In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import os
import seaborn as sns
data = pd.read_csv('parking_duration_of_parking_event_vs_street_ID.csv')
print("Data Loaded")
```

```
Data Loaded
```

In [2]:

```
# Which streets have the most parking violations?
```

In [3]:

```
grouped_data = data.groupby("Street Name").mean()
sorted_data = grouped_data.sort_values(by=["In Violation?"], ascending=False)
street_violations = sorted_data['In Violation?']
print("Streets with highest average \'In Violations?")
print(sorted_data['In Violation?'])
```

```
Streets with highest average 'In Violations?
Street Name
ANDERSON STREET      0.694432
PRINCESS STREET      0.482243
FAWKNER STREET       0.355798
UNION STREET         0.353401
ANTHONY STREET       0.348657
                       ...
SWANSTON STREET      0.060473
WELLINGTON PARADE    0.054233
ERROL STREET         0.053391
DUDLEY STREET        0.042318
CITY ROAD            0.039162
Name: In Violation?, Length: 75, dtype: float64
```

In [4]:

```
# data['Arrival Time'] = data['Arrival Time'].astype('datetime64')
data['Arrival Time'] = pd.to_datetime(data['Arrival Time'], dayfirst=True)
data['year'] = pd.DatetimeIndex(data['Arrival Time']).year
data['month'] = pd.DatetimeIndex(data['Arrival Time']).month
grouped_data_by_date = data.groupby(["Area Name", "year", "month"]).agg({'In Violation?': np.mean}).reset_index()
print("Group data to calculate the average violations by month in each year")
```

```
Group data to calculate the average violations by month in each year
```

In [5]:

```
# Convert Arrival Time to datetime format
data['Arrival Time'] = pd.to_datetime(data['Arrival Time'], dayfirst=True)

# Check for errors
pd.to_datetime(data['Arrival Time'], errors='coerce').isnull().value_counts()

# Print datatype
print("Converted 'Arrival Time' to datetime format:")
print(data['Arrival Time'].dtypes)
```

```
Converted 'Arrival Time' to datetime format:
datetime64[ns]
```

In [6]:

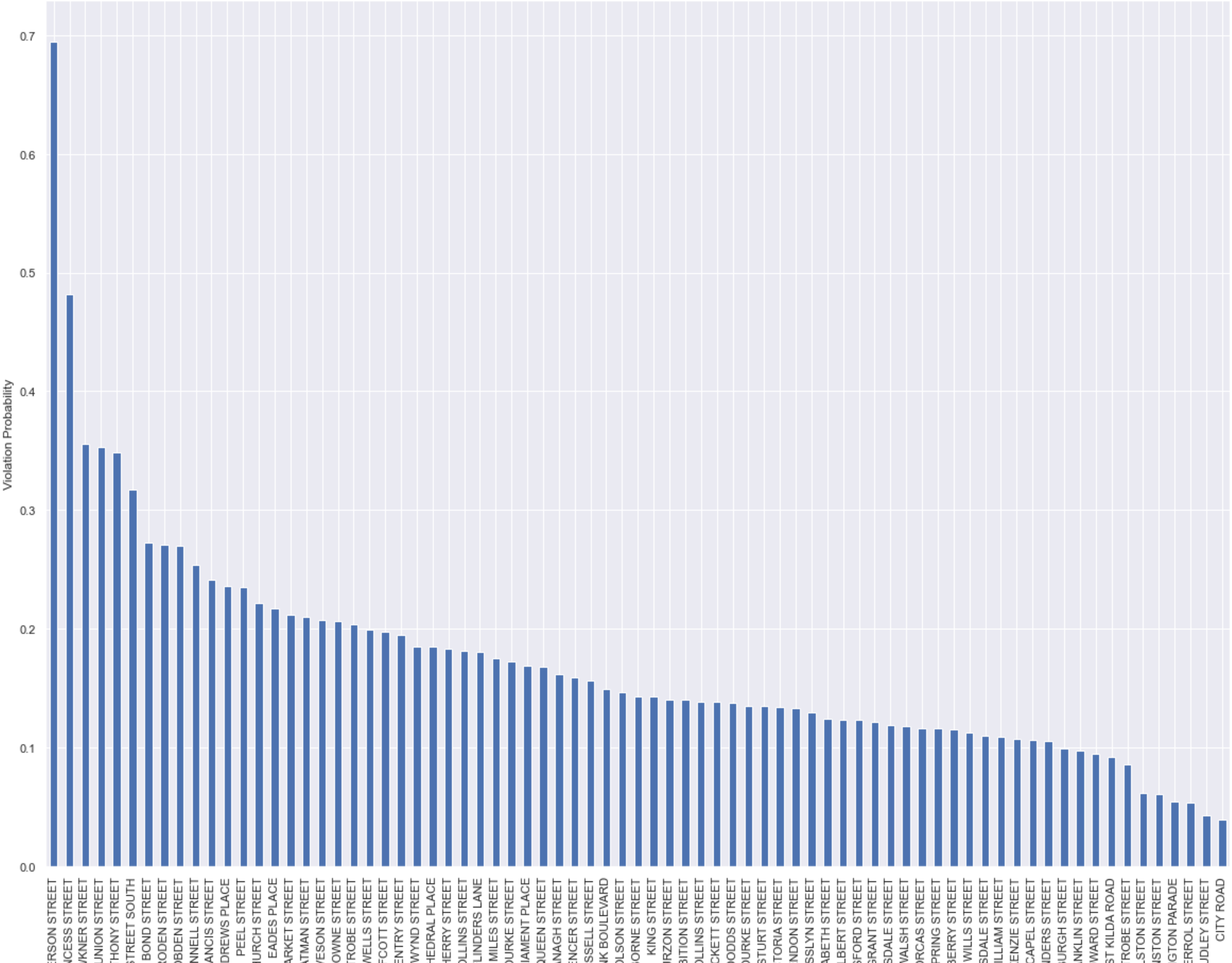
```
#Can we predict the probability that parking in a certain street will result in a parking violation?
```

In [7]:

```
#Set Plot Dimensions
sns.set(rc={'figure.figsize':(20,15)})

#Display Plot
my_plot = sorted_data['In Violation?'].plot(kind='bar')
my_plot.set_xlabel("Street Name")
my_plot.set_ylabel("Violation Probability")


plt.show()
```


Violation Probability
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
ERSON STREET
NCESS STREET
WKNER STREET
UNION STREET
THONY STREET
STREET SOUTH
BOND STREET
RODEN STREET
OBDEN STREET
NNELL STREET
ANCIS STREET
DREWS PLACE
PEEL STREET
HURCH STREET
EADES PLACE
ARKET STREET
ATMAN STREET
VESON STREET
OWNE STREET
TROBE STREET
WELLS STREET
FCOTT STREET
ENTRY STREET
WYND STREET
HEDRAL PLACE
HERRY STREET
OLLINS STREET
LINDERS LANE
MILES STREET
OURKE STREET
IAMENT PLACE
QUEEN STREET
ANAGH STREET
ENCER STREET
SSELL STREET
NK BOULEVARD
OLSON STREET
BORNE STREET
KING STREET
URZON STREET
BITION STREET
OLLINS STREET
CKETT STREET
ODDS STREET
OURKE STREET
STURT STREET
TORIA STREET
NDON STREET
SSLYN STREET
ABETH STREET
LBERT STREET
SFORD STREET
GRANT STREET
SDALE STREET
WALSH STREET
ORCAS STREET
PRING STREET
BERRY STREET
WILLS STREET
SDALE STREET
ILLIAM STREET
ENZIE STREET
CAPEL STREET
NDERS STREET
BURGH STREET
ANKLIN STREET
WARD STREET
ST KILDA ROAD
TROBE STREET
LSTON STREET
NSTON STREET
GTON PARADE
ERROL STREET
UDLEY STREET
CITY ROAD

In [ ]: