015-09-26	18020	169	238	46	15	68	44	4.0	22
4	2015-10-10	7123	85	205	40	4	71	7	0.0	0

In [359]:
```
#2016
df2016=pd.read_csv('2016.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2016=df2016.iloc[:,1:]
df2016.head()
```

Out[359]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2016-09-03	15078	202	259	29	64	52	39	5.0	31
1	2016-09-10	46263	121	273	96	11	61	40	3.0	21
2	2016-09-17	9385	210	184	122	23	52	17	1.0	1
3	2016-10-01	10001	141	223	80	0	63	28	2.0	12
4	2016-10-08	4319	76	303	113	0	75	6	1.0	3

In [360]:
```
#2017
df2017=pd.read_csv('2017.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2017=df2017.iloc[:,1:]
df2017.head()
```

Out[360]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2017-08-31	24333	238	145	44	0	59	26	3.0	17
1	2017-09-09	47407	160	78	49	41	76	21	4.0	18
2	2017-09-17	17102	241	273	63	7	38	19	0.0	0
3	2017-09-23	6484	100	159	160	3	77	21	1.0	5
4	2017-09-30	11013	83	195	48	0	73	14	0.0	0

In [361]:
```
#2018
df2018=pd.read_csv('2018.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2018=df2018.iloc[:,1:]
df2018.head()
```

Out[361]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2018-09-01	14069	201	324	49	64	54	43	5.0	35
1	2018-09-22	7670	195	349	150	51	63	30	2.0	15
2	2018-09-29	27340	104	269	105	3	69	5	0.0	0
3	2018-10-06	12201	164	325	134	34	77	19	0.0	0
4	2018-10-13	3318	63	307	140	-1	74	19	1.0	7

In [362]:
```
#2019
df2019=pd.read_csv('2019.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
df2019=df2019.iloc[:,1:]
df2019.head()
```

Out[362]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2019-08-31	13458	117	330	13	61	58	3	0.0	0
1	2019-09-07	20912	207	259	108	0	49	13	1.0	7
2	2019-09-14	48347	141	389	155	0	55	19	1.0	8
3	2019-09-21	8683	71	334	74	15	65	13	1.0	3
4	2019-09-28	8861	112	270	38	7	74	11	1.0	5

In [363]:
```
#append all annual stats dataframes into a single data frame (stacking rows)
fullstats=df2003.append([df2004,df2005,df2006,df2007,df2008,df2009,df2010,df2011,
                         df2012,df2013,df2014,df2015,df2016,df2017,df2018,df2019],ignore_index=True)
#sort full data frame by date
fullstats=fullstats.sort_values('date')
fullstats.head()
```

Out[363]:

	date	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards	TSUpreturnyards	TSUtackles	TSUtackleyd	TSUsacks	TSUsackyd
0	2003-08-30	18124	113	365	69	88	69	33	3.0	21
1	2003-09-06	18085	141	146	147	8	68	22	0.0	0
2	2003-09-13	52603	209	132	138	107	84	26	1.0	4
3	2003-09-20	70185	153	138	86	2	50	22	0.0	0
4	2003-09-27	8434	142	285	11	29	76	46	4.0	22

In [364]:
```
#read in scores data
scores=pd.read_csv('scores.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
scores=scores.iloc[:,1:]
scores.head()
```

Out[364]:

	date	city	state	winscore	losscore	loser	winner	locale	TSU score	opponent score	scorediff	scorediff_abs	winloss	year
0	2003-08-30	Nashville	TN	37	20	South Carolina State University	Tennessee State University	Home	37	20	17	17	Win	2003
1	2003-09-06	Huntsville	AL	31	24	Tennessee State University	Alabama A&M University	Away	24	31	-7	7	Loss	2003
2	2003-09-13	Memphis	TN	44	14	Jackson State University	Tennessee State University	Away	44	14	30	30	Win	2003
3	2003-09-20	Atlanta	GA	10	7	Tennessee State University	Florida A&M University	Away	7	10	-3	3	Loss	2003
4	2003-09-27	Nashville	TN	41	10	University of Tennessee Martin	Tennessee State University	Home	41	10	31	31	Win	2003

In [365]:
```
#merge scores and fullstats data frames
#year variable is in both data frames so using suffixes to distinguish which variable came from which data frame
mydata=scores.merge(fullstats,on='date', suffixes=('_scores','_fullstats'))
mydata
```

Out[365]:

	date	city	state	winscore	losscore	loser	winner	locale	TSU score	opponent score	...	TSUrushyards	TSUreceiveyards
0	2003-08-30	Nashville	TN	37	20	South Carolina State University	Tennessee State University	Home	37	20	...	113	365
1	2003-09-06	Huntsville	AL	31	24	Tennessee State University	Alabama A&M University	Away	24	31	...	141	146
2	2003-09-13	Memphis	TN	44	14	Jackson State University	Tennessee State University	Away	44	14	...	209	132
3	2003-09-20	Atlanta	GA	10	7	Tennessee State University	Florida A&M University	Away	7	10	...	153	138
4	2003-09-27	Nashville	TN	41	10	University of Tennessee Martin	Tennessee State University	Home	41	10	...	142	285
...	...	...	...	...	...	...	...	...	...	...	...	...	...
186	2019-10-19	Nashville	TN	26	24	Austin Peay State University	Tennessee State University	Home	26	24	...	116	220
187	2019-11-02	Nashville	TN	32	13	Tennessee State University	Southeast Missouri State University	Home	13	32	...	83	217
188	2019-11-09	Nashville	TN	49	38	Tennessee State University	Eastern Illinois University	Home	38	49	...	97	209
189	2019-11-16	Martin	TN	28	17	Tennessee State University	University of Tennessee Martin	Away	17	28	...	193	123
190	2019-11-23	Cookeville	TN	37	27	Tennessee Tech University	Tennessee State University	Away	37	27	...	198	225

191 rows × 25 columns

In [366]:
```
#get information on full data frame
mydata.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 191 entries, 0 to 190
Data columns (total 25 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   date             191 non-null    object
 1   city             191 non-null    object
 2   state            191 non-null    object
 3   winscore         191 non-null    int64
 4   losscore         191 non-null    int64
 5   loser            191 non-null    object
 6   winner           191 non-null    object
 7   locale           191 non-null    object
 8   TSU score        191 non-null    int64
 9   opponent score   191 non-null    int64
 10  scorediff        191 non-null    int64
 11  scorediff_abs    191 non-null    int64
 12  winloss          191 non-null    object
 13  year_scores      191 non-null    int64
 14  attendance       191 non-null    int64
 15  TSUrushyards     191 non-null    int64
 16  TSUreceiveyards  191 non-null    int64
 17  TSUkreturnyards  191 non-null    int64
 18  TSUpreturnyards  191 non-null    int64
 19  TSUtackles       191 non-null    int64
 20  TSUtackleyd      191 non-null    int64
 21  TSUsacks         191 non-null    float64
 22  TSUsackyd        191 non-null    int64
 23  TSUpunt          191 non-null    int64
 24  year_fullstats   191 non-null    int64
dtypes: float64(1), int64(17), object(7)
memory usage: 38.8+ KB
```

More Data Wrangling-creating variables for opponent and game type (conference/nonconference), dropping one of the year variables, and making date a datetime variable

In [367]:
```
#create opponent column
mydata['opponent']=np.where(mydata['loser']!='Tennessee State University',mydata['loser'],mydata['winner'])
#frequency of opponenets-
#TSU played Eastern Illinois University more than any other school (17 times). Followed by Jackson State Univer
#University of Tennessee Martin, and Tennessee Tech University (16 times).
mydata['opponent'].value_counts()
```

Out[367]:
```
Eastern Illinois University            17
Jackson State University               16
University of Tennessee Martin         16
Tennessee Tech University              16
Southeast Missouri State University    15
Eastern Kentucky University            15
Murray State University                15
Jacksonville State University          14
Austin Peay State University           13
Florida A&M University                 13
Alabama A&M University                  8
Samford University                      5
Bethune Cookman University              4
Southern University & A&M College       4
Vanderbilt University                   3
University of Arkansas Pine Bluff       3
Alabama State University                2
South Carolina State University         2
North Carolina A&T State University     2
Air Force Academy                       1
Central State University                1
Butler University                       1
Edward Waters College                   1
Georgia State University                1
Virginia University Lynchburg           1
Mississippi Valley State University     1
Middle Tennessee State University       1
Name: opponent, dtype: int64
```

In [368]:
```
#create variable for conference/nonconference games
mydata['gametype']=np.where(mydata['opponent'].isin(["Austin Peay State University","Southeast Missouri State U
                                                     "Tennessee Tech University","Eastern Illinois University",
                                                      "Belmont University","Eastern Kentucky University",
                                                       "Jacksonville State University", "Morehead State Universit
                                                     "Murray State University", "Southern Illinois University Ev
                                                     "University of Tennessee Martin"]),
                                 "conference", "nonconference")
#frequency of game type
#TSU played 51 more conference games than nonconference games between 2003 and 2019. About two-thirds of TSU ga
#conference games(not shown).
mydata['gametype'].value_counts()
```

Out[368]:
```
conference       121
nonconference     70
Name: gametype, dtype: int64
```

In [369]:
```
#drop one of the year variables from the merge
mydata['year']=mydata['year_scores']
mydata=mydata.drop(['year_scores','year_fullstats'], axis=1)
mydata.columns
```

Out[369]:
```
Index(['date', 'city', 'state', 'winscore', 'losscore', 'loser', 'winner',
       'locale', 'TSU score', 'opponent score', 'scorediff', 'scorediff_abs',
       'winloss', 'attendance', 'TSUrushyards', 'TSUreceiveyards',
       'TSUkreturnyards', 'TSUpreturnyards', 'TSUtackles', 'TSUtackleyd',
       'TSUsacks', 'TSUsackyd', 'TSUpunt', 'opponent', 'gametype', 'year'],
      dtype='object')
```

In [370]:
```
#check data types for variables
mydata.dtypes
```

Out[370]:
```
date               object
city               object
state              object
winscore            int64
losscore            int64
loser              object
winner             object
locale             object
TSU score           int64
opponent score      int64
scorediff           int64
scorediff_abs       int64
winloss            object
attendance          int64
TSUrushyards        int64
TSUreceiveyards     int64
TSUkreturnyards     int64
TSUpreturnyards     int64
TSUtackles          int64
TSUtackleyd         int64
TSUsacks          float64
TSUsackyd           int64
TSUpunt             int64
opponent           object
gametype           object
year                int64
dtype: object
```

In [371]:
```
#make date a datetime variable
mydata['date']=pd.to_datetime(mydata['date'],format='%Y-%m-%d')
#check data types
mydata.dtypes
```

Out[371]:
```
date               datetime64[ns]
city                       object
state                      object
winscore                    int64
losscore                    int64
loser                      object
winner                     object
locale                     object
TSU score                   int64
opponent score              int64
scorediff                   int64
scorediff_abs               int64
winloss                    object
attendance                  int64
TSUrushyards                int64
TSUreceiveyards             int64
TSUkreturnyards             int64
TSUpreturnyards             int64
TSUtackles                  int64
TSUtackleyd                 int64
TSUsacks                  float64
TSUsackyd                   int64
TSUpunt                     int64
opponent                   object
gametype                   object
year                        int64
dtype: object
```

In [372]:
```
#save data with additional variables
mydata.to_csv("mydata.csv", encoding="utf-8")
```

Exploratory Data Analysis

In [373]:
```
#read in full dataset
mydata=pd.read_csv('mydata.csv')
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
mydata=mydata.iloc[:,1:]
mydata.head()
```

Out[373]:

	date	city	state	winscore	losscore	loser	winner	locale	TSU score	opponent score	...	TSUkreturnyards	TSUpreturnyards
0	2003-08-30	Nashville	TN	37	20	South Carolina State University	Tennessee State University	Home	37	20	...	69	88
1	2003-09-06	Huntsville	AL	31	24	Tennessee State University	Alabama A&M University	Away	24	31	...	147	8
2	2003-09-13	Memphis	TN	44	14	Jackson State University	Tennessee State University	Away	44	14	...	138	107
3	2003-09-20	Atlanta	GA	10	7	Tennessee State University	Florida A&M University	Away	7	10	...	86	2
4	2003-09-27	Nashville	TN	41	10	University of Tennessee Martin	Tennessee State University	Home	41	10	...	11	29

5 rows × 26 columns

In [374]:
```
#get data types
mydata.dtypes
```

Out[374]:
```
date               object
city               object
state              object
winscore            int64
losscore            int64
loser              object
winner             object
locale             object
TSU score           int64
opponent score      int64
scorediff           int64
scorediff_abs       int64
winloss            object
attendance          int64
TSUrushyards        int64
TSUreceiveyards     int64
TSUkreturnyards     int64
TSUpreturnyards     int64
TSUtackles          int64
TSUtackleyd         int64
TSUsacks          float64
TSUsackyd           int64
TSUpunt             int64
opponent           object
gametype           object
year                int64
dtype: object
```

In [375]:
```
#get descriptive statistics for numeric columns in data frame
#There were 191 rows in the data frame and each row provided data on all 17 numeric columns in the data.
#The winscore column ranged from 9 to 73 with a mean of about 33 and a standard deviation of about 11.
#The lossscore column ranged from 0 to 44 with a mean of about 27 and a standard deviation of about 13.
#The TSU score column ranged from 0 to 73 with a mean of about 26 and a standard deviation of about 13.
#The opponent score column ranged from 0 to 63 with a mean of about 76 and a standard deviation of about 13.
#The scorediff column ranged from -49 to 67 with a mean of about -0.2 and a standard deviation of about 19.
#The scorediff_abs column ranged from 1 to 67 with a mean of about 15 and a standard deviation of about 12.
#The attendance column ranged from 1,776 to 70,185 with a mean of about 16,985 and a standard deviation of abou
#The TSUrushyards column ranged from -18 to 439 with a mean of about 158 and a standard deviation of about 75.
#The TSUreceiveyards column ranged from 43 to 463 with a mean of about 210 and a standard deviation of about 82
#The TSUkreturnyards column ranged from -9 to 222 with a mean of about 79 and a standard deviation of about 44.
#The TSUpreturnyards column ranged from -19 to 142 with a mean of about 19 and a standard deviation of about 25
#The TSUtackles column ranged from 37 to 95 with a mean of about 67 and a standard deviation of about 12.
#The TSUtackleyd column ranged from 0 to 70 with a mean of about 25 and a standard deviation of about 14.
#The TSUsacks column ranged from 0 to 8 with a mean of about 2 and a standard deviation of about 2.
#The TSUsackyd column ranged from 0 to 55 with a mean of about 14 and a standard deviation of about 12.
#The TSUpunt column ranged from 0 to 11 with a mean of about 5 and a standard deviation of about 2.
mydata.describe()
```

Out[375]:

	winscore	losscore	TSU score	opponent score	scorediff	scorediff_abs	attendance	TSUrushyards	TSUreceiveyards	TSUkreturnyards
count	191.000000	191.000000	191.000000	191.00000	191.000000	191.000000	191.000000	191.000000	191.000000	191.000000
mean	32.984293	18.481675	25.643979	25.82199	-0.178010	14.502618	16985.230366	157.769634	210.361257	78.670157
std	10.659553	10.215490	12.489530	12.94969	19.109373	12.400286	15493.766885	74.855646	82.371433	44.059662
min	9.000000	0.000000	0.000000	0.00000	-49.000000	1.000000	1776.000000	-18.000000	43.000000	-9.000000
25%	26.500000	12.000000	17.000000	15.00000	-11.000000	5.000000	6402.500000	107.000000	151.000000	48.000000
50%	31.000000	17.000000	26.000000	27.00000	-3.000000	11.000000	10001.000000	152.000000	203.000000	72.000000
75%	40.000000	26.000000	34.000000	33.50000	10.000000	21.500000	22306.000000	200.000000	273.000000	110.500000
max	73.000000	44.000000	73.000000	63.00000	67.000000	67.000000	70185.000000	439.000000	463.000000	222.000000

In [376]:
```
#find NAs (There were none)
mydata.isna().sum()
```

Out[376]:
```
date               0
city               0
state              0
winscore           0
losscore           0
loser              0
winner             0
locale             0
TSU score          0
opponent score     0
scorediff          0
scorediff_abs      0
winloss            0
attendance         0
TSUrushyards       0
TSUreceiveyards    0
TSUkreturnyards    0
TSUpreturnyards    0
TSUtackles         0
TSUtackleyd        0
TSUsacks           0
TSUsackyd          0
TSUpunt            0
opponent           0
gametype           0
year               0
dtype: int64
```

In [377]:
```
#find nulls(There were none.)
mydata.isnull().sum()
```

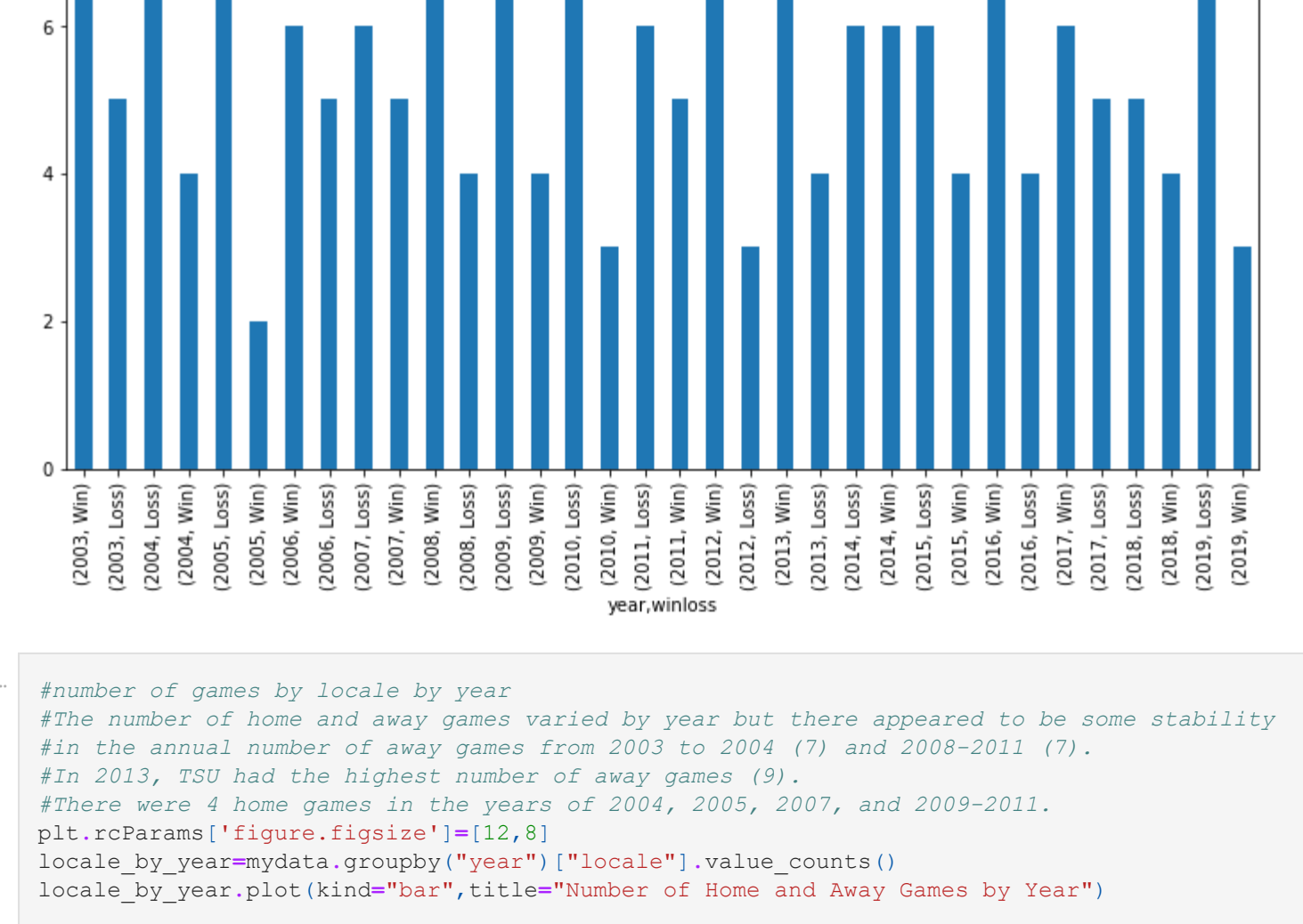
Out[377]:
```
date               0
city               0
state              0
winscore           0
losscore           0
loser              0
winner             0
locale             0
TSU score          0
opponent score     0
scorediff          0
scorediff_abs      0
winloss            0
attendance         0
TSUrushyards       0
TSUreceiveyards    0
TSUkreturnyards    0
TSUpreturnyards    0
TSUtackles         0
TSUtackleyd        0
TSUsacks           0
TSUsackyd          0
TSUpunt            0
opponent           0
gametype           0
year               0
dtype: int64
```

In [378]:
```
#get number of wins and losses
#Across the 2003-2019 time span, TSU had a record of 92-99.
mydata.winloss.value_counts()
```

Out[378]:
```
Loss    99
Win     92
Name: winloss, dtype: int64
```

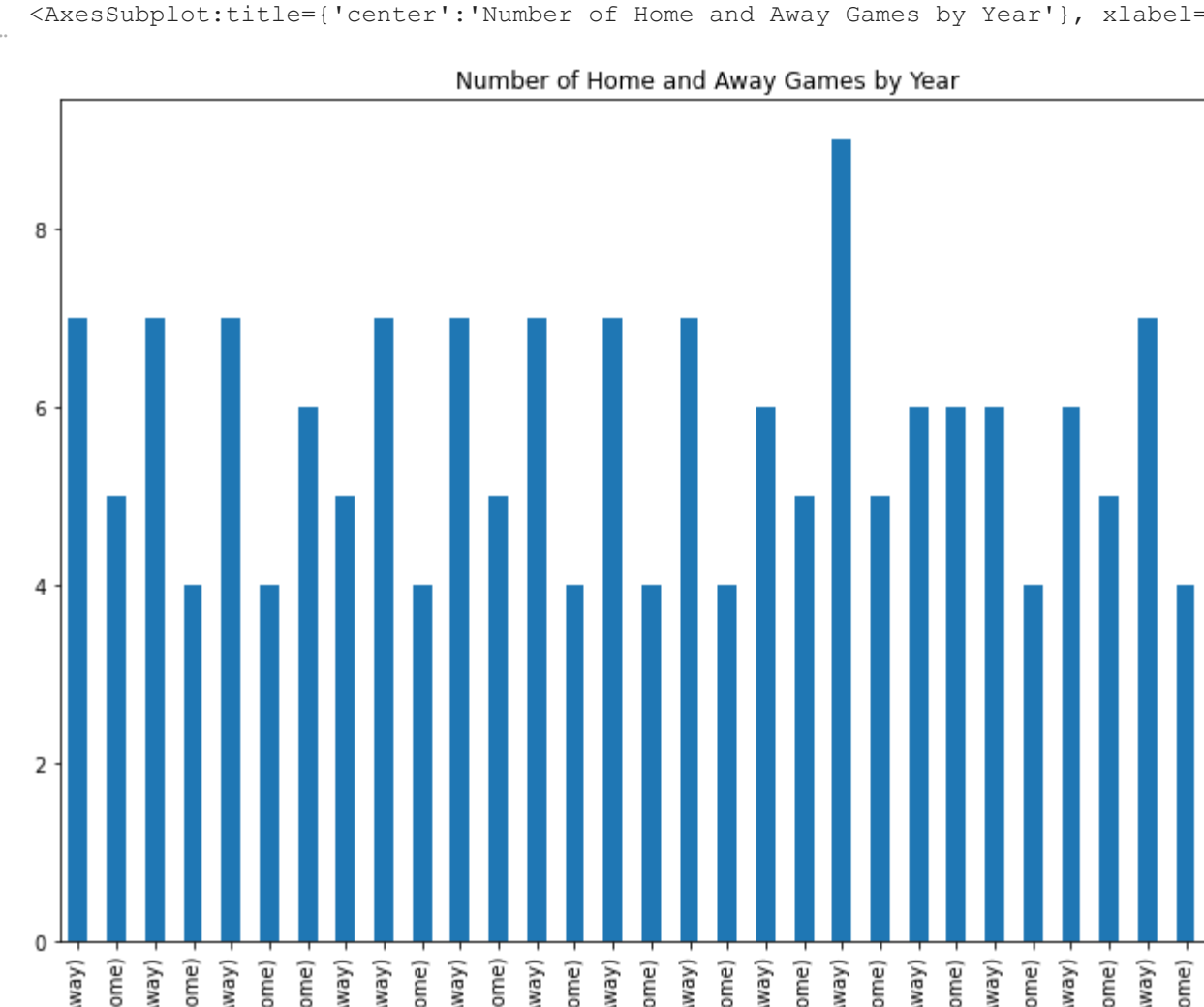
In [379]:
```
#get number of wins and losses per year
#There was fulctions in the numbers of wins and losses each year. 2013 had the highest number of wins across th
#while 2005 and 2019 tied for the greatest number of losses.
plt.rcParams['figure.figsize']=[12,8]
winloss_by_year=mydata.groupby("year")["winloss"].value_counts()
winloss_by_year.plot(kind="bar",title="Number of Wins and Losses by Year")
```

Out[379]: <AxesSubplot:title={'center':'Number of Wins and Losses by Year'}, xlabel='year,winloss'>

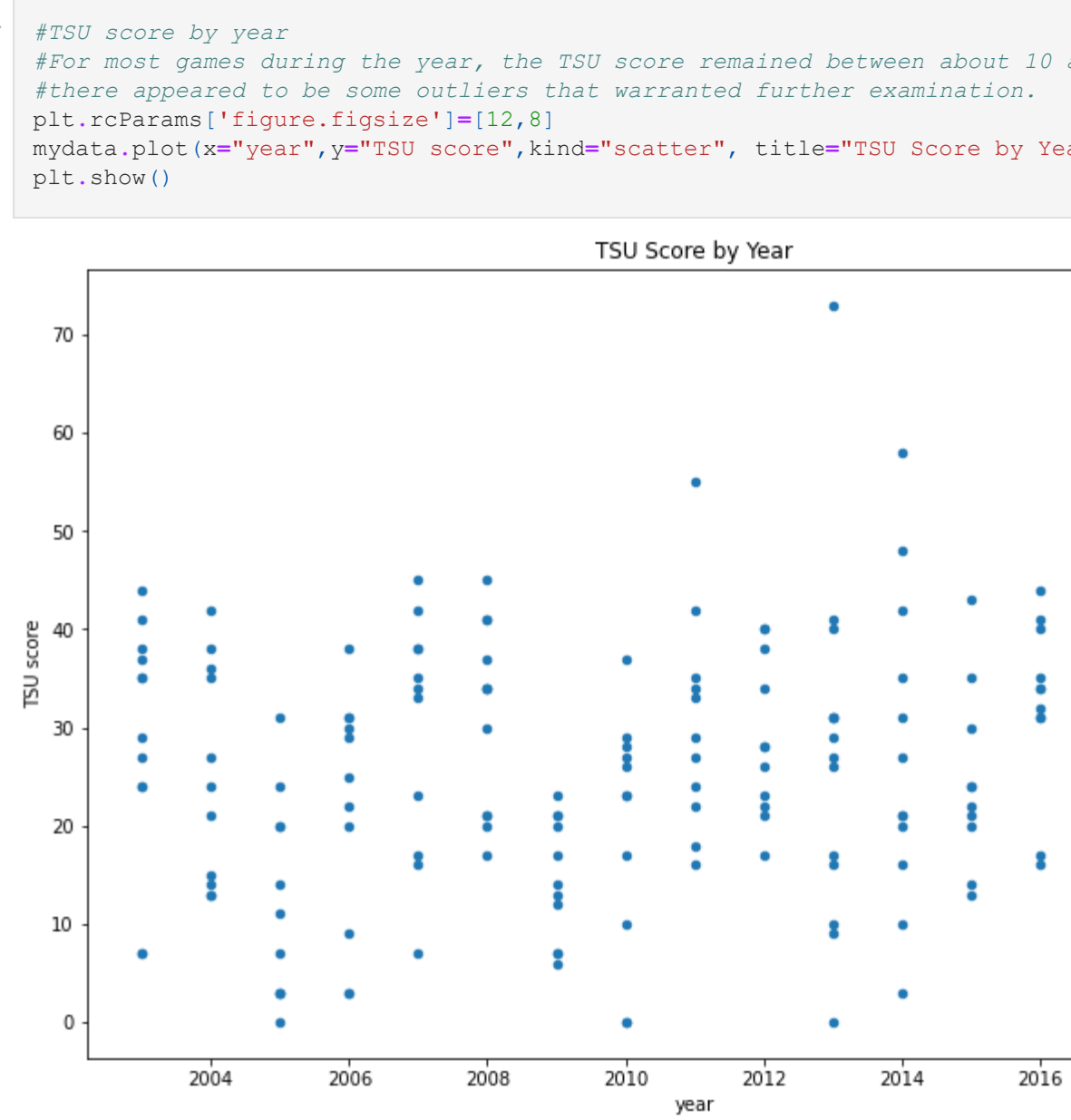


In [380]:
```
#number of games by locale by year
#The number of home and away games varied by year but there appeared to be some stability
#in the annual number of away games from 2003 to 2004 (7) and 2008-2011 (7).
#In 2013, TSU had the highest number of away games (9).
#There were 4 home games in the years of 2004, 2005, 2007, and 2009-2011.
plt.rcParams['figure.figsize']=[12,8]
locale_by_year=mydata.groupby("year")["locale"].value_counts()
locale_by_year.plot(kind="bar",title="Number of Home and Away Games by Year")
```

Out[386… <AxesSubplot:title={'center':'Number of Home and Away Games by Year'}, xlabel='year,locale'>



In [381…
```
#TSU score by year
#For most games during the year, the TSU score remained between about 10 and 45 points. Based on the scatterplo
#there appeared to be some outliers that warranted further examination.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSU score",kind="scatter", title="TSU Score by Year")
plt.show()
```



In [382…
```
#get outliers of TSU score
#There were 2 outliers on the TSU score, a 73 points against Central State University and
#60 points against Virginia University Lynchburg.
#Another TSU score variable was made with the values of thes outliers being replaced the mean.
q1 = mydata["TSU score"].quantile(0.25)
q3 = mydata["TSU score"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSU score"]][(mydata["TSU score"]>outlier_top_lim) | (mydata["TSU score"]<out
```

Out[382…

	date	city	opponent	TSU score
116	2013-09-28	St. Louis	Central State University	73
167	2017-11-04	Nashville	Virginia University Lynchburg	60

In [383…
```
#replace outliers of TSU score with the mean
mydata["TSU score_new"]=np.where((mydata["TSU score"]>outlier_top_lim) | (mydata["TSU score"]<outlier_bottom_li
```

In [384…
```
#descriptive statistics for TSU score
mydata["TSU score"].describe()
```

Out[384…
```
count    191.000000
mean      25.643979
std       12.489530
min        0.000000
25%       17.000000
50%       26.000000
75%       34.000000
max       73.000000
Name: TSU score, dtype: float64
```

In [385…
```
#descriptive statistics for TSU score with corrected outliers
#The descriptive statistics for the new variable showed a lower mean(25.2 compared to 25.6) and standard deviati
#(11.7 compared to 12.5) than the original variable after replacement of outliers.
mydata["TSU score_new"].describe()
```

Out[385…
```
count    191.000000
mean      25.216167
std       11.738355
min        0.000000
25%       17.000000
50%       25.643979
75%       34.000000
max       58.000000
Name: TSU score_new, dtype: float64
```

In [386…
```
#opponent score by year
#The scatterplot of the opponent score showed that for most games the opponents score remained between 0 and 5
#were suspected and examined below.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="opponent score",kind="scatter", title="Opponent Score by Year")
plt.show()
```

In [387…
```
#get outliers of opponent score
#There was one outlier on the opponent score variable, 63 points scored by the Air Force Academy.
#This was replaced with the mean below.
q1 = mydata["opponent score"].quantile(0.25)
q3 = mydata["opponent score"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","opponent score"]][(mydata["opponent score"]>outlier_top_lim) | (mydata["oppon
```

Out[387…

	date	city	opponent	opponent score
93	2011-09-24	Air Force Academy	Air Force Academy	63

In [388…
```
#replace outliers of opponent score with the mean
mydata["opponent score_new"]=np.where((mydata["opponent score"]>outlier_top_lim) | (mydata["opponent score"]<ou
```

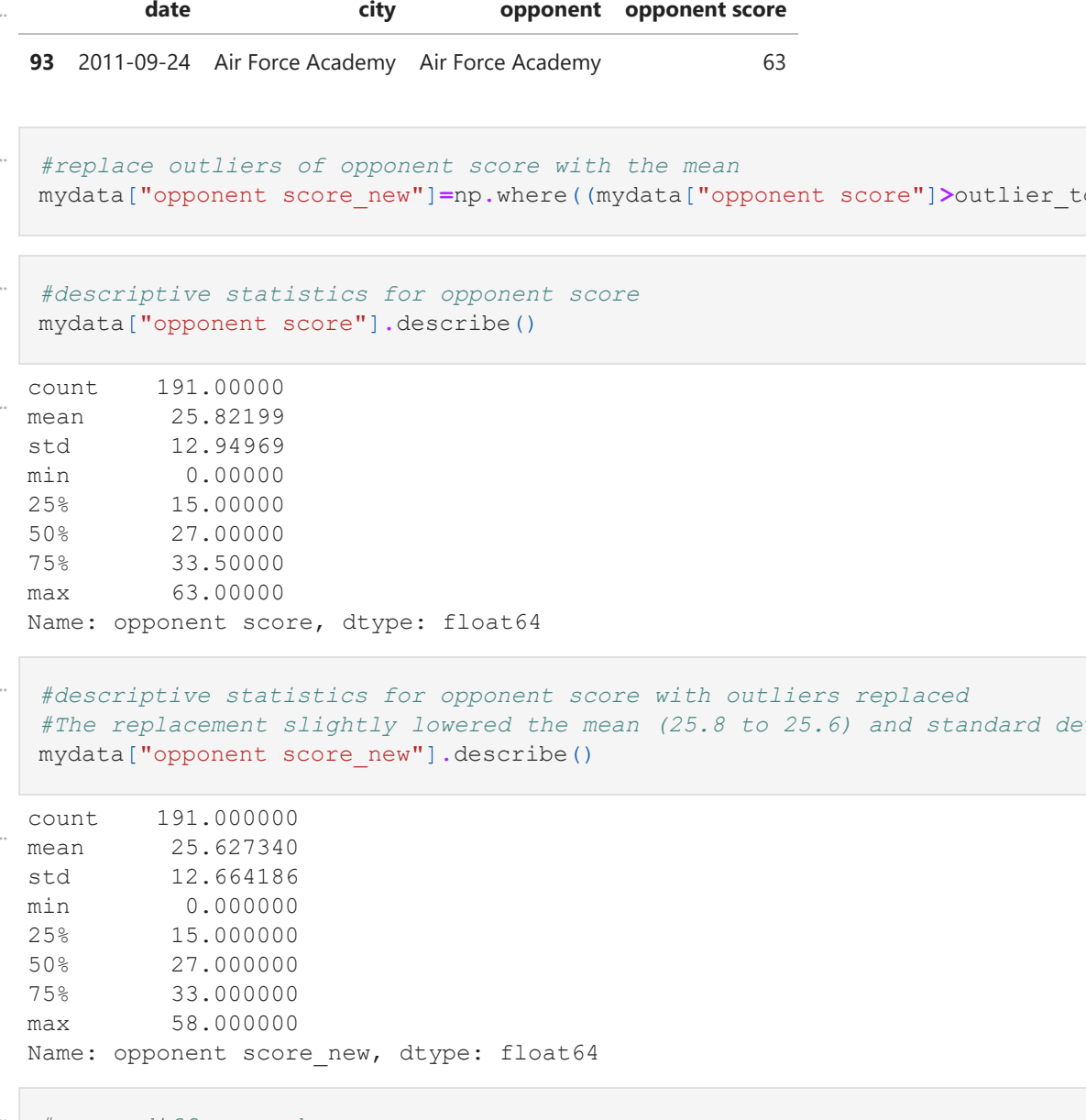
In [389…
```
#descriptive statistics for opponent score
mydata["opponent score"].describe()
```

Out[389…
```
count    191.000000
mean      25.82199
std       12.94969
min        0.00000
25%       15.00000
50%       27.00000
75%       33.50000
max       63.00000
Name: opponent score, dtype: float64
```

In [390…
```
#descriptive statistics for opponent score with outliers replaced
#The replacement slightly lowered the mean (25.8 to 25.6) and standard deviation (12.9 to 12.7) for opponent sc
mydata["opponent score_new"].describe()
```

Out[390…
```
count    191.000000
mean      25.627340
std       12.664186
min        0.000000
25%       15.000000
50%       27.000000
75%       33.000000
max       58.000000
Name: opponent score_new, dtype: float64
```

In [391…
```
#score difference by year
#The scatterplot of the score difference showed that it remained between -20 and 40 for most games. It also sho
#which were examined below.
plt.rcParams["figure.figsize"]=[12,8]
mydata.plot(x="year",y="scorediff",kind="scatter", title="Score Difference by Year")
plt.show()
```



In [392…
```
#get outliers of score difference
#There were 7 outliers on the score difference variable. 2 were games in which TSU lost to Easter Kentucky Univ
#at least 44 points. The other 5 games where those in which TSU won by at least 44 points. A new variables was
#that replaced these outliers with the mean.
q1 = mydata["scorediff"].quantile(0.25)
q3 = mydata["scorediff"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","scorediff"]][(mydata["scorediff"]>outlier_top_lim) | (mydata["scorediff"]<out
```

Out[392…

	date	city	opponent	scorediff
33	2005-11-19	Nashville	Eastern Kentucky University	-49
95	2011-10-08	Nashville	Southeast Missouri State University	52
116	2013-09-28	St. Louis	Central State University	67
121	2013-11-02	Richmond	Eastern Kentucky University	-44
126	2014-08-30	Nashville	Edward Waters College	52
148	2016-09-03	Nashville	University of Arkansas Pine Bluff	44
167	2017-11-04	Nashville	Virginia University Lynchburg	60

In [393…
```
#replace outliers of score difference with mean
mydata["scorediff_new"]=np.where((mydata["scorediff"]>outlier_top_lim) | (mydata["scorediff"]<outlier_bottom_li
```

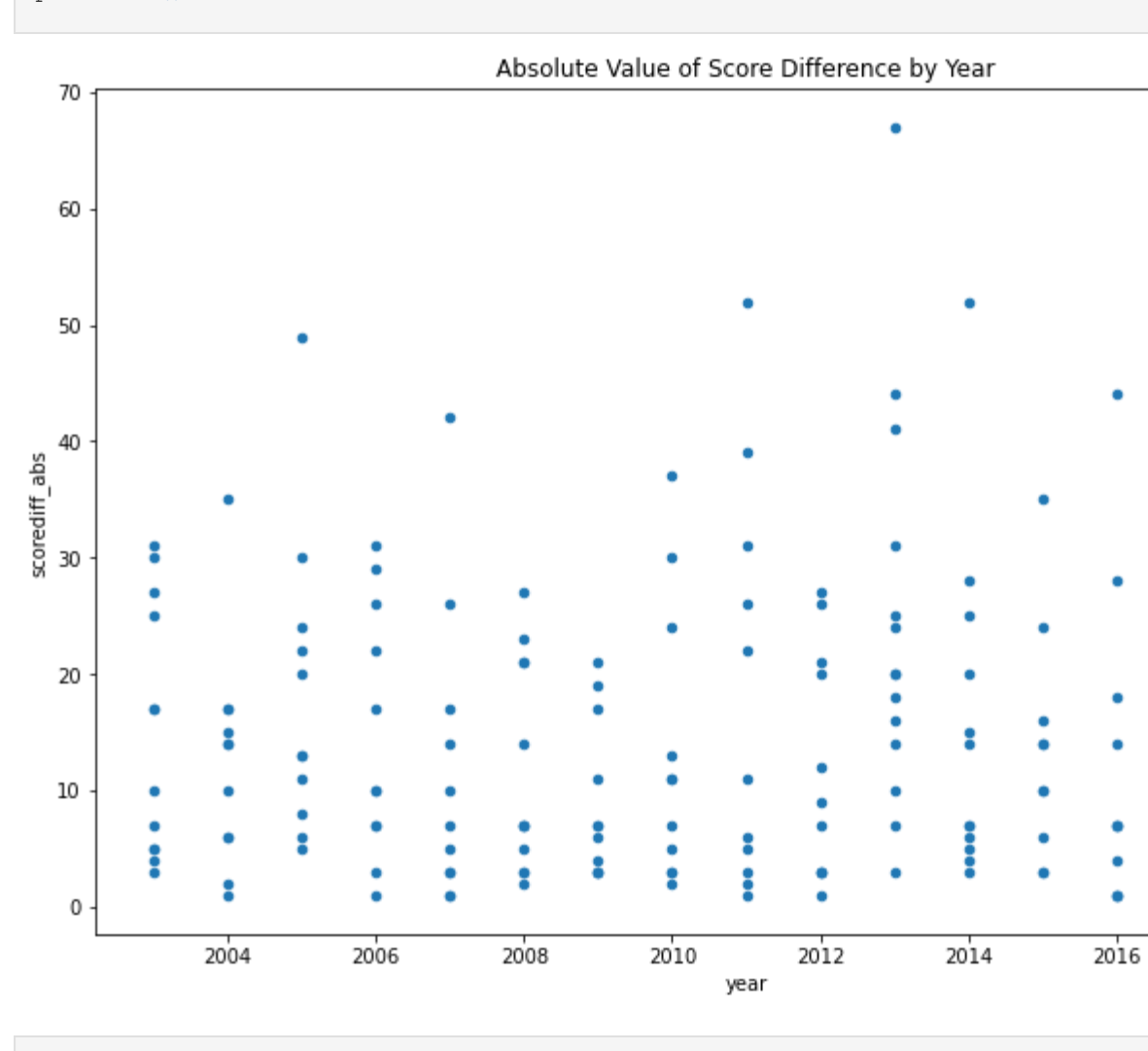
In [394…
```
#descriptive statistics for score difference
mydata.scorediff.describe()
```

Out[394…
```
count    191.000000
mean      -0.178010
std       19.109373
min      -49.000000
25%      -11.000000
50%       -3.000000
75%       10.000000
max       67.000000
Name: scorediff, dtype: float64
```

In [395…
```
#descriptive statistics for score difference with outliers replaced
#With the ouliers replaced with the mean, the mean of the score difference dropped to -1.1
#and the standard deviation dropped to 16.1
mydata.scorediff_new.describe()
```

Out[395…
```
count    191.000000
mean      -1.137414
std       16.119802
min      -42.000000
25%      -11.000000
50%       -1.000000
75%       10.000000
max       35.000000
Name: scorediff_new, dtype: float64
```

In [396…
```
#absolute value of score difference by year
#Based on the scatterplot below, for most games, the absolute value of the score difference remained between 0
#There were also outliers that were examined below.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="scorediff_abs",kind="scatter", title="Absolute Value of Score Difference by Year")
plt.show()
```



In [397…
```
#get outliers of absolute value of score difference
#There were 5 outliers on the absolute value of the score difference variable, all which had values of at least
q1 = mydata["scorediff_abs"].quantile(0.25)
q3 = mydata["scorediff_abs"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","scorediff_abs"]][(mydata["scorediff_abs"]>outlier_top_lim) | (mydata["scored
```

Out[397…

	date	city	opponent	scorediff_abs
33	2005-11-19	Nashville	Eastern Kentucky University	49
95	2011-10-08	Nashville	Southeast Missouri State University	52
116	2013-09-28	St. Louis	Central State University	67
126	2014-08-30	Nashville	Edward Waters College	52
167	2017-11-04	Nashville	Virginia University Lynchburg	60

In [398…
```
#replace outliers of absolute value of score difference with mean
mydata["scorediff_abs_new"]=np.where((mydata["scorediff_abs"]>outlier_top_lim) | (mydata["scorediff_abs"]<outl
```

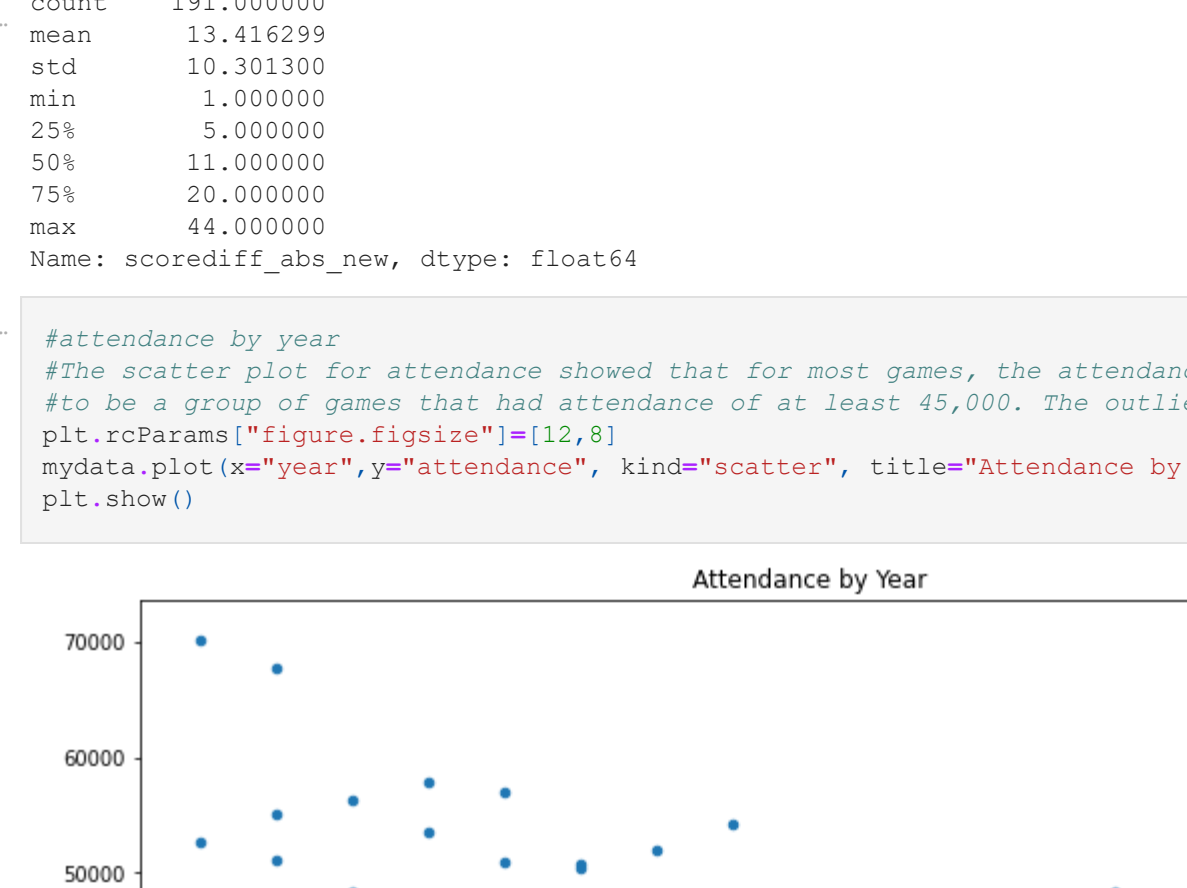
In [399…
```
#descriptive statistics for absolute difference of score difference
mydata.scorediff_abs.describe()
```

Out[399…
```
count    191.000000
mean      14.502618
std       12.400286
min        1.000000
25%        5.000000
50%       11.000000
75%       21.500000
max       67.000000
Name: scorediff_abs, dtype: float64
```

In [400…
```
#descriptive statistics for absolute difference of score difference with outliers replaced
#As with the previous variables that replaced the outliers with the mean, the mean and standard deviation of th
#absolute value of the score difference dropped once the outliers were replaced.
mydata.scorediff_abs_new.describe()
```

Out[400…
```
count    191.000000
mean      13.416299
std       10.301300
min        1.000000
25%        5.000000
50%       11.000000
75%       20.000000
max       44.000000
Name: scorediff_abs_new, dtype: float64
```

In [401…
```
#attendance by year
#The scatter plot for attendance showed that for most games, the attendance was less than 35,000. However, the
#to be a group of games that had attendance of at least 45,000. The outliers were assessed below.
plt.rcParams["figure.figsize"]=[12,8]
mydata.plot(x="year",y="attendance", kind="scatter", title="Attendance by Year")
plt.show()
```



In [402…
```
#get outliers of attendance
#There were 20 outliers for attendance. These games had attendance figures of at least about 47,000. These game
#were against Jackson State University, Florida A&M University, and South Carolina State University. The attenda
#for these games were replaced with the mean.
q1 = mydata["attendance"].quantile(0.25)
q3 = mydata["attendance"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","attendance"]][(mydata["attendance"]>outlier_top_lim) | (mydata["attendance"]<
```

Out[402…

	date	city	opponent	attendance
2	2003-09-13	Memphis	Jackson State University	52603
3	2003-09-20	Atlanta	Florida A&M University	70185
14	2004-09-18	Memphis	Jackson State University	55015
15	2004-09-25	Atlanta	Florida A&M University	67712
16	2004-10-02	Indianapolis	South Carolina State University	51082
24	2005-09-10	Memphis	Jackson State University	48300
26	2005-09-24	Atlanta	Florida A&M University	56297
36	2006-09-16	Memphis	Jackson State University	53441
38	2006-09-30	Atlanta	Florida A&M University	57885
46	2007-09-08	Memphis	Jackson State University	50879
49	2007-09-29	Atlanta	Florida A&M University	56990
58	2008-09-13	Memphis	Jackson State University	50794
60	2008-09-27	Atlanta	Florida A&M University	50428
71	2009-09-26	Atlanta	Florida A&M University	51950
82	2010-09-25	Atlanta	Florida A&M University	54202
128	2014-09-13	Memphis	Jackson State University	46914
139	2015-09-12	Memphis	Jackson State University	48385
149	2016-09-10	Memphis	Jackson State University	46263
160	2017-09-09	Memphis	Jackson State University	47407
181	2019-09-14	Memphis	Jackson State University	48347

In [403…
```
#replace outliers of attendance with mean
mydata["attendance_new"]=np.where((mydata["attendance"]>outlier_top_lim) | (mydata["attendance"]<outlier_botto
```

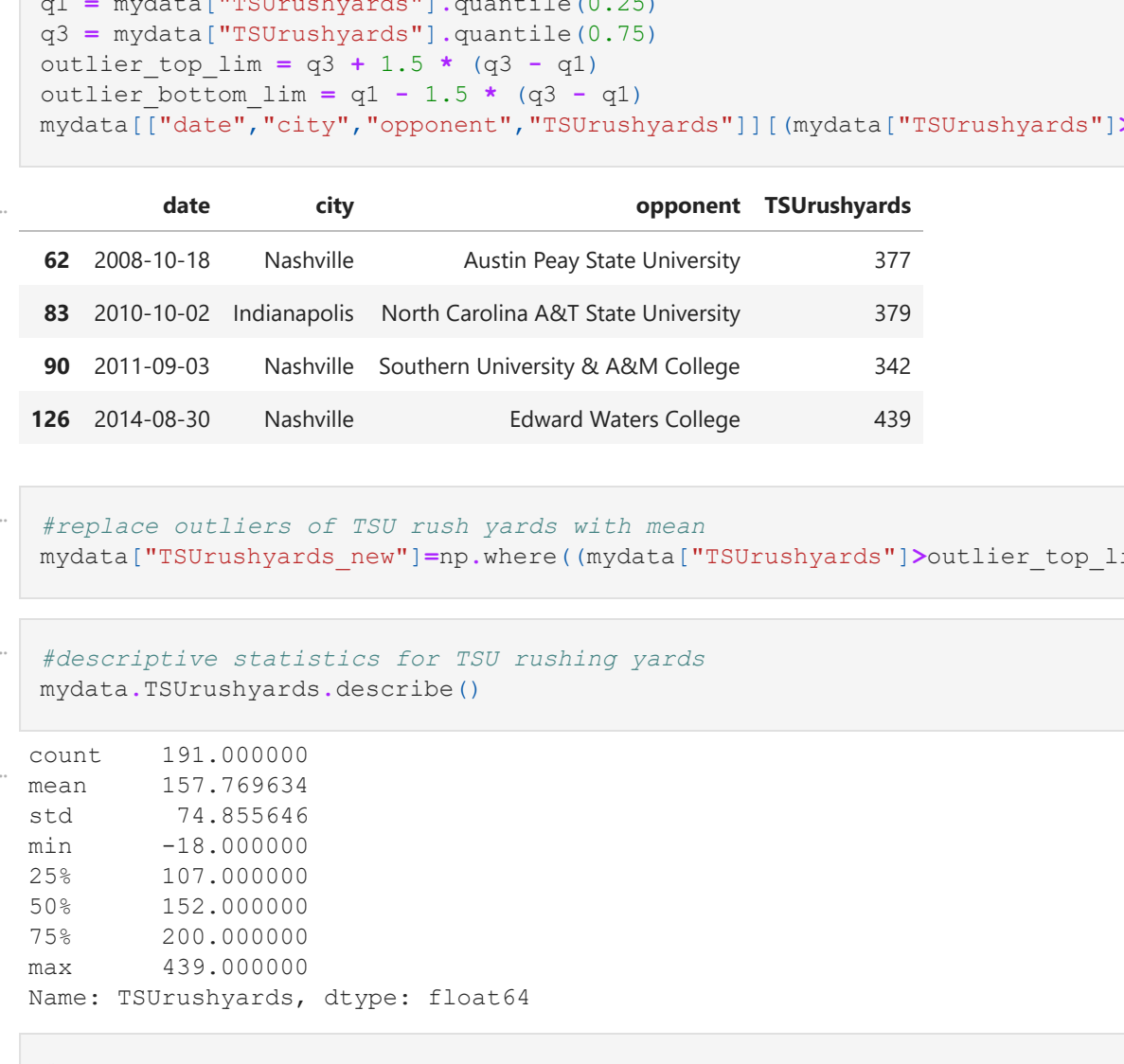
In [404…
```
#descriptive statistics for attendance
mydata.attendance.describe()
```

Out[404…
```
count      191.000000
mean     16985.230366
std      15493.766885
min       1776.000000
25%       6402.500000
50%      10001.000000
75%      22306.000000
max      70185.000000
Name: attendance, dtype: float64
```

In [405…
```
#descriptive statistics for attendance with outliers replaced
#When the outliers were replaced, the mean of attendance dropped from 16,985 to 13,187. The standard deviation
#from 15,493 to 9,113.
mydata.attendance_new.describe()
```

Out[405…
```
count      191.000000
mean     13187.458677
std       9113.636740
min       1776.000000
25%       6402.500000
50%      10001.000000
75%      16985.230366
max      44688.000000
Name: attendance_new, dtype: float64
```

In [406…
```
#TSU rushing yards by year
#The scatter plot for TSU rushing yards showed that the rushing yards ranged from 0 to 300 per game annually.
#There appeared to be outliers which were assessed below.
plt.rcParams["figure.figsize"]=[12,8]
mydata.plot(x="year",y="TSUrushyards",kind="scatter",title="TSU Rushing Yards by Year")
plt.show()
```



In [407…
```
#get outliers of TSU rushing yards
#There were 4 outliers on the TSU rushing yards variable. In these 4 games, TSU had at least 342 rushing yards.
#These outliers were replaced with the mean below.
q1 = mydata["TSUrushyards"].quantile(0.25)
q3 = mydata["TSUrushyards"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUrushyards"]][(mydata["TSUrushyards"]>outlier_top_lim) | (mydata["TSUrushya
```

Out[407…

	date	city	opponent	TSUrushyards
62	2008-10-18	Nashville	Austin Peay State University	377
83	2010-10-02	Indianapolis	North Carolina A&T State University	379
90	2011-09-03	Nashville	Southern University & A&M College	342
126	2014-08-30	Nashville	Edward Waters College	439

In [408…
```
#replace outliers of TSU rush yards with mean
mydata["TSUrushyards_new"]=np.where((mydata["TSUrushyards"]>outlier_top_lim) | (mydata["TSUrushyards"]<outlier_
```

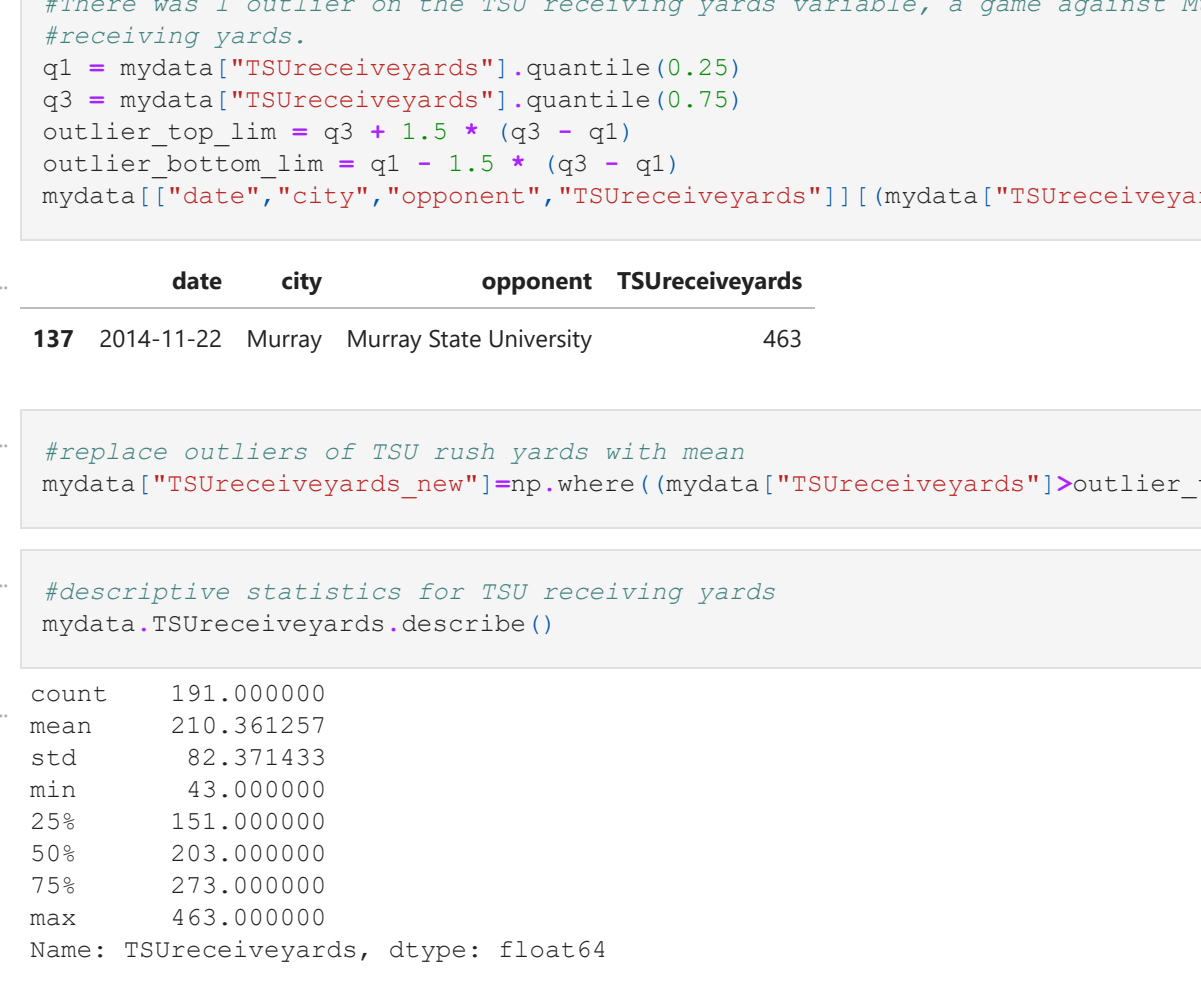
In [409…
```
#descriptive statistics for TSU rushing yards
mydata.TSUrushyards.describe()
```

Out[409…
```
count    191.000000
mean     157.769634
std       74.855646
min      -18.000000
25%      107.000000
50%      152.000000
75%      200.000000
max      439.000000
Name: TSUrushyards, dtype: float64
```

In [410…
```
#descriptive statistics for TSU rushing yards with outliers replaced
#The mean and standard deivation for TSU rushing yards dropped to 153 and about 67, respectively when the outli
#were replaced.
mydata.TSUrushyards_new.describe()
```

Out[410…
```
count    191.000000
mean     153.026589
std       66.897678
min      -18.000000
25%      107.000000
50%      152.000000
75%      195.500000
max      320.000000
Name: TSUrushyards_new, dtype: float64
```

In [411…
```
#TSU receiving yards by year
#A scatterplot of TSU recieving yards showed that for most games the receiving  yard ranged between 100 and 300
#Outliers on this variable were assessed below.
plt.rcParams["figure.figsize"]=[12,8]
mydata.plot(x="year",y="TSUreceiveyards",kind="scatter",title="TSU Receiving Yards by Year")
plt.show()
```



In [412…
```
#get outliers of TSU receiving yards
#There was 1 outlier on the TSU receiving yards variable, a game against Murray State University where TSU had
#receiving yards.
q1 = mydata["TSUreceiveyards"].quantile(0.25)
q3 = mydata["TSUreceiveyards"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUreceiveyards"]][(mydata["TSUreceiveyards"]>outlier_top_lim) | (mydata["TSU
```

Out[412…

	date	city	opponent	TSUreceiveyards
137	2014-11-22	Murray	Murray State University	463

In [413…
```
#replace outliers of TSU rush yards with mean
mydata["TSUreceiveyards_new"]=np.where((mydata["TSUreceiveyards"]>outlier_top_lim) | (mydata["TSUreceiveyards"]
```

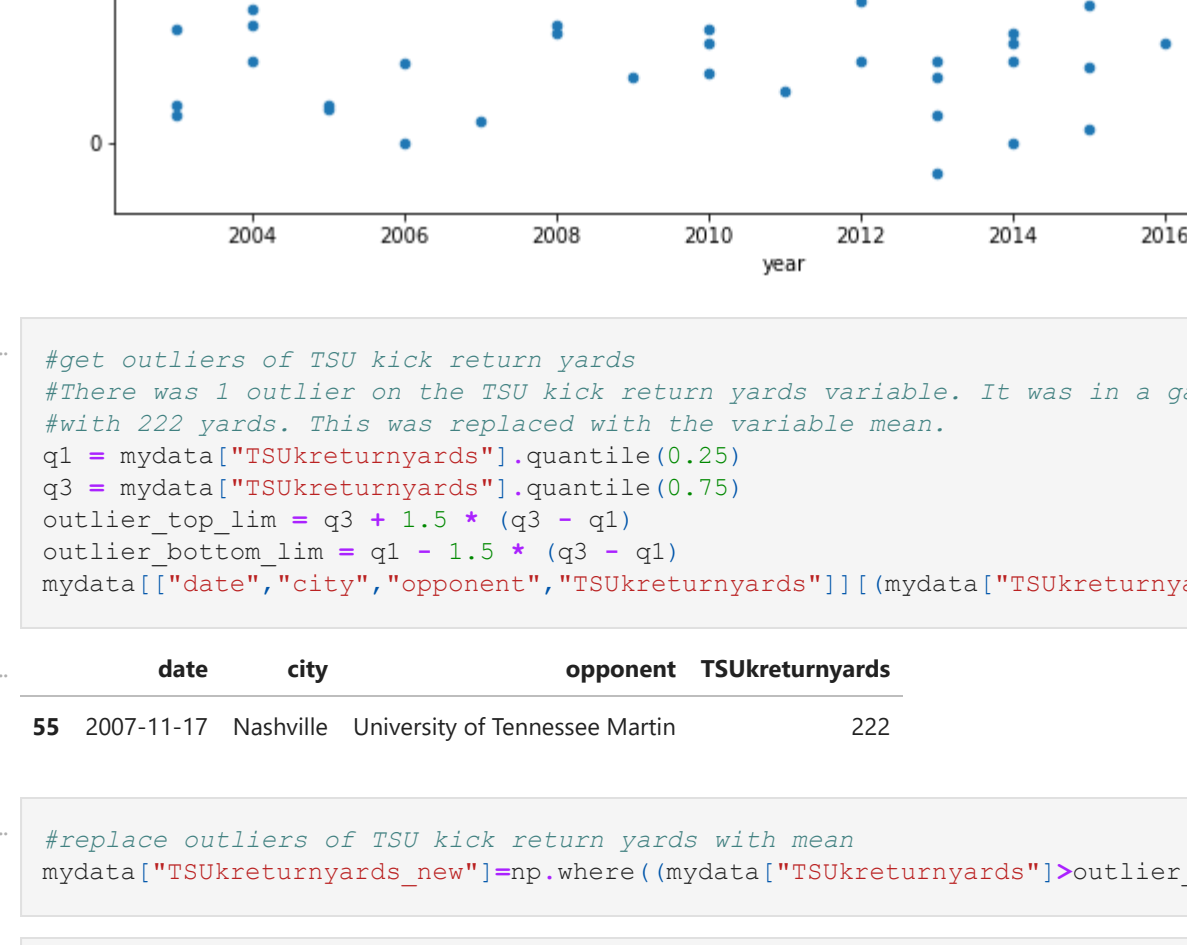
In [414…
```
#descriptive statistics for TSU receiving yards
mydata.TSUreceiveyards.describe()
```

Out[414…
```
count    191.000000
mean     210.361257
std       82.371433
min       43.000000
25%      151.000000
50%      203.000000
75%      273.000000
max      463.000000
Name: TSUreceiveyards, dtype: float64
```

In [415…
```
#descriptive statistics for TSU receiving yards with outliers replaced
#Replacing the outlier with the mean lowered the mean of the variable slightly to 210 and
#lowered the standard deviation slightly to 80.
mydata.TSUreceiveyards_new.describe()
```

Out[415…
```
count    191.000000
mean     209.038541
std       80.295493
min       43.000000
25%      151.000000
50%      203.000000
75%      272.000000
max      395.000000
Name: TSUreceiveyards_new, dtype: float64
```

In [416…
```
#TSU kick return yards by year
#A scatterplot of kick return yards showed that for most games, the kick return yards ranged from 0 to about 1
#Potential outliers were examined below.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUkreturnyards",kind="scatter", title="TSU Kick Return Yards by Year")
plt.show()
```



In [417…
```
#get outliers of TSU kick return yards
#There was 1 outlier on the TSU kick return yards variable. It was in a game against University of Tennessee Ma
#with 222 yards. This was replaced with the variable mean.
q1 = mydata["TSUkreturnyards"].quantile(0.25)
q3 = mydata["TSUkreturnyards"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUkreturnyards"]][(mydata["TSUkreturnyards"]>outlier_top_lim) | (mydata["TSU
```

Out[417…

	date	city	opponent	TSUkreturnyards
55	2007-11-17	Nashville	University of Tennessee Martin	222

In [418…
```
#replace outliers of TSU kick return yards with mean
mydata["TSUkreturnyards_new"]=np.where((mydata["TSUkreturnyards"]>outlier_top_lim) | (mydata["TSUkreturnyards"]
```

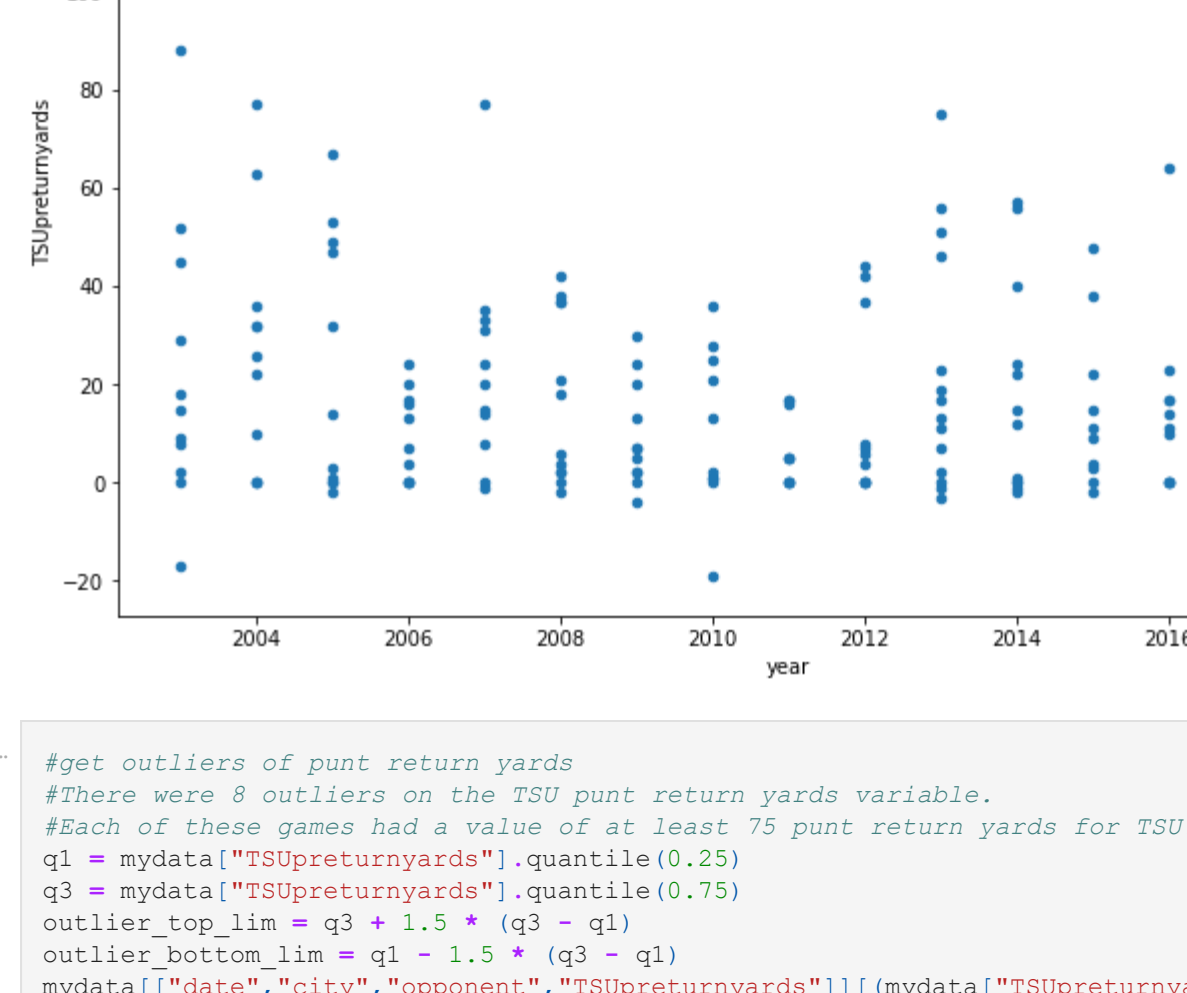
In [419…
```
#descriptive statistics for TSU kick return yards
mydata.TSUkreturnyards.describe()
```

Out[419…
```
count    191.000000
mean      78.670157
std       44.059662
min       -9.000000
25%       48.000000
50%       72.000000
75%      110.500000
max      222.000000
Name: TSUkreturnyards, dtype: float64
```

In [420…
```
#descriptive statistics for TSU kick return yards with outliers replaced
#Replacing the value on the outlier caused the mean and standard deviation of TSU kick return yards to drop sl
mydata.TSUkreturnyards_new.describe()
```

Out[420…
```
count    191.000000
mean      77.919739
std       42.808461
min       -9.000000
25%       48.000000
50%       72.000000
75%      109.000000
max      198.000000
Name: TSUkreturnyards_new, dtype: float64
```

In [421…
```
#TSU punt return yards by year
#A scatter plot of the TSU punt return yrad showed that for most games it ranged from 0 to 55. It also showed t
#for outliers which are examined below.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUpreturnyards",kind="scatter", title="TSU Punt Return Yards by Year")
plt.show()
```



In [422…
```
#get outliers of punt return yards
#There were 8 outliers on the TSU punt return yards variable.
#Each of these games had a value of at least 75 punt return yards for TSU. These were replaced with the mean be
q1 = mydata["TSUpreturnyards"].quantile(0.25)
q3 = mydata["TSUpreturnyards"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUpreturnyards"]][(mydata["TSUpreturnyards"]>outlier_top_lim) | (mydata["TSU
```

Out[422…

	date	city	opponent	TSUpreturnyards
0	2003-08-30	Nashville	South Carolina State University	88
2	2003-09-13	Memphis	Jackson State University	107
19	2004-10-30	Charleston	Eastern Illinois University	77
53	2007-11-03	Nashville	Murray State University	77
82	2010-09-25	Atlanta	Florida A&M University	142
95	2011-10-08	Nashville	Southeast Missouri State University	115
116	2013-09-28	St. Louis	Central State University	75
167	2017-11-04	Nashville	Virginia University Lynchburg	94

In [423…
```
#replace outliers of TSU punt return yards with mean
mydata["TSUpreturnyards_new"]=np.where((mydata["TSUpreturnyards"]>outlier_top_lim) | (mydata["TSUpreturnyards"]
```

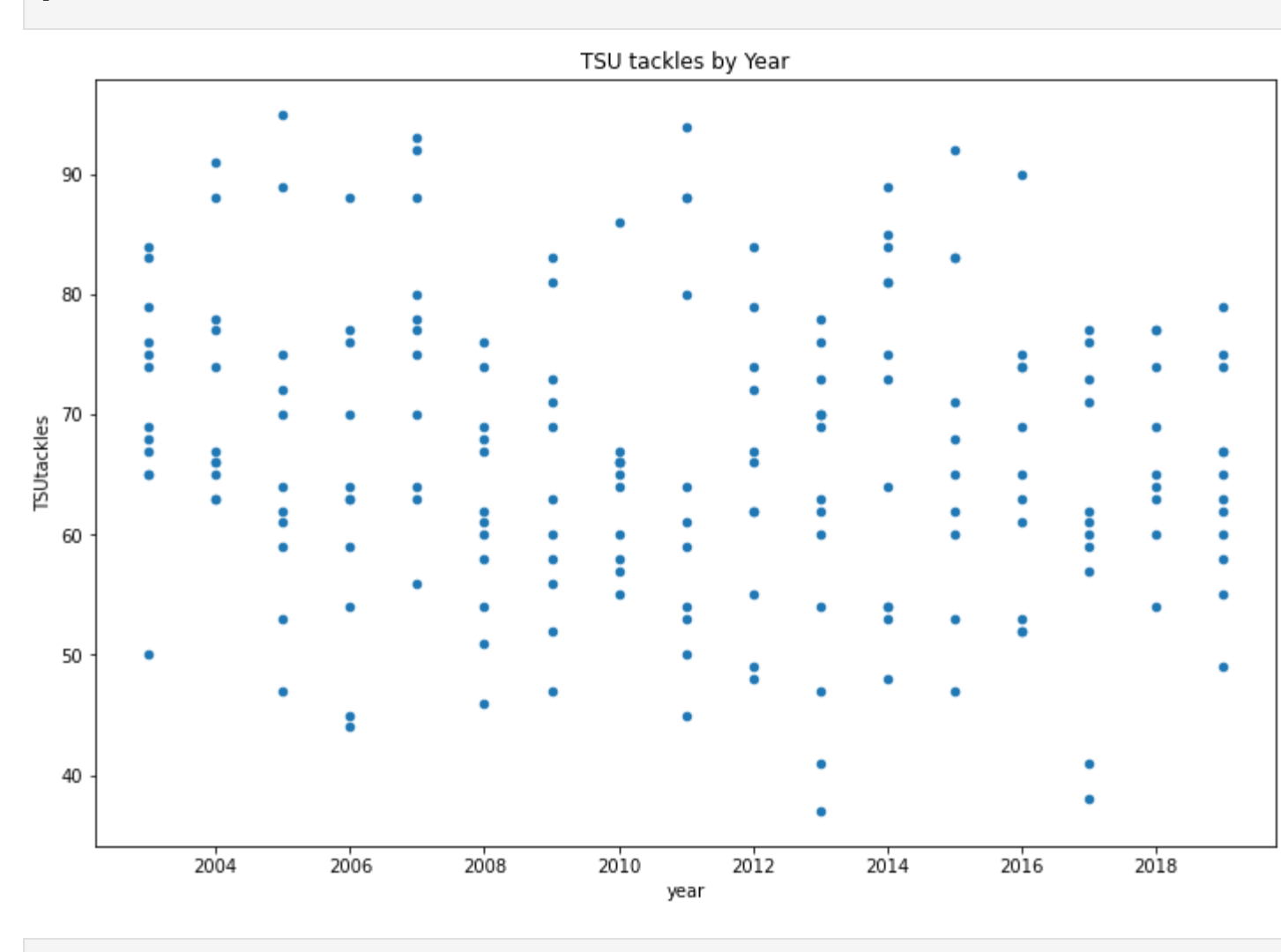
In [424…
```
#descriptive statistics for TSU punt return yards
mydata.TSUpreturnyards.describe()
```

Out[424…
```
count    191.000000
mean      18.753927
std       24.579563
min      -19.000000
25%        0.000000
50%       11.000000
75%       29.000000
max      142.000000
Name: TSUpreturnyards, dtype: float64
```

In [425…
```
#descriptive statistics for TSU punt return yards with outliers replaced
#Replacing the outliers dropped the mean of the variable to 15 and the standard deviation to about 18.
mydata.TSUpreturnyards_new.describe()
```

Out[425…
```
count    191.000000
mean      15.481840
std       17.785031
min      -19.000000
25%        0.000000
50%       11.000000
75%       24.000000
max       67.000000
Name: TSUpreturnyards_new, dtype: float64
```

In [426…
```
#TSU tackles by year
#A scatter plot of TSU tackes per game showed that for most games they ranged from 45 to 85.
#Anaylsis below looked for outliers on the variable.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUtackles",kind="scatter", title="TSU tackles by Year")
plt.show()
```



In [427…
```
#get outliers of TSU tackles
#There were no outliers on the variable so no change was made to the variable.
q1 = mydata["TSUtackles"].quantile(0.25)
q3 = mydata["TSUtackles"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUtackles"]][(mydata["TSUtackles"]>outlier_top_lim) | (mydata["TSUtackles"]<
```

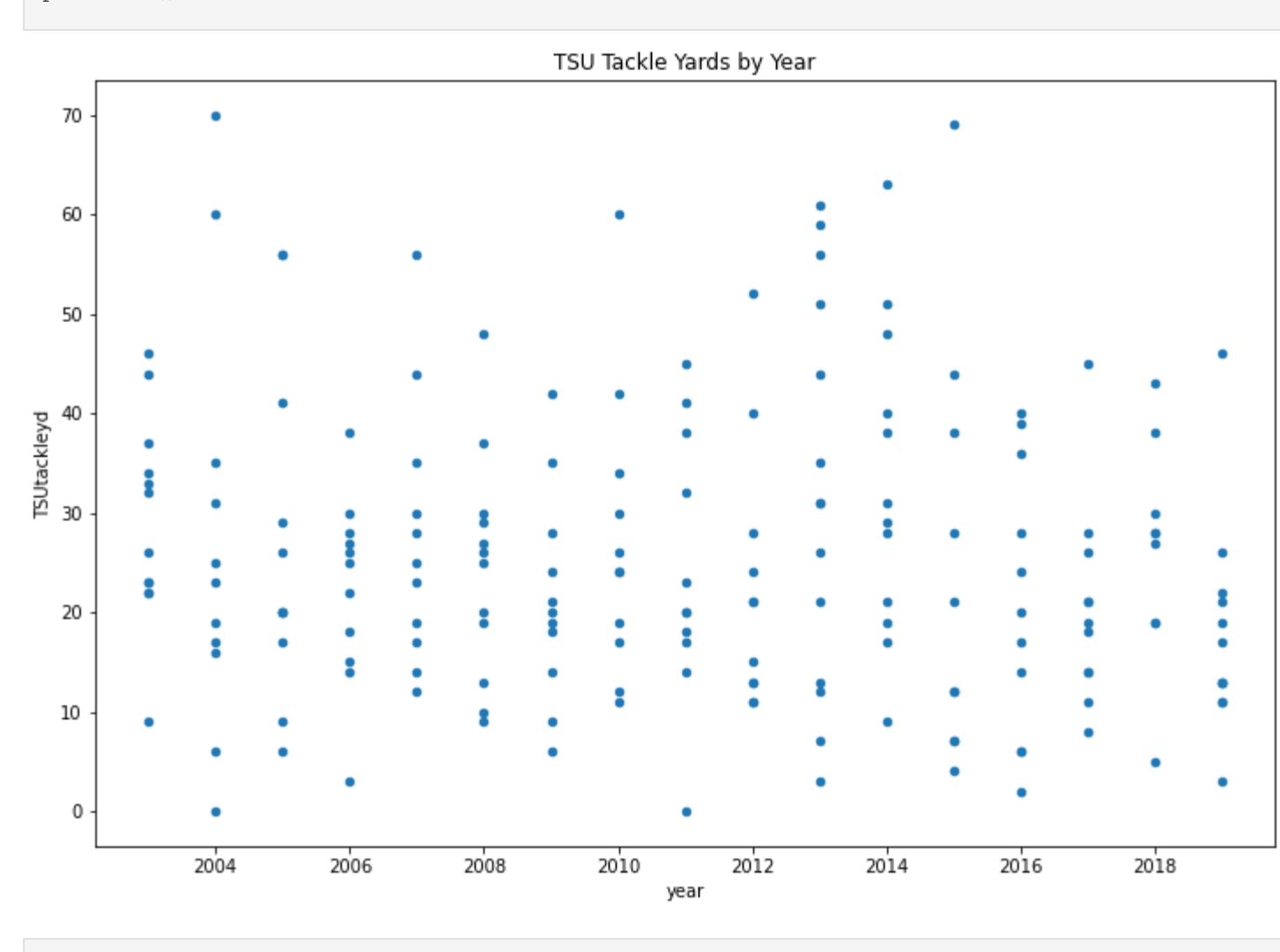
Out[427…

date	city	opponent	TSUtackles

In [428…
```
#descriptive statistics for TSU tackles
mydata.TSUtackles.describe()
```

Out[428…
```
count    191.000000
mean      66.685864
std       12.195588
min       37.000000
25%       59.000000
50%       66.000000
75%       75.000000
max       95.000000
Name: TSUtackles, dtype: float64
```

In [429…
```
#TSU tackle yards by year
#TSU tackle yards for most games from 2003 to 2019 ranged from 0 to  about 45 per game, based on a scatterplot
#of the variable. Analysis was done below to examine outliers.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUtackleyd",kind="scatter", title="TSU Tackle Yards by Year")
plt.show()
```



In [430…
```
#get outliers of TSU tackle yards
#There were 6 outliers for TSU tackle yards. They came in 2 games against Florida A&M University, 2 games agai
#University, and a game each against University of Tennessee Martin and Jacksonville State University. In each
#games, TSU had more than 60 tackle yards. For each of these outliers, the value was replaced with the mean.
q1 = mydata["TSUtackleyd"].quantile(0.25)
q3 = mydata["TSUtackleyd"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUtackleyd"]][(mydata["TSUtackleyd"]>outlier_top_lim) | (mydata["TSUtackleyd
```

Out[430…

	date	city	opponent	TSUtackleyd
13	2004-09-09	Martin	University of Tennessee Martin	60
15	2004-09-25	Atlanta	Florida A&M University	70
82	2010-09-25	Atlanta	Florida A&M University	60
118	2013-10-12	Jacksonville	Jacksonville State University	61
137	2014-11-22	Murray	Murray State University	63
146	2015-11-07	Nashville	Murray State University	69

In [431…
```
#replace outliers of TSU tackle yards with mean
mydata["TSUtackleyd_new"]=np.where((mydata["TSUtackleyd"]>outlier_top_lim) | (mydata["TSUtackleyd"]<outlier_bot
```

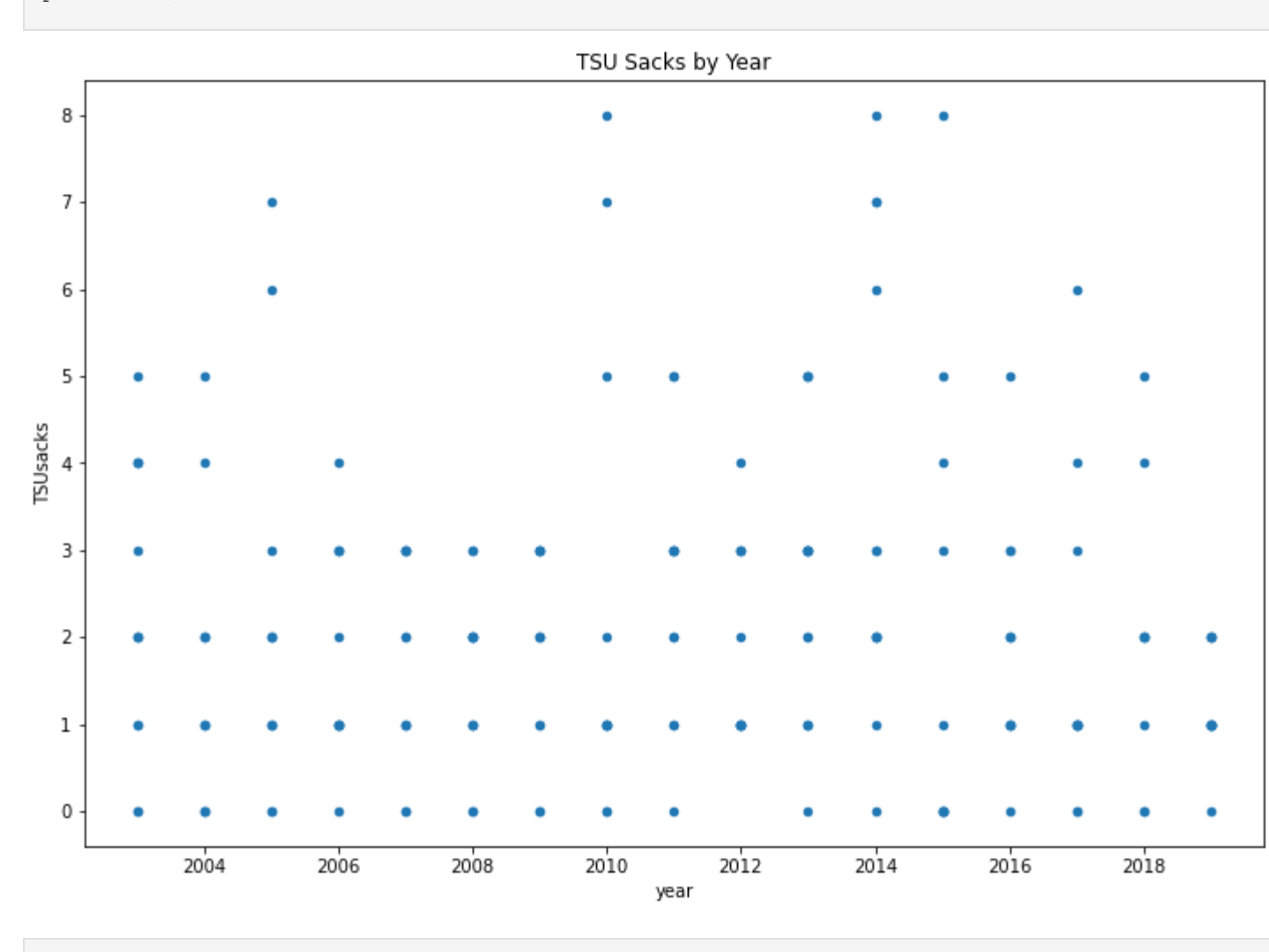
In [432…
```
#descriptive statistics for TSU tackle yards
mydata.TSUtackleyd.describe()
```

Out[432…
```
count    191.000000
mean      25.350785
std       14.392157
min        0.000000
25%       14.500000
50%       23.000000
75%       32.500000
max       70.000000
Name: TSUtackleyd, dtype: float64
```

In [433…
```
#descriptive statistics for TSU tackle yards with outliers replaced
#Replacing the outliers on TSU tackle yards lowered the mean slightly to 24 and
#lowered its standard deviation slightly to about 13.
mydata.TSUtackleyd_new.describe()
```

Out[433…
```
count    191.000000
mean      24.141909
std       12.584055
min        0.000000
25%       14.500000
50%       23.000000
75%       30.500000
max       59.000000
Name: TSUtackleyd_new, dtype: float64
```

In [434…
```
#TSU sacks by year
#For most games from 2003 to 2019, TSU had 5 or fewer sacks. The scatterplot of the TSU sacks variable
#showed evidence of outliers which were examined below.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUsacks",kind="scatter", title="TSU Sacks by Year")
plt.show()
```



In [435…
```
#get outliers of TSU sacks
#There were 7 outliers on the TSU sacks variable in games against Florida A&M University,
#Alabama A&M University, Jackson State University, and Murray State University. In each of these games, TSU had
#at least 7 sacks. These were replaced with the mean.
q1 = mydata["TSUsacks"].quantile(0.25)
q3 = mydata["TSUsacks"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUsacks"]][(mydata["TSUsacks"]>outlier_top_lim) | (mydata["TSUsacks"]<outlie
```

Out[435…

	date	city	opponent	TSUsacks
26	2005-09-24	Atlanta	Florida A&M University	7.0
79	2010-09-04	Nashville	Alabama A&M University	7.0
82	2010-09-25	Atlanta	Florida A&M University	8.0
128	2014-09-13	Memphis	Jackson State University	7.0
130	2014-09-27	Nashville	Florida A&M University	7.0
137	2014-11-22	Murray	Murray State University	8.0
146	2015-11-07	Nashville	Murray State University	8.0

In [436…
```
#replace outliers of TSU sacks with mean
mydata["TSUsacks_new"]=np.where((mydata["TSUsacks"]>outlier_top_lim) | (mydata["TSUsacks"]<outlier_bottom_lim),
```

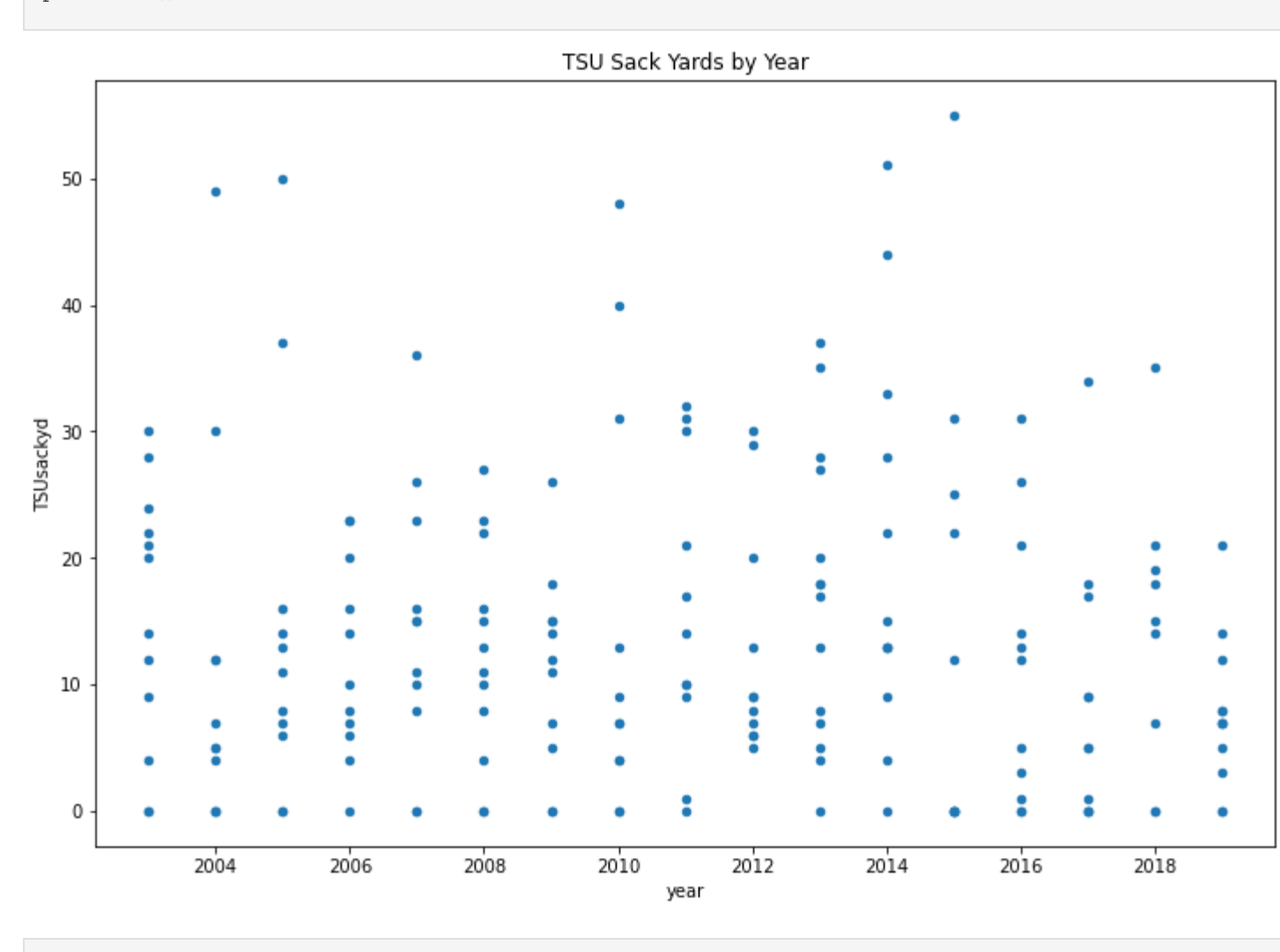
In [437…
```
#descriptive statistics for TSU sacks
mydata.TSUsacks.describe()
```

Out[437…
```
count    191.000000
mean       2.089005
std        1.776287
min        0.000000
25%        1.000000
50%        2.000000
75%        3.000000
max        8.000000
Name: TSUsacks, dtype: float64
```

In [438…
```
#descriptive statistics for TSU sacks with outliers replaced
#With the outliers replaced, the mean TSU sacks were 1.9 with a standard deviation of 1.4.
mydata.TSUsacks_new.describe()
```

Out[438…
```
count    191.000000
mean       1.893314
std        1.434319
min        0.000000
25%        1.000000
50%        2.000000
75%        3.000000
max        6.000000
Name: TSUsacks_new, dtype: float64
```

In [439…
```
#TSU sack yards by year
#A scatterplot shows that for most games the TSU sack yards ranged from 0 to 30. Outliers were examined below.
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUsackyd",kind="scatter", title="TSU Sack Yards by Year")
plt.show()
```



In [440…
```
#get outliers of TSU sack yards
#There were 6 outliers on the TSU sack yards variable. These were in games against Florida A&M University and M
#University where TSU. In each game, TSU had at least 44 sack yards. These values were placed with the mean be
q1 = mydata["TSUsackyd"].quantile(0.25)
q3 = mydata["TSUsackyd"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUsackyd"]][(mydata["TSUsackyd"]>outlier_top_lim) | (mydata["TSUsackyd"]<out
```

Out[440…

	date	city	opponent	TSUsackyd
15	2004-09-25	Atlanta	Florida A&M University	49
26	2005-09-24	Atlanta	Florida A&M University	50
82	2010-09-25	Atlanta	Florida A&M University	48
130	2014-09-27	Nashville	Florida A&M University	44
137	2014-11-22	Murray	Murray State University	51
146	2015-11-07	Nashville	Murray State University	55

In [441…
```
#replace outliers of TSU sack yards with mean
mydata["TSUsackyd_new"]=np.where((mydata["TSUsackyd"]>outlier_top_lim) | (mydata["TSUsackyd"]<outlier_bottom_li
```

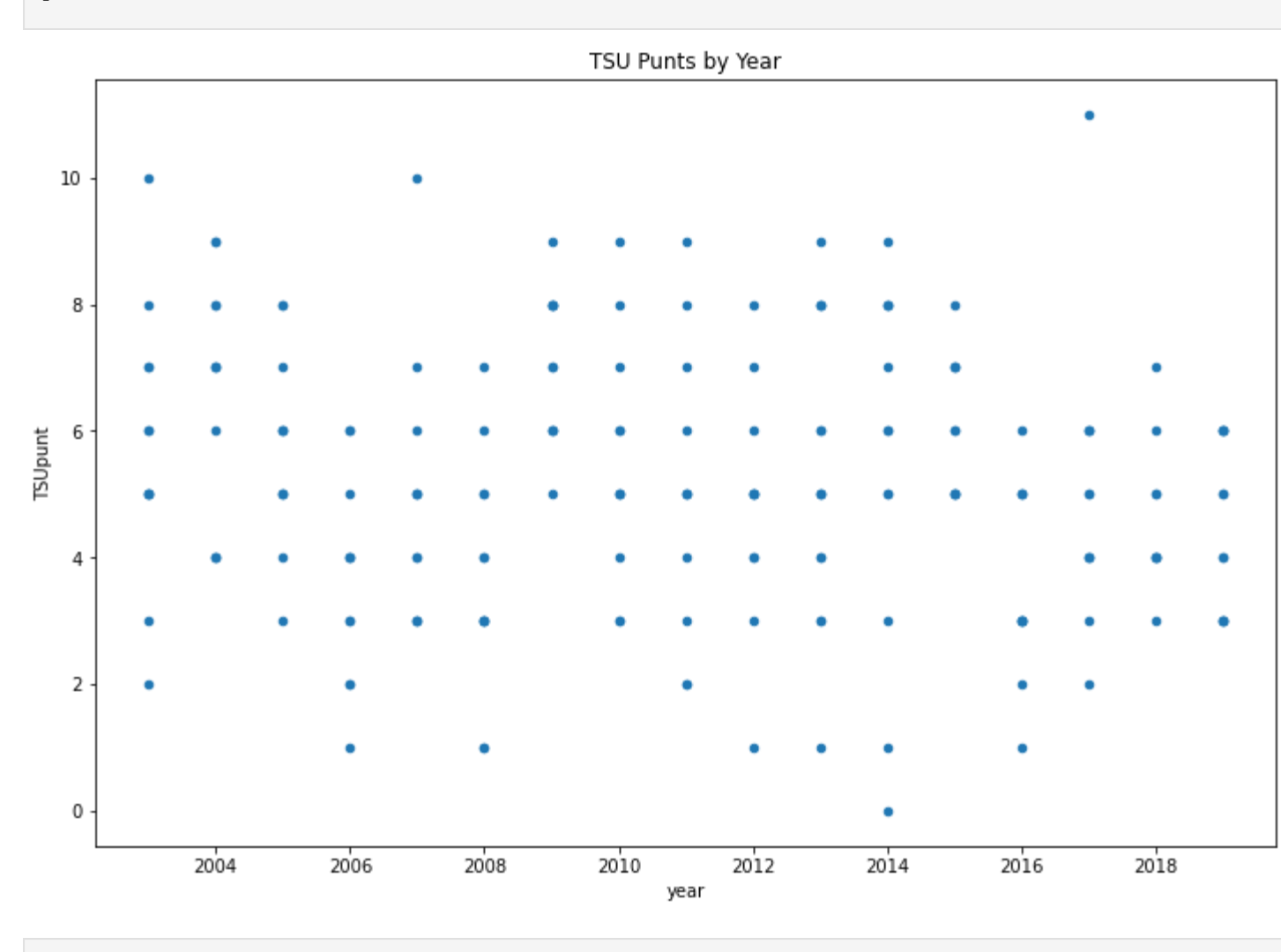
In [442…
```
#descriptive statistics for TSU sack yards
mydata.TSUsackyd.describe()
```

Out[442…
```
count    191.000000
mean      13.534031
std       11.944946
min        0.000000
25%        5.000000
50%       11.000000
75%       20.000000
max       55.000000
Name: TSUsackyd, dtype: float64
```

In [443…
```
#descriptive statistics for TSU sack yards with outliers replaced
#Replacing these outliers, caused the mean of TSU sack yards to drop to 12.4 while the standard deviation dropp
mydata.TSUsackyd_new.describe()
```

Out[443…
```
count    191.000000
mean      12.404210
std       10.010235
min        0.000000
25%        5.000000
50%       11.000000
75%       18.000000
max       40.000000
Name: TSUsackyd_new, dtype: float64
```

In [444…
```
#TSU punts by year
#A scatter plot of TSU punts shows that for most games, TSU had between 2 and 8 punts. Outliers wree assessed b
plt.rcParams['figure.figsize']=[12,8]
mydata.plot(x="year",y="TSUpunt",kind="scatter", title="TSU Punts by Year")
plt.show()
```



In [445…
```
#get outliers of TSU punt
#There were 4 outliers on the TSU punt variable. They occurred in a game against Austin Peay State University
#a game against Jacksonville State University (11) and in 2 games against Eastern Kentucky University (10 each)
#These outliers were replaced with the mean of the variable.
q1 = mydata["TSUpunt"].quantile(0.25)
q3 = mydata["TSUpunt"].quantile(0.75)
outlier_top_lim = q3 + 1.5 * (q3 - q1)
outlier_bottom_lim = q1 - 1.5 * (q3 - q1)
mydata[["date","city","opponent","TSUpunt"]][(mydata["TSUpunt"]>outlier_top_lim) | (mydata["TSUpunt"]<outlier_b
```

Out[445…

	date	city	opponent	TSUpunt
10	2003-11-15	Nashville	Eastern Kentucky University	10
51	2007-10-20	Richmond	Eastern Kentucky University	10
136	2014-11-08	Clarksville	Austin Peay State University	0
169	2017-11-16	Jacksonville	Jacksonville State University	11

In [446…
```
#replace outliers of TSU punts with mean
mydata["TSUpunt_new"]=np.where((mydata["TSUpunt"]>outlier_top_lim) | (mydata["TSUpunt"]<outlier_bottom_lim),myd
```

In [447…
```
#descriptive statistics for TSU punts
mydata.TSUpunt.describe()
```

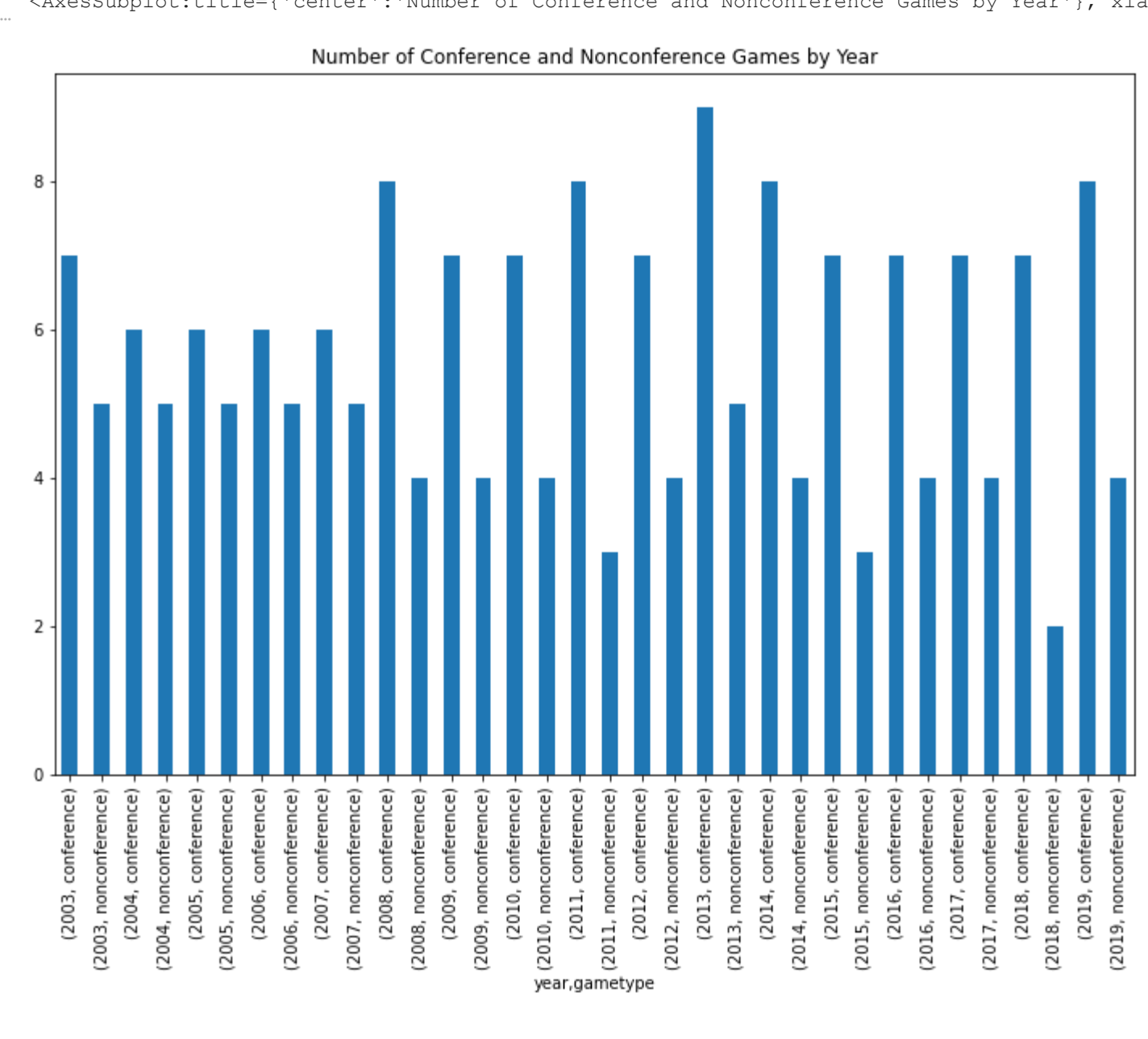
Out[447…
```
count    191.000000
mean       5.162304
std        2.064702
min        0.000000
25%        4.000000
50%        5.000000
75%        6.000000
max       11.000000
Name: TSUpunt, dtype: float64
```

In [448…
```
#descriptive statistics for TSU punts with outliers replaced
#Replacing the outliers made neglibile difference in the mean and standard deviation for the TSU punts variable
mydata.TSUpunt_new.describe()
```

Out[448…
```
count    191.000000
mean       5.108111
std        1.921996
min        1.000000
25%        4.000000
50%        5.000000
75%        6.000000
max        9.000000
Name: TSUpunt_new, dtype: float64
```

In [449…
```
#number of games by game type by year
#The number of conference and nonconference games varied by year. TSU had more conference games than nonconfere
#each year from 2003 to 2019.From 2004 to 2007, TSU had 6 conference games and 5 nonconference games.
#TSU had 7 conference and 4 nonconference games in 2009, 2010, 2012, 2016, 2017, and 2018.
#In 2013, TSU had the most conference games (9).
plt.rcParams['figure.figsize']=[12,8]
gametype_by_year=mydata.groupby("year")["gametype"].value_counts()
gametype_by_year.plot(kind="bar",title="Number of Conference and Nonconference Games by Year")
```

Out[449…
```
<AxesSubplot:title={'center':'Number of Conference and Nonconference Games by Year'}, xlabel='year,gametype'>
```



More data wrangling to prep for machine learning

In [450…
```
#creating new file with variables with outliers replaced
mydata_1=mydata
#saving new file with variables (not saving mydata again to maintain original dataset)
mydata_1.to_csv("mydata_1.csv",encoding='utf-8')
```

In [451…
```
#checking independence of features (all variables except winloss)
#reading in data with updated variables
mydata_1=pd.read_csv("mydata_1.csv")
#get rid of unnamed 1st variable that's just the index put on when file was saved to a csv file
mydata_1=mydata_1.iloc[:,1:]
mydata_1.columns
```

Out[451…
```
Index(['date', 'city', 'state', 'winscore', 'losscore', 'loser', 'winner',
       'locale', 'TSU score', 'opponent score', 'scorediff', 'scorediff_abs',
       'winloss', 'attendance', 'TSUrushyards', 'TSUreceiveyards',
       'TSUkreturnyards', 'TSUpreturnyards', 'TSUtackles', 'TSUtackleyd',
       'TSUsacks', 'TSUsackyd', 'TSUpunt', 'opponent', 'gametype', 'year',
       'TSU score_new', 'opponent score_new', 'scorediff_new',
       'scorediff_abs_new', 'attendance_new', 'TSUrushyards_new',
       'TSUreceiveyards_new', 'TSUkreturnyards_new', 'TSUpreturnyards_new',
       'TSUtackleyd_new', 'TSUsacks_new', 'TSUsackyd_new', 'TSUpunt_new'],
      dtype='object')
```

In [452…
```
#create binary variables to use later in machine learning below
mydata_1["locale_01"]=np.where(mydata_1["locale"]=="Home",1,0)
mydata_1["gametype_01"]=np.where(mydata_1["gametype"]=="conference",1,0)
```

In [453…
```
#check columns
mydata_1.columns
```

Out[453…
```
Index(['date', 'city', 'state', 'winscore', 'losscore', 'loser', 'winner',
       'locale', 'TSU score', 'opponent score', 'scorediff', 'scorediff_abs',
       'winloss', 'attendance', 'TSUrushyards', 'TSUreceiveyards',
       'TSUkreturnyards', 'TSUpreturnyards', 'TSUtackles', 'TSUtackleyd',
       'TSUsacks', 'TSUsackyd', 'TSUpunt', 'opponent', 'gametype', 'year',
       'TSU score_new', 'opponent score_new', 'scorediff_new',
       'scorediff_abs_new', 'attendance_new', 'TSUrushyards_new',
       'TSUreceiveyards_new', 'TSUkreturnyards_new', 'TSUpreturnyards_new',
       'TSUtackleyd_new', 'TSUsacks_new', 'TSUsackyd_new', 'TSUpunt_new',
       'locale_01', 'gametype_01'],
      dtype='object')
```

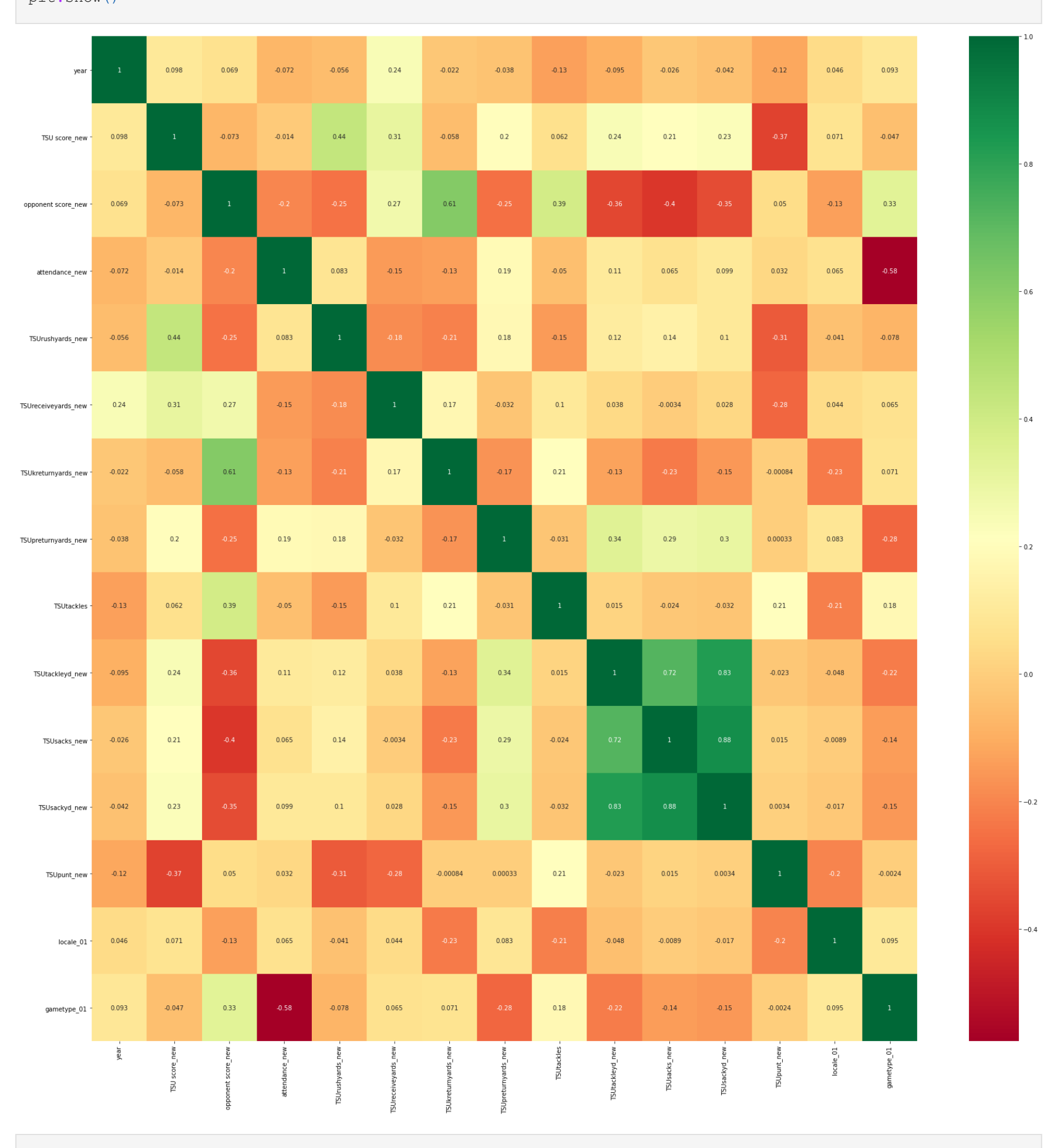
In [454…
```
#narrow down mydata_1 to variables needed for analysis dropping unneeded variables
mydata_1=mydata_1[['year','TSU score_new', 'opponent score_new',
        'attendance_new', 'TSUrushyards_new','TSUreceiveyards_new', 'TSUkreturnyards_new',
        'TSUpreturnyards_new', 'TSUtackles','TSUtackleyd_new', 'TSUsacks_new',
        'TSUsackyd_new', 'TSUpunt_new','locale_01','winloss','gametype_01']]
mydata_1.columns
```

Out[454…
```
Index(['year', 'TSU score_new', 'opponent score_new', 'attendance_new',
       'TSUrushyards_new', 'TSUreceiveyards_new', 'TSUkreturnyards_new',
       'TSUpreturnyards_new', 'TSUtackles', 'TSUtackleyd_new', 'TSUsacks_new',
       'TSUsackyd_new', 'TSUpunt_new', 'locale_01', 'winloss', 'gametype_01'],
      dtype='object')
```

In [455…
```
#get info on dataset
mydata_1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 191 entries, 0 to 190
Data columns (total 16 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   year                 191 non-null    int64
 1   TSU score_new        191 non-null    float64
 2   opponent score_new   191 non-null    float64
 3   attendance_new       191 non-null    float64
 4   TSUrushyards_new     191 non-null    float64
 5   TSUreceiveyards_new  191 non-null    float64
 6   TSUkreturnyards_new  191 non-null    float64
 7   TSUpreturnyards_new  191 non-null    float64
 8   TSUtackles           191 non-null    int64
 9   TSUtackleyd_new      191 non-null    float64
 10  TSUsacks_new         191 non-null    float64
 11  TSUsackyd_new        191 non-null    float64
 12  TSUpunt_new          191 non-null    float64
 13  locale_01            191 non-null    int32
 14  winloss              191 non-null    object
 15  gametype_01          191 non-null    int32
dtypes: float64(11), int32(2), int64(2), object(1)
memory usage: 22.5+ KB
```

In [456…
```
#correlation heatmap
cormat=mydata_1.corr()
feature_ind=cormat.index
plt.figure(figsize=(30,30))
sns.heatmap(mydata_1[feature_ind].corr(),
            annot=True,cmap="RdYlGn")
plt.show()
```



In [457…
```
#printing correlation matrix to see correlation coefficients better
print(cormat)
```

```
                         year  TSU score_new  opponent score_new  \
year                 1.000000       0.098034            0.069488
TSU score_new        0.098034       1.000000           -0.073290
opponent score_new   0.069488      -0.073290            1.000000
attendance_new      -0.071744      -0.014115           -0.199377
TSUrushyards_new    -0.056248       0.435246           -0.248612
TSUreceiveyards_new  0.238479       0.307372            0.273069
TSUkreturnyards_new -0.022121      -0.058038            0.607815
TSUpreturnyards_new -0.037710       0.201383           -0.253238
TSUtackles          -0.134393       0.061882            0.389535
TSUtackleyd_new     -0.094670       0.238959           -0.358462
TSUsacks_new        -0.026175       0.207029           -0.401664
TSUsackyd_new       -0.042450       0.228340           -0.346219
TSUpunt_new         -0.118423      -0.369115            0.049828
locale_01            0.045810       0.071018           -0.133741
gametype_01          0.092807      -0.046550            0.329552

                     attendance_new  TSUrushyards_new  TSUreceiveyards_new  \
year                      -0.071744         -0.056248             0.238479
TSU score_new             -0.014115          0.435246             0.307372
opponent score_new        -0.199377         -0.248612             0.273069
attendance_new             1.000000          0.082653            -0.147965
TSUrushyards_new           0.082653          1.000000            -0.182135
TSUreceiveyards_new       -0.147965         -0.182135             1.000000
TSUkreturnyards_new       -0.133523         -0.210861             0.173211
TSUpreturnyards_new        0.186817          0.177903            -0.031669
TSUtackles                -0.049826         -0.147586             0.100492
TSUtackleyd_new            0.105645          0.115734             0.037608
TSUsacks_new               0.064635          0.142303            -0.003381
TSUsackyd_new              0.098668          0.101176             0.027815
TSUpunt_new                0.032203         -0.306926            -0.278999
locale_01                  0.065018         -0.041005             0.043982
gametype_01               -0.575838         -0.077752             0.065084

                     TSUkreturnyards_new  TSUpreturnyards_new  TSUtackles  \
year                           -0.022121            -0.037710   -0.134393
TSU score_new                  -0.058038             0.201383    0.061882
opponent score_new              0.607815            -0.253238    0.389535
attendance_new                 -0.133523             0.186817   -0.049826
TSUrushyards_new               -0.210861             0.177903   -0.147586
TSUreceiveyards_new             0.173211            -0.031669    0.100492
TSUkreturnyards_new             1.000000            -0.168115    0.209326
TSUpreturnyards_new            -0.168115             1.000000   -0.031371
TSUtackles                      0.209326            -0.031371    1.000000
TSUtackleyd_new                -0.127686             0.337636    0.015167
TSUsacks_new                   -0.226563             0.288105   -0.024418
TSUsackyd_new                  -0.154778             0.300919   -0.032026
TSUpunt_new                    -0.000845             0.000326    0.208592
locale_01                      -0.230626             0.082838   -0.213631
gametype_01                     0.070591            -0.278124    0.176882

                     TSUtackleyd_new  TSUsacks_new  TSUsackyd_new  \
year                       -0.094670     -0.026175      -0.042450
TSU score_new               0.238959      0.207029       0.228340
opponent score_new         -0.358462     -0.401664      -0.346219
attendance_new              0.105645      0.064635       0.098668
TSUrushyards_new            0.115734      0.142303       0.101176
TSUreceiveyards_new         0.037608     -0.003381       0.027815
TSUkreturnyards_new        -0.127686     -0.226563      -0.154778
TSUpreturnyards_new         0.337636      0.288105       0.300919
TSUtackles                  0.015167     -0.024418      -0.032026
TSUtackleyd_new             1.000000      0.720046       0.825525
TSUsacks_new                0.720046      1.000000       0.875692
TSUsackyd_new               0.825525      0.875692       1.000000
TSUpunt_new                -0.022511      0.015281       0.003401
locale_01                  -0.048195     -0.008888      -0.017292
gametype_01                -0.220558     -0.143653      -0.152191

                     TSUpunt_new  locale_01  gametype_01
year                   -0.118423   0.045810     0.092807
TSU score_new          -0.369115   0.071018    -0.046550
opponent score_new      0.049828  -0.133741     0.329552
attendance_new          0.032203   0.065018    -0.575838
TSUrushyards_new       -0.306926  -0.041005    -0.077752
TSUreceiveyards_new    -0.278999   0.043982     0.065084
TSUkreturnyards_new    -0.000845  -0.230626     0.070591
TSUpreturnyards_new     0.000326   0.082838    -0.278124
TSUtackles              0.208592  -0.213631     0.176882
TSUtackleyd_new        -0.022511  -0.048195    -0.220558
TSUsacks_new            0.015281  -0.008888    -0.143653
TSUsackyd_new           0.003401  -0.017292    -0.152191
TSUpunt_new             1.000000  -0.201983    -0.002450
locale_01              -0.201983   1.000000     0.095127
gametype_01            -0.002450   0.095127     1.000000
```

In [458…
```
#Get pairs with moderate to high correlations
#create matrix with absolute values of correlatinos
cormat_abs=mydata_1.corr().abs()
#unstack matrix so all correlations are listed on top of each other instead of a matrix
s = cormat_abs.unstack()
#sort correlations in ascending order
so = s.sort_values(ascending=True)
#get all correlations with absolute value of at least .5
print(so[so>=.5])
#These results show 5 pairs of features that are at least moderately correlated (absolute value of r >.=5). The
#game type (conference/nonconference) and attendance; opponent score and TSU kick return yards;
#TSU tackle yards and TSU sacks; TSU tackle yards and TSU sack yards; and TSU sacks and TSU sack yards.
```

```
gametype_01          attendance_new         0.575838
attendance_new       gametype_01            0.575838
opponent score_new   TSUkreturnyards_new    0.607815
TSUkreturnyards_new  opponent score_new     0.607815
TSUtackleyd_new      TSUsacks_new           0.720046
TSUsacks_new         TSUtackleyd_new        0.720046
TSUtackleyd_new      TSUsackyd_new          0.825525
TSUsackyd_new        TSUtackleyd_new        0.825525
                     TSUsacks_new           0.875692
TSUsacks_new         TSUsackyd_new          0.875692
year                 year                   1.000000
TSUpreturnyards_new  TSUpreturnyards_new    1.000000
TSUpunt_new          TSUpunt_new            1.000000
TSUsackyd_new        TSUsackyd_new          1.000000
TSUsacks_new         TSUsacks_new           1.000000
TSUtackleyd_new      TSUtackleyd_new        1.000000
TSUtackles           TSUtackles             1.000000
TSUkreturnyards_new  TSUkreturnyards_new    1.000000
TSUreceiveyards_new  TSUreceiveyards_new    1.000000
TSUrushyards_new     TSUrushyards_new       1.000000
attendance_new       attendance_new         1.000000
opponent score_new   opponent score_new     1.000000
TSU score_new        TSU score_new          1.000000
locale_01            locale_01              1.000000
gametype_01          gametype_01            1.000000
dtype: float64
```

In [459…
```
#save file with features and outcome variable only
mydata_1.to_csv("mydata_1.csv",encoding="utf-8")
```

Machine learning with a decision tree

In [460…
```
#read in file for machine learning
mydata_1=pd.read_csv("mydata_1.csv")
#drop 1st column
mydata_1=mydata_1.iloc[:,1:]
mydata_1.head()
```

Out[460…

	year	TSU score_new	opponent score_new	attendance_new	TSUrushyards_new	TSUreceiveyards_new	TSUkreturnyards_new	TSUpreturnyards_new	TSUtack
0	2003	37.0	20.0	18124.000000	113.0	365.0	69.0	18.753927	
1	2003	24.0	31.0	18085.000000	141.0	146.0	147.0	8.000000	
2	2003	44.0	14.0	16985.230366	209.0	132.0	138.0	18.753927	
3	2003	7.0	10.0	16985.230366	153.0	138.0	86.0	2.000000	
4	2003	41.0	10.0	8434.000000	142.0	285.0	11.0	29.000000	

In [461…
```
#data split for test and training data sets
#create separate datasets for independent and dependent variables
x = mydata_1.drop('winloss', axis = 1).values
y =mydata_1['winloss'].values

#split x (features) and y (outcome) datasets with 30% of data in test dataset
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.3, random_state =0)
```

In [462…
```
#create decision tree model
tree_model = DecisionTreeClassifier(max_depth = 7, criterion = 'entropy', random_state=0)
#fit model to training data
tree_model.fit(x_train, y_train)
#use model to predict on test data
tree_yhat = tree_model.predict(x_test)
```

In [463…
```
#Accuracy score (compare predicted values based on test data (tree_yhat) with actual outcomes in test data)
print('Accuracy score of the Decision Tree model is {}'.format(accuracy_score(y_test, tree_yhat)))
#The accuracy score of the decision tree model is about .76 or 76% which means that the model is good.
```

```
Accuracy score of the Decision Tree model is 0.7586206896551724
```

In [464…
```
#print decision tree
#create labels for features
fn=['year', 'TSU score_new', 'opponent score_new', 'attendance_new',
       'TSUrushyards_new', 'TSUreceiveyards_new', 'TSUkreturnyards_new',
       'TSUpreturnyards_new', 'TSUtackles', 'TSUtackleyd_new', 'TSUsacks_new',
       'TSUsackyd_new', 'TSUpunt_new', 'locale_01', 'gametype_01', ]
cn=['Loss','Win']
#plot tree
plt.rcParams['figure.figsize']=[12,12]
tree.plot_tree(tree_model, feature_names=fn,class_names=cn, filled=True)
```