Project Name: Storm is Coming.. Are You Safe!?

Member 1: **Name:** Huzefa Saifee **UID:** u1274086 **Email:** u1274086@utah.edu
Member 2: **Name:** Siddharth Hatkar **UID:** u1273107 **Email:** u1273107@utah.edu
Member 3: **Name:** Soumith Reddy Mada **UID:** u1288778 **Email:** u1288778@utah.edu

Background and Motivation:

Predicting the intensity of a storm by different weather forecasts is a boon which helps us predict what kind of storm is going to hit and by what intensity, so that we can take preventive measures for such storms beforehand.

However, predicting the intensities of storms on the basis of damage and the casualties caused by them can help in taking better preventive measures for different types of storms.

In 2019 alone, the Atlantic storms caused damage of approximately $22 billion in America, according to a report by AccuWeather.

On an average:

- 80 deaths and 1,500 injuries each year are directly attributed to thousands of tornadoes reported. (source: www.nssl.noaa.gov)
- 31 deaths per year are caused by thunderstorm winds; Number of thunderstorms occurring in the United States a year: 100,000.

Project objective:

Our goal is to Analyze different types of storms in different parts of America based on historical data gathered from the National Centers for Environmental Information datasets of the past 20 years. Also, analyzing the deaths caused by different storms in a geospatial way.

The most important questions that arise when we hear about Storms are:

- What was the number of casualties due to that storm?
- What was the damage caused because of the storm?

With this project, we are trying to convert these *"was"* questions into the prediction of the damage that *"will be"* caused by the storm, i.e. We are trying to predict the damage caused by the Storm in any location based on the multiple factors like Magnitude of the Storms.

In [ ]:
```
# imports and setup
import pandas as pd
import scipy as sc
import numpy as np
import seaborn as sns

#%matplotlib notebook
import matplotlib.pyplot as plt
plt.style.use('ggplot')
%matplotlib inline
plt.rcParams['figure.figsize'] = (10, 6)
```

Data:

We have gathered the data from the following website:
https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/?C=M;O=D (https://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/?C=M;O=D)

It has a dataset of storms that occurred in the USA every year (for 2019, 2018, 2017...).

Our goal is to merge all of the different datasets of the different year into one single dataset and perform descriptive analysis, and predictive analysis of the storm data generated.

The Metadata of the columns is described in the following link:
http://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/Storm-Data-Export-Format.docx
(http://www1.ncdc.noaa.gov/pub/data/swdi/stormevents/csvfiles/Storm-Data-Export-Format.docx)

In [288]:
```
# Data Retrieval
storm = pd.read_csv("StormEvents_2019details.csv")
storm.head()
storm.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64439 entries, 0 to 64438
Data columns (total 51 columns):
BEGIN_YEARMONTH       64439 non-null int64
BEGIN_DAY             64439 non-null int64
BEGIN_TIME            64439 non-null int64
END_YEARMONTH         64439 non-null int64
END_DAY               64439 non-null int64
END_TIME              64439 non-null int64
EPISODE_ID            64439 non-null int64
EVENT_ID              64439 non-null int64
STATE                 64439 non-null object
STATE_FIPS            64439 non-null int64
YEAR                  64439 non-null int64
MONTH_NAME            64439 non-null object
EVENT_TYPE            64439 non-null object
CZ_TYPE               64439 non-null object
CZ_FIPS               64439 non-null int64
CZ_NAME               64439 non-null object
WFO                   64439 non-null object
BEGIN_DATE_TIME       64439 non-null object
CZ_TIMEZONE           64439 non-null object
END_DATE_TIME         64439 non-null object
INJURIES_DIRECT       64439 non-null int64
INJURIES_INDIRECT     64439 non-null int64
DEATHS_DIRECT         64439 non-null int64
DEATHS_INDIRECT       64439 non-null int64
DAMAGE_PROPERTY       52112 non-null object
DAMAGE_CROPS          52558 non-null object
SOURCE                64439 non-null object
MAGNITUDE             35031 non-null float64
MAGNITUDE_TYPE        26025 non-null object
FLOOD_CAUSE           8946 non-null object
CATEGORY              10 non-null float64
TOR_F_SCALE           1649 non-null object
TOR_LENGTH            1649 non-null float64
TOR_WIDTH             1649 non-null float64
TOR_OTHER_WFO         194 non-null object
TOR_OTHER_CZ_STATE    194 non-null object
TOR_OTHER_CZ_FIPS     194 non-null float64
TOR_OTHER_CZ_NAME     194 non-null object
BEGIN_RANGE           42858 non-null float64
BEGIN_AZIMUTH         42858 non-null object
BEGIN_LOCATION        42858 non-null object
END_RANGE             42858 non-null float64
END_AZIMUTH           42858 non-null object
END_LOCATION          42858 non-null object
BEGIN_LAT             42858 non-null float64
BEGIN_LON             42858 non-null float64
END_LAT               42858 non-null float64
END_LON               42858 non-null float64
EPISODE_NARRATIVE     64439 non-null object
EVENT_NARRATIVE       50629 non-null object
DATA_SOURCE           64439 non-null object
dtypes: float64(11), int64(15), object(25)
memory usage: 25.1+ MB
```

Ethical Consideration:

Data collected or recorded are accurate. We are performing this analysis only for America, as the number of storm cases in America contributes to 40% of the entire world.

The data is openly available at https://www.ncdc.noaa.gov/stormevents/ftp.jsp (https://www.ncdc.noaa.gov/stormevents/ftp.jsp) by the government of America for public interest and we have verified that anyone is allowed to access and perform ethical data analysis.

Since we are working on a dataset of different types of Storms, which are Natural Disasters and is not in control of humankind, the risk of violating ethical consideration reduces to a great extent. The only bias our models might have is of showing more storms in the areas which are prone to that kind of storms and that is a valid bias as well.

We also made sure that our data doesn't have any sort of bias that might impair any individual or community. Therefore, we strongly believe that our data is ethical and our analysis will help mankind.

Data Processing:

For this proposal, we have considered only the 2019 dataset and as stated above, for the submission of our final project, we will try to get data for at least 20 years.

We have created new fields like Tot_Deaths & Tot_Injuries by adding columns of direct and indirect deaths and injuries to get a better initial understanding of the effects of the storms.

In [292]:
```
# Addition of Deaths & Injuries into one new Column
storm['Tot_Deaths'] = storm['DEATHS_DIRECT'] + storm['DEATHS_INDIRECT']
storm['Tot_Injures'] = storm['INJURIES_DIRECT'] + storm['INJURIES_INDIRECT']
```

Exploratory Data Analysis:

After performing a final EDA, we might find a few columns that have outliers and needs to be removed before further analysis.

We have also converted the “Damage Property” column from string datatype to int (1k to 1000).

We found a few columns with null values and further research is required whether we should remove those values or whether we are supposed to perform null value imputation on them.
Also, we can remove a few columns because they might not be significant in performing analysis or they might are highly correlated with other columns in the dataset.

In [310]:
```
storm.describe()
```

Out[310]:

	BEGIN_YEARMONTH	BEGIN_DAY	BEGIN_TIME	END_YEARMONTH	END_DAY	END_TIME	EPISODE_ID	EVENT_ID	STATE_FI
count	52112.000000	52112.000000	52112.000000	52112.000000	52112.000000	52112.000000	52112.000000	52112.000000	52112.0000
mean	201905.644170	16.240732	1291.740290	201905.644170	17.309276	1426.107921	138112.086640	830239.033006	32.4976
std	2.804604	8.821335	656.440183	2.804604	8.756046	620.178898	3222.463571	20202.124489	18.7755
min	201901.000000	1.000000	0.000000	201901.000000	1.000000	0.000000	132253.000000	791304.000000	1.0000
25%	201903.000000	9.000000	830.000000	201903.000000	10.000000	1016.000000	135439.000000	813159.750000	19.0000
50%	201906.000000	17.000000	1450.000000	201906.000000	18.000000	1545.000000	137824.000000	830208.500000	31.0000
75%	201908.000000	23.000000	1800.000000	201908.000000	25.000000	1900.000000	140807.000000	847617.250000	46.0000
max	201911.000000	31.000000	2359.000000	201911.000000	31.000000	2359.000000	144628.000000	869176.000000	99.0000

8 rows × 29 columns

In [311]:
```
# Outlier Suspection Method
storm.TOR_LENGTH.describe()
```

Out[311]:
```
count    1430.000000
mean        2.929126
std         3.821778
min         0.010000
25%         0.500000
50%         1.445000
75%         3.987500
max        31.400000
Name: TOR_LENGTH, dtype: float64
```

In [301]:
```
# Coverting Column Datatype (String to Integer)
for i in range(0, len(storm.DAMAGE_PROPERTY)):
    if(type(storm.DAMAGE_PROPERTY[i]) == str):
        if ('K' in storm.DAMAGE_PROPERTY[i]):
            storm.DAMAGE_PROPERTY[i] = storm.DAMAGE_PROPERTY[i].replace("K","")
            storm.DAMAGE_PROPERTY[i] =int(float(storm.DAMAGE_PROPERTY[i])*1000)
        elif('M' in storm.DAMAGE_PROPERTY[i]):
            storm.DAMAGE_PROPERTY[i] = storm.DAMAGE_PROPERTY[i].replace("M","")
            storm.DAMAGE_PROPERTY[i] =int(float(storm.DAMAGE_PROPERTY[i])*1000000)
        elif('B' in storm.DAMAGE_PROPERTY[i]):
            storm.DAMAGE_PROPERTY[i] = storm.DAMAGE_PROPERTY[i].replace("B","")
            storm.DAMAGE_PROPERTY[i] =int(float(storm.DAMAGE_PROPERTY[i])*1000000000)
```

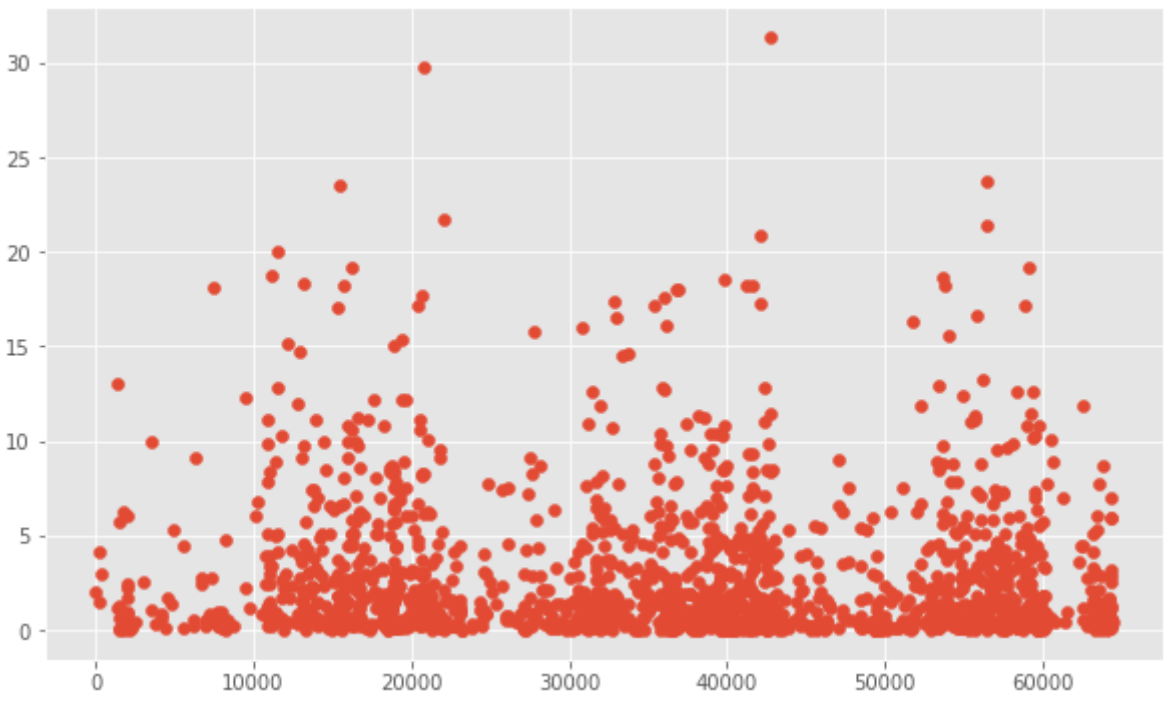
```
C:\Users\madas\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  after removing the cwd from sys.path.
C:\Users\madas\Anaconda3\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """
C:\Users\madas\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  import sys
C:\Users\madas\Anaconda3\lib\site-packages\ipykernel_launcher.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

C:\Users\madas\Anaconda3\lib\site-packages\ipykernel_launcher.py:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  # Remove the CWD from sys.path while we load stuff.
C:\Users\madas\Anaconda3\lib\site-packages\ipykernel_launcher.py:11: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  # This is added back by InteractiveShellApp.init_path()
```

In [313]:
```
# Removing NAs
storm = storm[storm['DAMAGE_PROPERTY'].notna()]
storm.DAMAGE_PROPERTY = storm.DAMAGE_PROPERTY.astype(int)
```

Data Visualization

In [291]:
```
# Outlier Detection Method
x = storm.TOR_LENGTH.index.values
plt.scatter(x, storm.TOR_LENGTH)
```

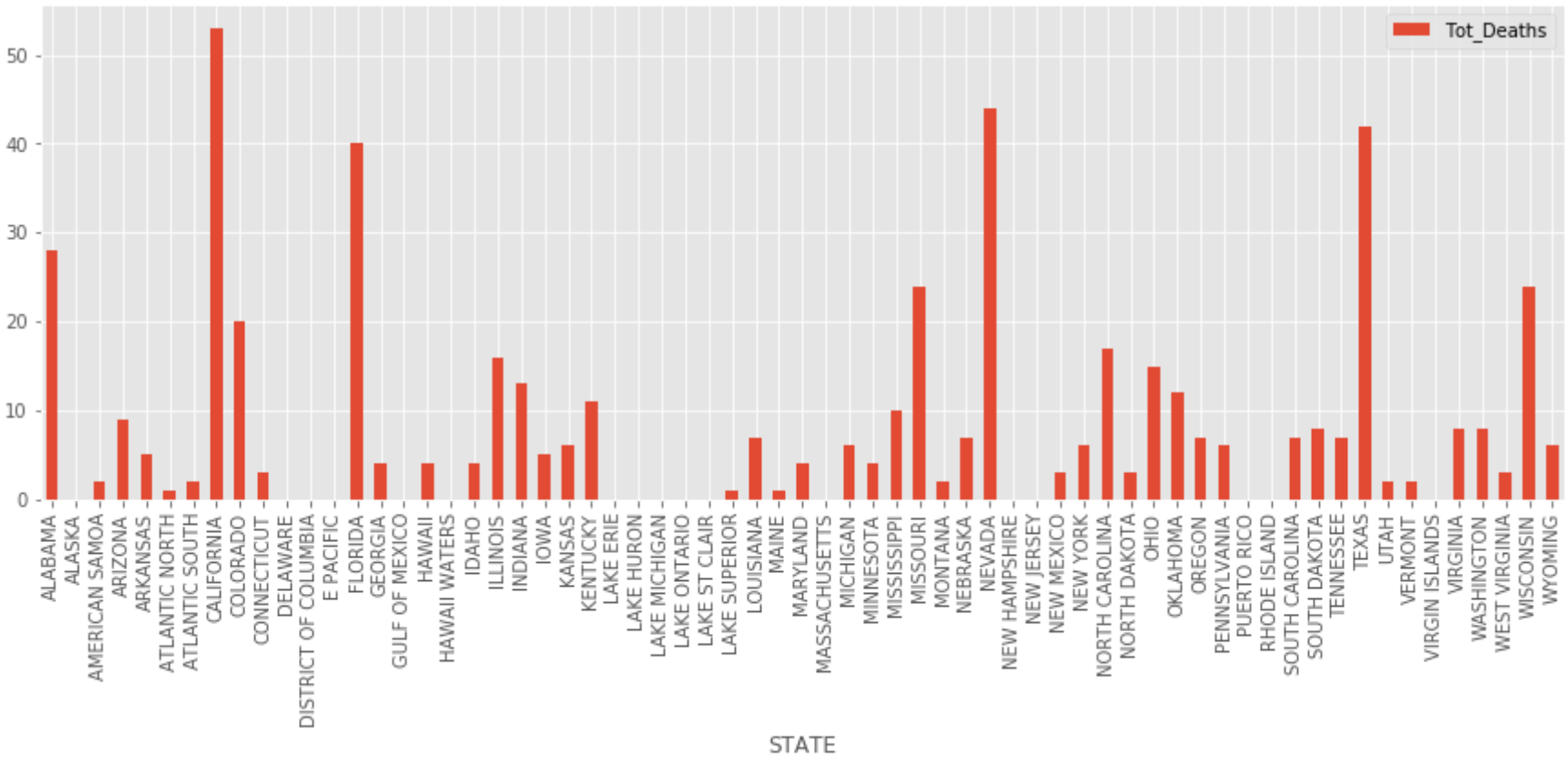
Out[291]: <matplotlib.collections.PathCollection at 0x1c68e7fbb48>



In [293]:
```
# Group by states
pd.set_option('display.max_rows',65)
storm_dea = storm.groupby('STATE').agg({'Tot_Deaths':'sum'})
#storm.plot(x='STATE', y='Tot_Deaths', kind="bar")
```

In [294]:
```
storm_dea.plot(kind="bar",figsize=(15,5))
```

Out[294]: <matplotlib.axes._subplots.AxesSubplot at 0x1c68cd44188>



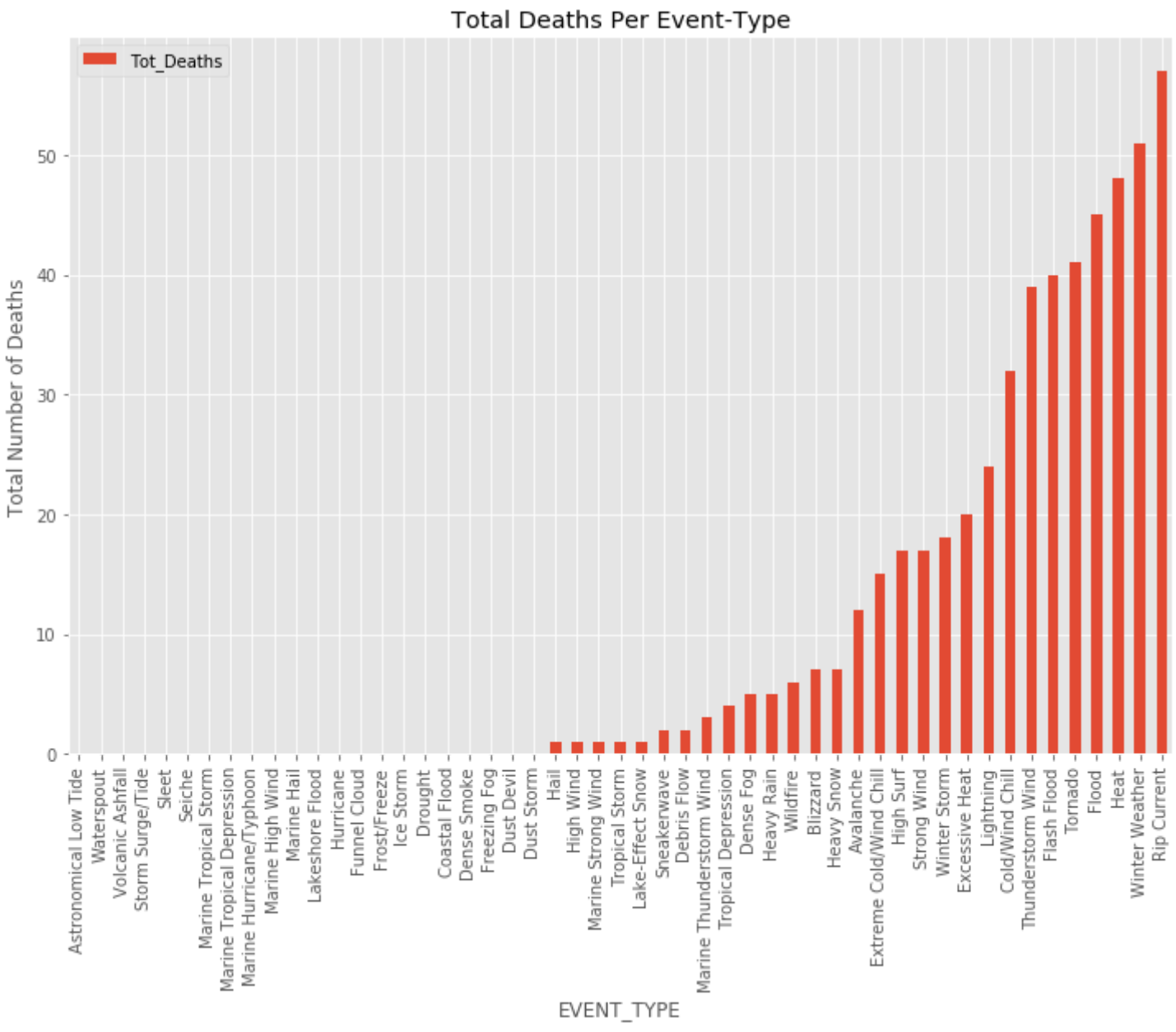
In [296]:
```
#Group by states
storm_type_dea = storm.groupby('EVENT_TYPE').agg({'Tot_Deaths':'sum'})
storm_type_dea
```

Out[296]:

EVENT_TYPE	Tot_Deaths
Astronomical Low Tide	0
Avalanche	12
Blizzard	7
Coastal Flood	0
Cold/Wind Chill	32
Debris Flow	2
Dense Fog	5
Dense Smoke	0
Drought	0
Dust Devil	0
Dust Storm	0
Excessive Heat	20
Extreme Cold/Wind Chill	15
Flash Flood	40
Flood	45
Freezing Fog	0
Frost/Freeze	0
Funnel Cloud	0
Hail	1
Heat	48
Heavy Rain	5
Heavy Snow	7
High Surf	17
High Wind	1
Hurricane	0
Ice Storm	0
Lake-Effect Snow	1
Lakeshore Flood	0
Lightning	24
Marine Hail	0
Marine High Wind	0
Marine Hurricane/Typhoon	0
Marine Strong Wind	1
Marine Thunderstorm Wind	3
Marine Tropical Depression	0
Marine Tropical Storm	0
Rip Current	57
Seiche	0
Sleet	0
Sneakerwave	2
Storm Surge/Tide	0
Strong Wind	17
Thunderstorm Wind	39
Tornado	41
Tropical Depression	4
Tropical Storm	1
Volcanic Ashfall	0
Waterspout	0
Wildfire	6
Winter Storm	18
Winter Weather	51

In [317]:
```
storm_type_dea.sort_values(by='Tot_Deaths',ascending=True).plot(kind="bar",figsize=(12,8))
plt.title("Total Deaths Per Event-Type")
plt.ylabel("Total Number of Deaths")
```

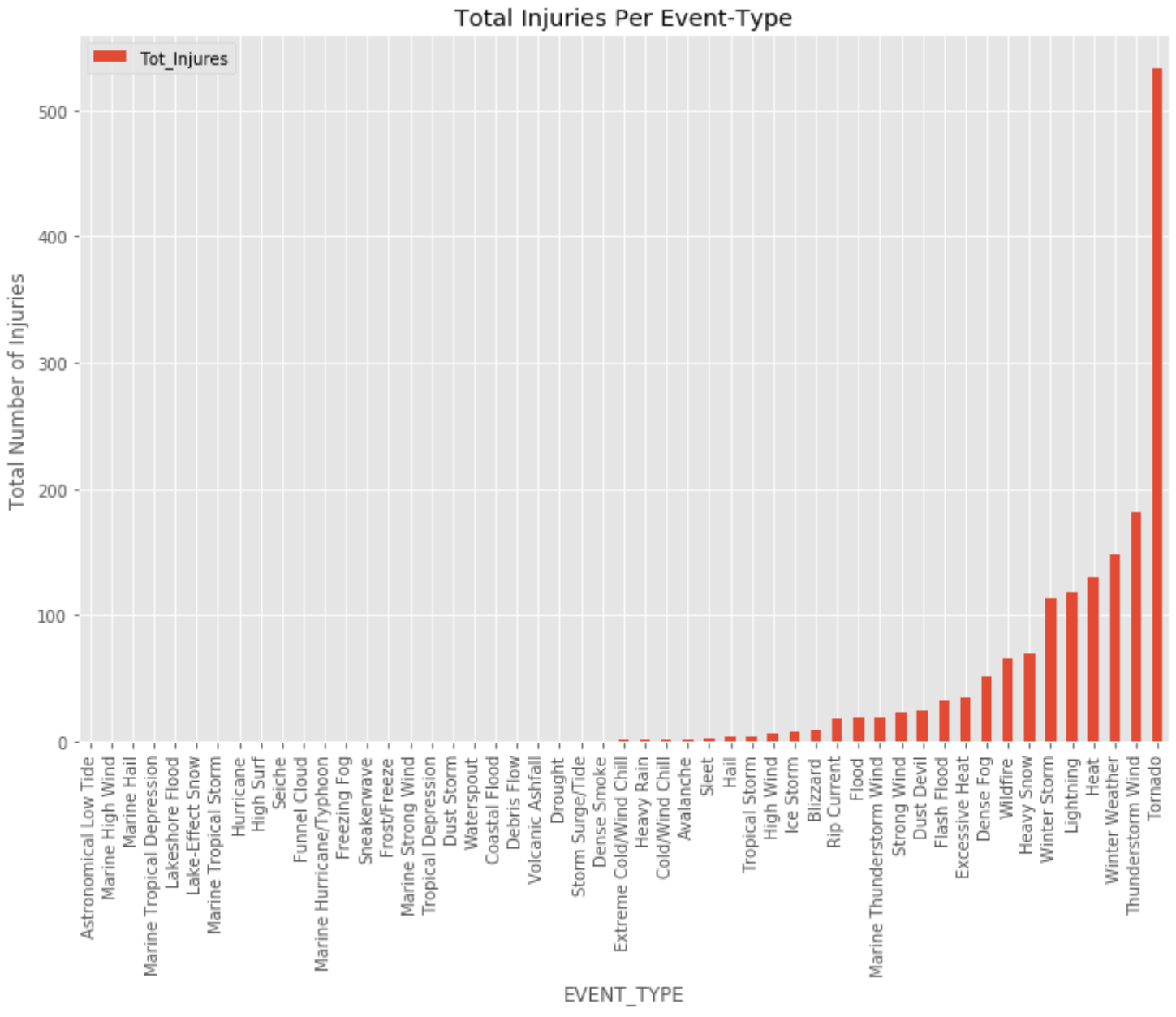
Out[317]: Text(0, 0.5, 'Total Number of Deaths')



In [298]:
```
storm_type_inj = storm.groupby('EVENT_TYPE').agg({'Tot_Injures':'sum'})
```

In [318]:
```
storm_type_inj.sort_values(by='Tot_Injures',ascending=True).plot(kind="bar",figsize=(12,8))
plt.title("Total Injuries Per Event-Type")
plt.ylabel("Total Number of Injuries")
```

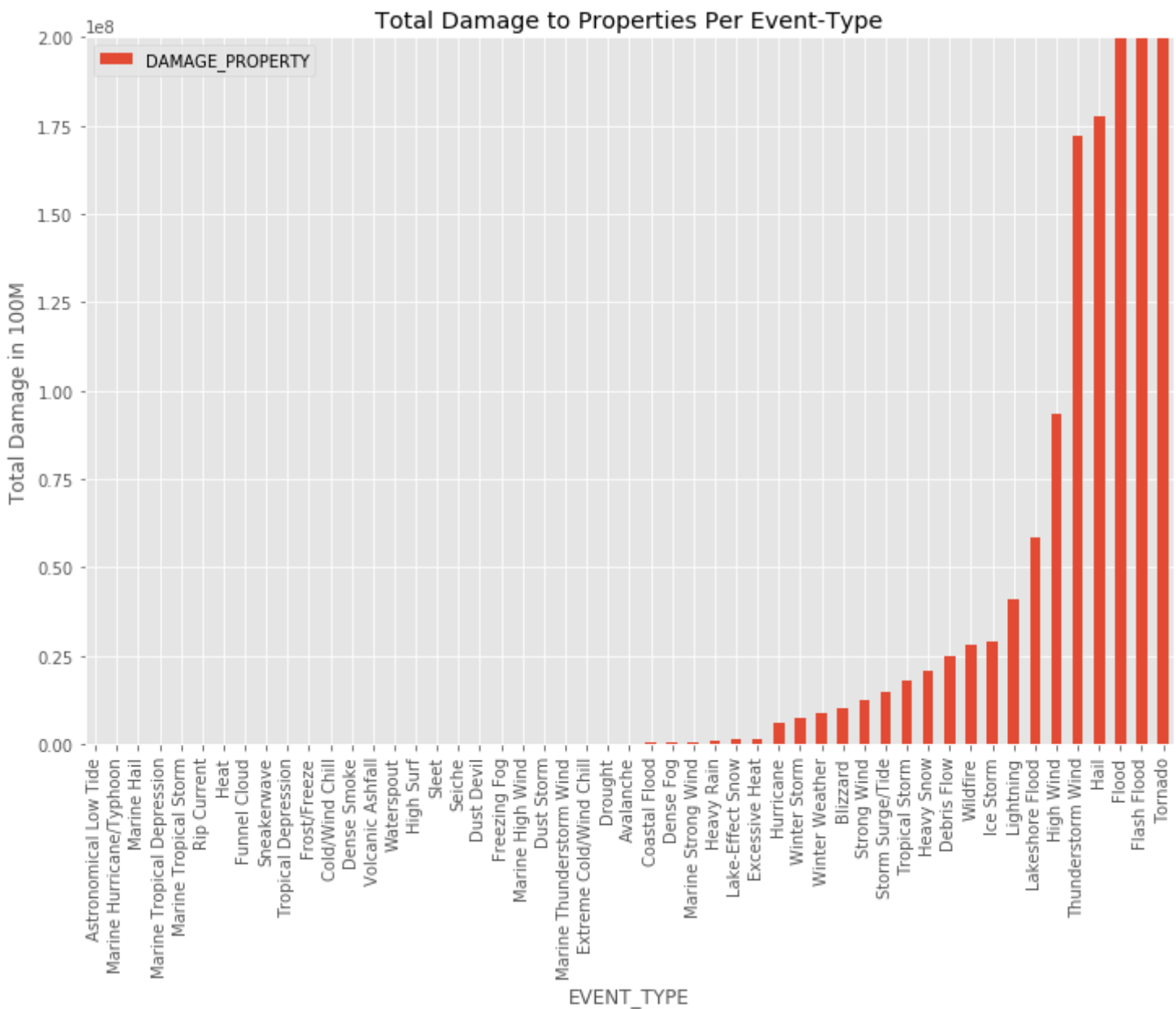
Out[318]: Text(0, 0.5, 'Total Number of Injuries')



In [314]:
```
storm_type_dam_prop = storm.groupby('EVENT_TYPE').agg({'DAMAGE_PROPERTY':'sum'})
```

In [320]:
```
fig_dp = storm_type_dam_prop.sort_values(by='DAMAGE_PROPERTY',ascending=True).plot(kind="bar",figsize=(12,8))
fig_dp.set_ylim(0, 200000000)
plt.title("Total Damage to Properties Per Event-Type")
plt.ylabel("Total Damage in 100M")

#NOTE: Floods, Flash Floods, and Tornados have damage of more than $200M.
```

Out[320]: Text(0, 0.5, 'Total Damage in 100M')



Analysis Methodology:

Our major considerations are some important attributes from the dataset, like the number of deaths or injuries, and locations that were affected by the storms also the total damage caused.

We are interested in finding the relationships, if any, on the above-stated questions and understanding what kind of magnitudes of the storm are influencing how many deaths, injuries in America.

Based on this we are planning to model our data and predict the number of deaths in the future years due to a particular storm of some specific magnitude to take preventive measures in a much better way.

Project Schedule:

Proposal Date: February 28
Meeting with the staff: TBD
Milestone: March 29
Meeting with the staff: TBD
Final Submission: April 19
Project Presentation: April 21

References:

http://www-das.uwyo.edu/~geerts/cwx/notes/chap03/nat_hazard.html (http://www-das.uwyo.edu/~geerts/cwx/notes/chap03/nat_hazard.html)
https://www.depts.ttu.edu/nwi/research/DebrisImpact/Reports/DDS.pdf (https://www.depts.ttu.edu/nwi/research/DebrisImpact/Reports/DDS.pdf)
https://www.nbcnews.com/news/weather/atlantic-hurricane-seasons-2019-2010-graphics-data-n1091986 (https://www.nbcnews.com/news/weather/atlantic-hurricane-seasons-2019-2010-graphics-data-n1091986)
https://www.c2es.org/content/hurricanes-and-climate-change/ (https://www.c2es.org/content/hurricanes-and-climate-change/)