In [1]:
```
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, min, max, isnan, when, desc, udf,
import matplotlib.pyplot as plt
from pyspark.sql.types import StringType
import re

spark = SparkSession.builder.master("local").appName("demo1").getOrCreate

import warnings
warnings.filterwarnings('ignore')
```

```
23/05/20 00:30:23 WARN SparkSession: Using an existing Spark session; o
nly runtime SQL configurations will take effect.
```

In [2]:
```
#import data
df = spark.read.load('hdfs://orion11:11001/Project3/Part2/NYPD_Complaint_
                     , format='csv', sep=','
                     , inferSchema='true'
                     , header='true')
```

In [3]:
```
df.take(2)
```

```
23/05/19 18:34:58 WARN package: Truncated the string representation of
a plan since it was too large. This behavior can be adjusted by setting
'spark.sql.debug.maxToStringFields'.
```

Out[3]:
```
[Row(CMPLNT_NUM='469069650', CMPLNT_FR_DT='09/04/2019', CMPLNT_FR_TM=da
tetime.datetime(2023, 5, 19, 9, 0), CMPLNT_TO_DT='09/04/2019', CMPLNT_T
O_TM='10:00:00', ADDR_PCT_CD=9, RPT_DT='09/04/2019', KY_CD=341, OFNS_DE
SC='PETIT LARCENY', PD_CD=321, PD_DESC='LARCENY,PETIT FROM AUTO', CRM_A
TPT_CPTD_CD='COMPLETED', LAW_CAT_CD='MISDEMEANOR', BORO_NM='MANHATTAN',
LOC_OF_OCCUR_DESC='REAR OF', PREM_TYP_DESC=None, JURIS_DESC='N.Y. POLIC
E DEPT', JURISDICTION_CODE=0, PARKS_NM=None, HADEVELOPT=None, HOUSING_P
SA=None, X_COORD_CD=989667, Y_COORD_CD=203049, SUSP_AGE_GROUP=None, SUS
P_RACE=None, SUSP_SEX=None, TRANSIT_DISTRICT=None, Latitude=40.72400587
0000035, Longitude=-73.98045825599996, Lat_Lon='(40.724005870000035, -7
3.98045825599996)', PATROL_BORO='PATROL BORO MAN SOUTH', STATION_NAME=N
one, VIC_AGE_GROUP='25-44', VIC_RACE='BLACK', VIC_SEX='F'),
 Row(CMPLNT_NUM='629841380', CMPLNT_FR_DT='08/31/2019', CMPLNT_FR_TM=da
tetime.datetime(2023, 5, 19, 18, 58), CMPLNT_TO_DT='08/31/2019', CMPLNT
_TO_TM='19:03:00', ADDR_PCT_CD=50, RPT_DT='08/31/2019', KY_CD=341, OFNS
_DESC='PETIT LARCENY', PD_CD=343, PD_DESC='LARCENY,PETIT OF BICYCLE', C
RM_ATPT_CPTD_CD='COMPLETED', LAW_CAT_CD='MISDEMEANOR', BORO_NM='BRONX',
LOC_OF_OCCUR_DESC='FRONT OF', PREM_TYP_DESC='STREET', JURIS_DESC='N.Y.
POLICE DEPT', JURISDICTION_CODE=0, PARKS_NM=None, HADEVELOPT=None, HOUS
ING_PSA=None, X_COORD_CD=1007013, Y_COORD_CD=260060, SUSP_AGE_GROUP=Non
e, SUSP_RACE=None, SUSP_SEX=None, TRANSIT_DISTRICT=None, Latitude=40.88
045772900006, Longitude=-73.91768494199994, Lat_Lon='(40.8804577290000
6, -73.91768494199994)', PATROL_BORO='PATROL BORO BRONX', STATION_NAME=
None, VIC_AGE_GROUP='45-64', VIC_RACE='UNKNOWN', VIC_SEX='M')]
```

In [4]:
```
df.dtypes
```

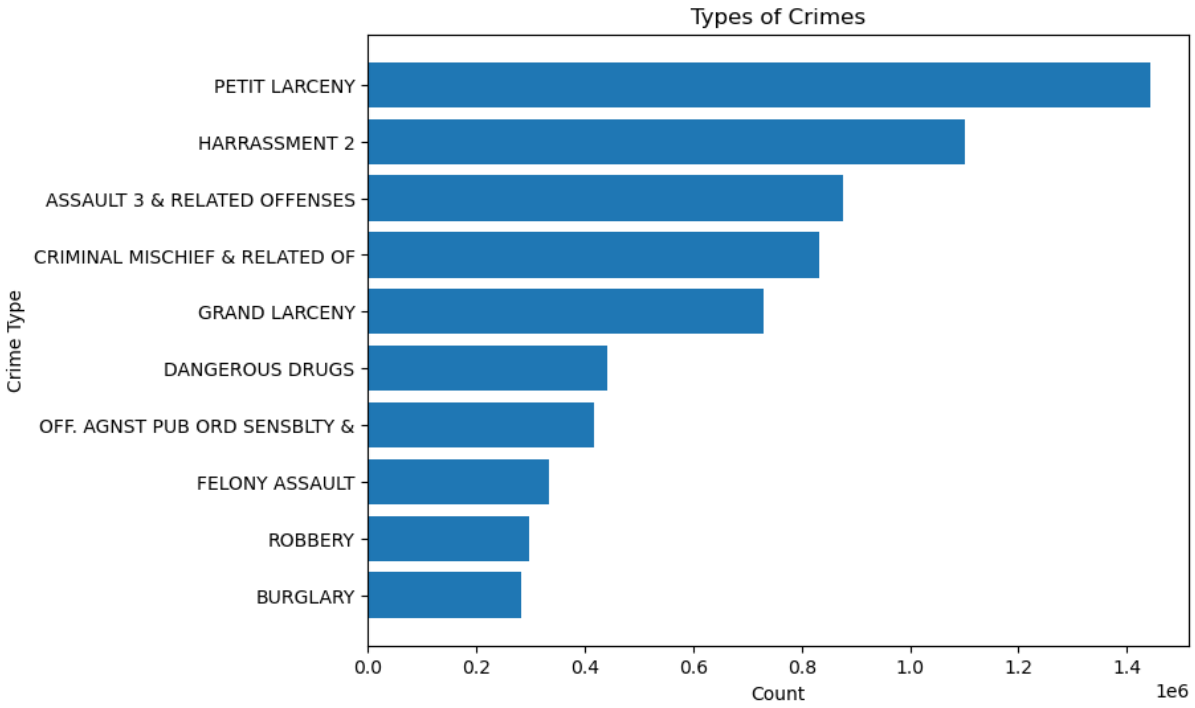
Out[4]:
```
[('CMPLNT_NUM', 'string'),
 ('CMPLNT_FR_DT', 'string'),
 ('CMPLNT_FR_TM', 'timestamp'),
 ('CMPLNT_TO_DT', 'string'),
 ('CMPLNT_TO_TM', 'string'),
 ('ADDR_PCT_CD', 'int'),
 ('RPT_DT', 'string'),
 ('KY_CD', 'int'),
 ('OFNS_DESC', 'string'),
 ('PD_CD', 'int'),
 ('PD_DESC', 'string'),
 ('CRM_ATPT_CPTD_CD', 'string'),
 ('LAW_CAT_CD', 'string'),
 ('BORO_NM', 'string'),
 ('LOC_OF_OCCUR_DESC', 'string'),
 ('PREM_TYP_DESC', 'string'),
 ('JURIS_DESC', 'string'),
 ('JURISDICTION_CODE', 'int'),
 ('PARKS_NM', 'string'),
 ('HADEVELOPT', 'string'),
 ('HOUSING_PSA', 'string'),
 ('X_COORD_CD', 'int'),
 ('Y_COORD_CD', 'int'),
 ('SUSP_AGE_GROUP', 'string'),
 ('SUSP_RACE', 'string'),
 ('SUSP_SEX', 'string'),
 ('TRANSIT_DISTRICT', 'int'),
 ('Latitude', 'double'),
 ('Longitude', 'double'),
 ('Lat_Lon', 'string'),
 ('PATROL_BORO', 'string'),
 ('STATION_NAME', 'string'),
 ('VIC_AGE_GROUP', 'string'),
 ('VIC_RACE', 'string'),
 ('VIC_SEX', 'string')]
```

In [5]:
```
#####Clean the data
# Dropping rows with NaN values in specific columns
df = df.dropna(subset=['Y_COORD_CD','X_COORD_CD','Latitude','Longitude','

# Dropping columns that are not significant for future data exploration
df = df.drop(*['PARKS_NM','STATION_NAME','TRANSIT_DISTRICT','HADEVELOPT',

# Replacing NaN values in 'LOC_OF_OCCUR_DESC', 'VIC_RACE', 'VIC_AGE_GROUP
df = df.fillna('UNKNOWN', subset=['LOC_OF_OCCUR_DESC','VIC_RACE','VIC_AGE

# Print the shape of the cleaned dataset
print('Clean dataset:')
print('Observations:', df.count())
print('Variables:', len(df.columns))

# Examine the changes
df.show(5)
```

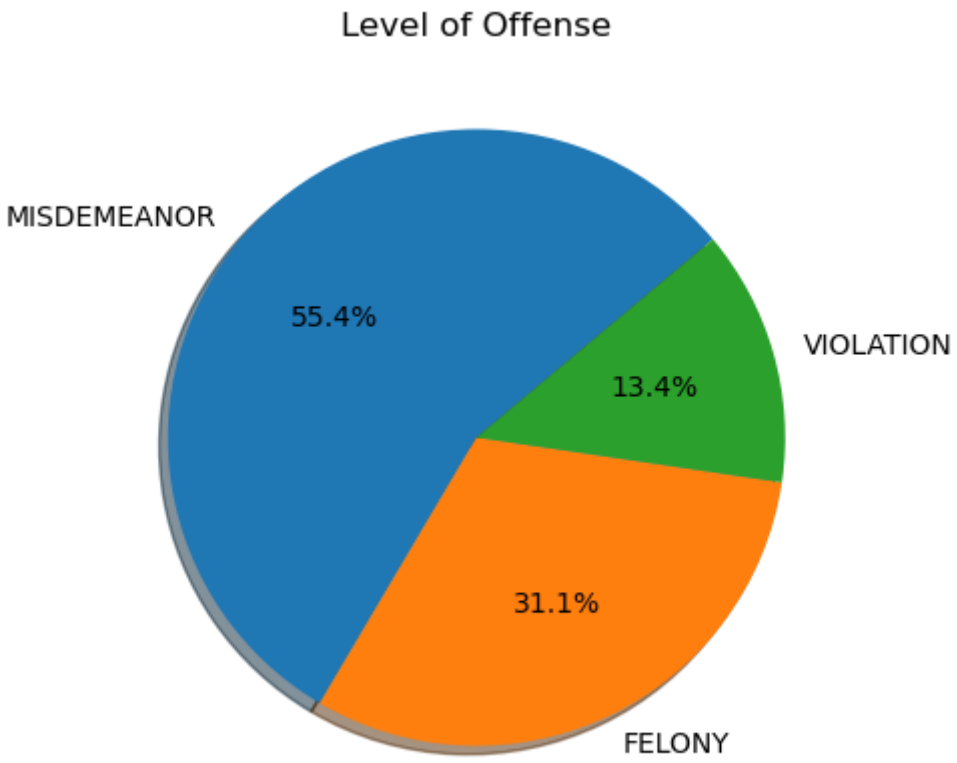
```
Clean dataset:
```

```
Observations: 8308403
Variables: 19
+----------+-----------+-------------------+----------+-----+---------
----+---------------+-----------+---------+-----------------+---------
-------+----------+----------+-----------------+-----------------+---
-----------------+-------------+--------+-------+
|CMPLNT_NUM|CMPLNT_FR_DT|       CMPLNT_FR_TM|    RPT_DT|KY_CD|    OFNS_
DESC|CRM_ATPT_CPTD_CD| LAW_CAT_CD|  BORO_NM|LOC_OF_OCCUR_DESC|      JUR
IS_DESC|X_COORD_CD|Y_COORD_CD|          Latitude|         Longitude|
Lat_Lon|VIC_AGE_GROUP|VIC_RACE|VIC_SEX|
+----------+-----------+-------------------+----------+-----+---------
----+---------------+-----------+---------+-----------------+---------
-------+----------+----------+-----------------+-----------------+---
-----------------+-------------+--------+-------+
| 469069650|  09/04/2019|2023-05-19 09:00:00|09/04/2019|  341|PETIT LAR
CENY|       COMPLETED|MISDEMEANOR|MANHATTAN|          REAR OF|N.Y. POLI
CE DEPT|    989667|    203049|40.724005870000035|-73.98045825599996|(4
0.7240058700000...|        25-44|   BLACK|      F|
| 629841380|  08/31/2019|2023-05-19 18:58:00|08/31/2019|  341|PETIT LAR
CENY|       COMPLETED|MISDEMEANOR|    BRONX|         FRONT OF|N.Y. POLI
CE DEPT|   1007013|    260060| 40.88045772900006|-73.91768494199994|(4
0.8804577290000...|        45-64| UNKNOWN|      M|
| 918597562|  08/31/2007|2023-05-19 17:00:00|09/04/2007|  107|     BURG
LARY|       COMPLETED|     FELONY|MANHATTAN|           INSIDE|N.Y. POLI
CE DEPT|   1001734|    247180|      40.845118059|     -73.936808674|(4
0.845118059, -7...|      UNKNOWN| UNKNOWN|      D|
| 224389328|  09/07/2019|2023-05-19 22:00:00|09/07/2019|  341|PETIT LAR
CENY|       COMPLETED|MISDEMEANOR|MANHATTAN|           INSIDE|N.Y. POLI
CE DEPT|    994570|    217188| 40.76280953700007|-73.96274775799998|(4
0.7628095370000...|      UNKNOWN| UNKNOWN|      D|
| 303540290|  02/05/2015|2023-05-19 13:55:00|02/05/2015|  107|     BURG
LARY|       COMPLETED|     FELONY| BROOKLYN|      OPPOSITE OF|N.Y. POLI
CE DEPT|   1001063|    186966|       40.67984749|     -73.939384473|(4
0.67984749, -73...|      UNKNOWN| UNKNOWN|      D|
+----------+-----------+-------------------+----------+-----+---------
----+---------------+-----------+---------+-----------------+---------
-------+----------+----------+-----------------+-----------------+---
-----------------+-------------+--------+-------+
only showing top 5 rows
```

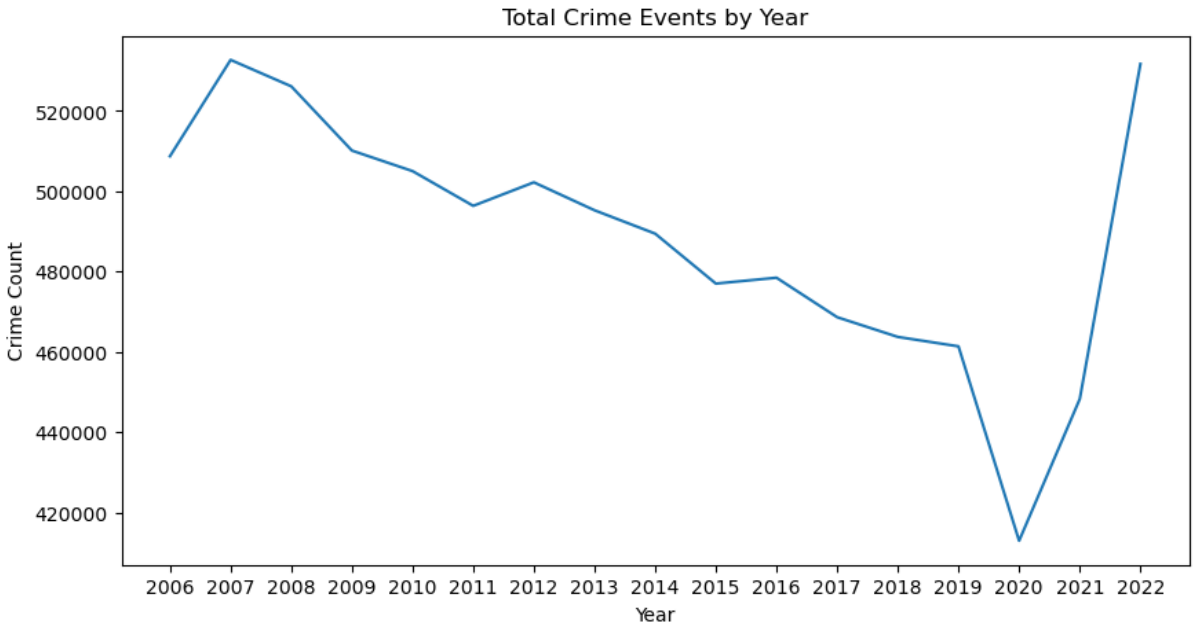
In [6]:
```
top_crimes = df.groupBy("OFNS_DESC").count().orderBy(desc("count")).limit

# Convert the DataFrame to Pandas for plotting
top_crimes_list = top_crimes.collect()

# Extract the crime types and counts into separate lists
crime_types = [row.OFNS_DESC for row in top_crimes_list]
crime_counts = [row["count"] for row in top_crimes_list]

# Plot the bar chart using PySpark
plt.figure(figsize=(8, 6))
plt.barh(crime_types, crime_counts)
plt.gca().invert_yaxis()
plt.title("Types of Crimes")
plt.xlabel("Count")
plt.ylabel("Crime Type")
plt.show()
```



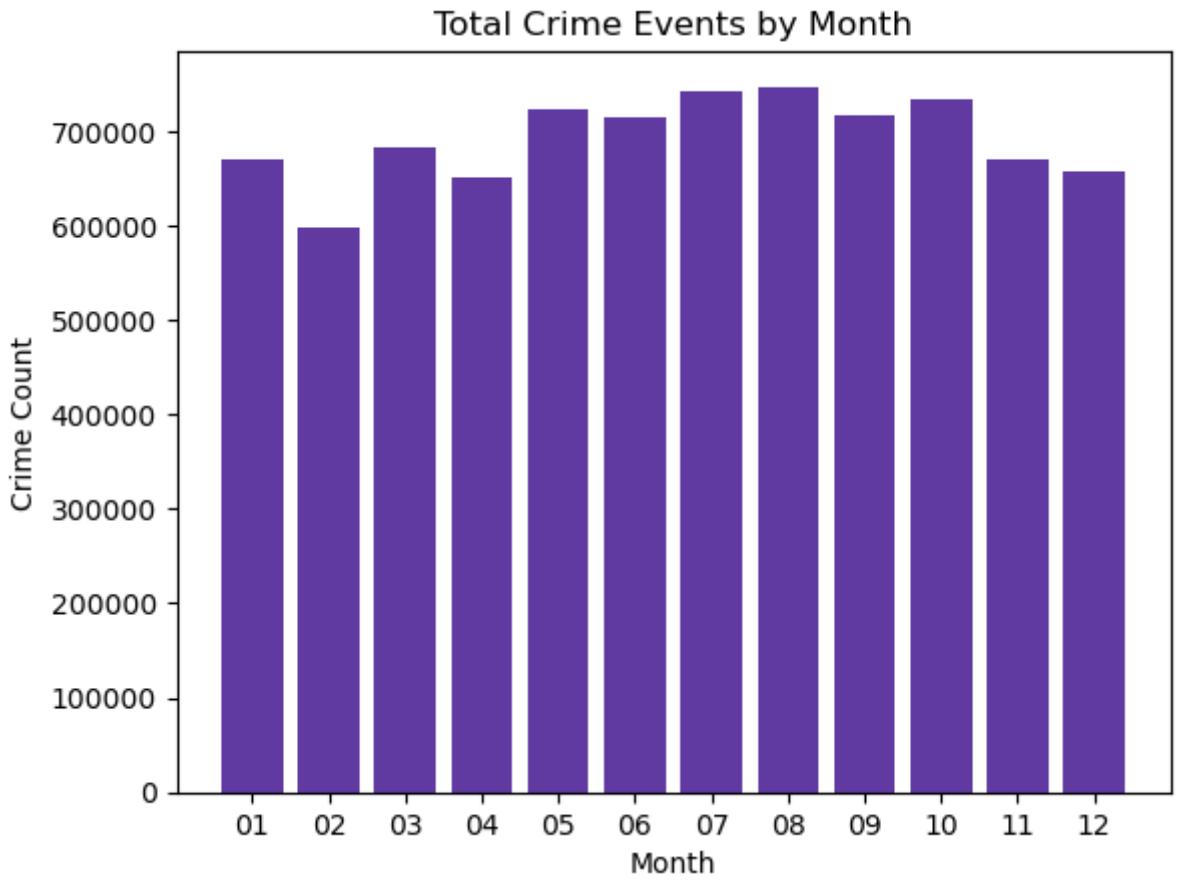
In [7]:
```
# Perform value counts
offense_counts = df.groupBy("LAW_CAT_CD").count().orderBy(desc("count"))

# Convert the DataFrame to Pandas for plotting
offense_counts_list = offense_counts.collect()

# Extract the offense levels and counts into separate lists
offense_levels = [row.LAW_CAT_CD for row in offense_counts_list]
offense_counts = [row["count"] for row in offense_counts_list]

# Plot the pie chart using PySpark
plt.figure(figsize=(10, 5))
plt.pie(offense_counts, labels=offense_levels, autopct='%1.1f%%', startan
plt.title("Level of Offense")
plt.show()
```



In [8]:
```
# Extract the year using regular expression and create a new column 'year
df = df.withColumn("year", regexp_extract("RPT_DT", r"\d{4}", 0))

# Perform value counts
year_counts = df.groupBy("year").count().orderBy("year")

# Convert the DataFrame to Pandas for plotting
year_counts_list = year_counts.collect()

# Extract the years and counts into separate lists
years = [row.year for row in year_counts_list]
crime_counts = [row["count"] for row in year_counts_list]

# Plot the line chart using PySpark
plt.figure(figsize=(10, 5))
plt.plot(years, crime_counts)
plt.title("Total Crime Events by Year")
plt.xlabel("Year")
plt.ylabel("Crime Count")
plt.show()
```



In [17]:
```
# Extract the month using regular expression and create a new column 'mon
df = df.withColumn("month", regexp_extract("RPT_DT", r"(\d{2})", 1))

# Group by month and get the count
month_counts = df.groupBy("month").count().orderBy("month")

# Collect the result into a list of Row objects
counts_list = month_counts.collect()

# Extract the month and count values from the list
months = [row['month'] for row in counts_list]
counts = [row['count'] for row in counts_list]

# Plot the bar chart using Matplotlib
plt.bar(months, counts, color="#603AA1")
plt.title("Total Crime Events by Month")
plt.xlabel("Month")
plt.ylabel("Crime Count")
plt.show()
```



In [19]:
```
# Extract the hour from the 'CMPLNT_FR_TM' column
df = df.withColumn("hour", hour("CMPLNT_FR_TM"))

# Group by hour and get the count
hour_counts = df.groupBy("hour").count().orderBy("hour")

# Convert the hour values to strings
hour_counts = hour_counts.withColumn("hour", hour_counts["hour"].cast("st

# Collect the result into a list of Row objects
counts_list = hour_counts.collect()

# Extract the hour and count values from the list
hours = [row['hour'] for row in counts_list]
counts = [row['count'] for row in counts_list]

# Plot the bar chart using Matplotlib
plt.bar(hours, counts, color="#298540")
plt.title("Total Crime Events by Hour")
plt.xlabel("Hour")
plt.ylabel("Crime Count")
plt.show()
```

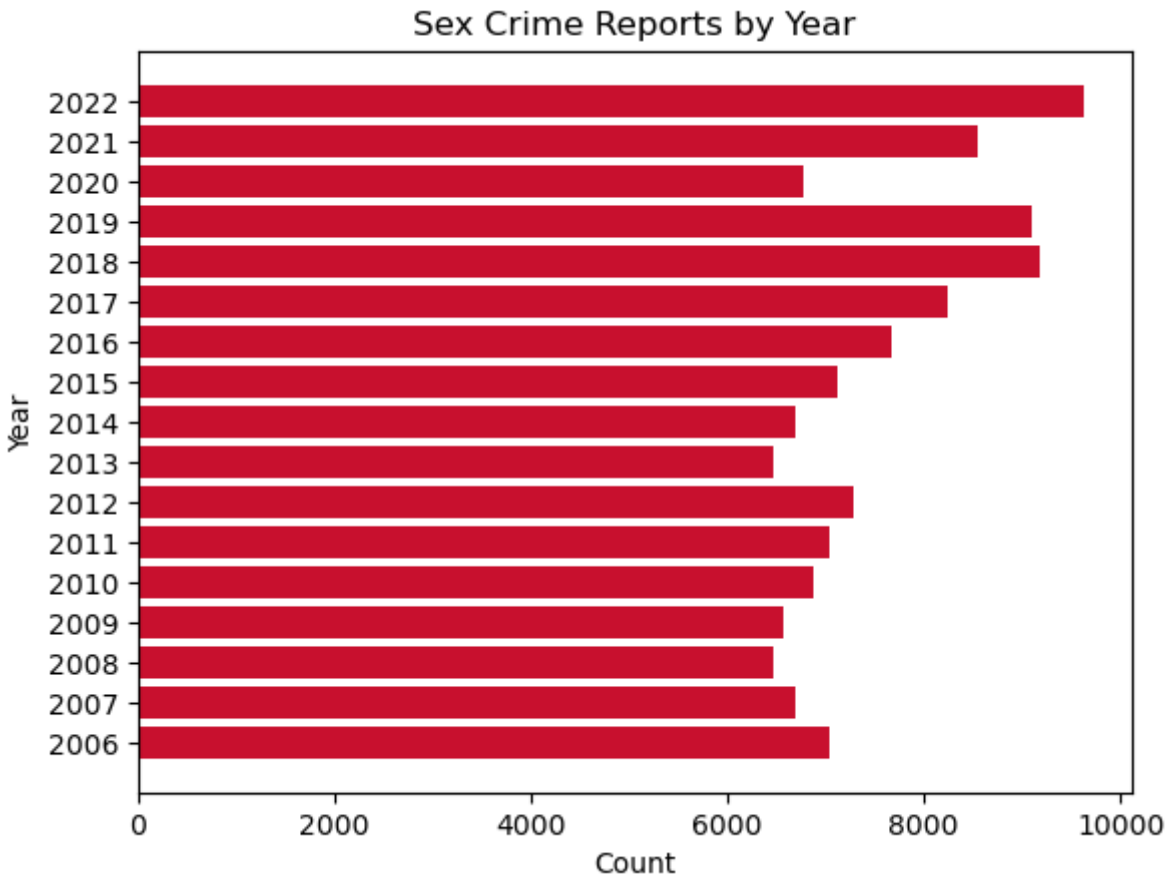
In [11]:
```
# Filter the DataFrame to include rows where the 'OFNS_DESC' column conta
sex_crimes = df.filter(col("OFNS_DESC").rlike("SEX CRIMES|RAPE"))

# Show the first few rows of the filtered DataFrame
sex_crimes.show()
```

```
+----------+------------+-------------------+----------+-----+---------
-+---------------+-----------+---------+-----------------+------------
----+----------+----------+-----------------+-----------------+------
-------------+------------+-------------------+-------+----+-----+--
--+
|CMPLNT_NUM|CMPLNT_FR_DT|       CMPLNT_FR_TM|    RPT_DT|KY_CD| OFNS_DES
C|CRM_ATPT_CPTD_CD| LAW_CAT_CD|  BORO_NM|LOC_OF_OCCUR_DESC|      JURIS_
DESC|X_COORD_CD|Y_COORD_CD|          Latitude|         Longitude|
Lat_Lon|VIC_AGE_GROUP|            VIC_RACE|VIC_SEX|year|month|hour|
+----------+------------+-------------------+----------+-----+---------
-+---------------+-----------+---------+-----------------+------------
----+----------+----------+-----------------+-----------------+------
-------------+------------+-------------------+-------+----+-----+--
--+
| 343363487|  10/28/2012|2023-05-19 05:00:00|06/27/2017|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|    BRONX|          UNKNOWN|N.Y. POLICE
DEPT|   1026387|    262634|      40.887451313|     -73.847607787|(40.88
7451313, -7...|        25-44|              WHITE|      F|2017|   06|
5|
| 668119706|  03/12/2010|2023-05-19 16:30:00|03/12/2010|  104|      RAP
E|       COMPLETED|     FELONY|    BRONX|           INSIDE|N.Y. POLICE
DEPT|   1027295|    251236|      40.856162957|     -73.844397124|(40.85
6162957, -7...|          <18|              BLACK|      F|2010|   03|
16|
| 352203686|  10/24/2018|2023-05-19 04:00:00|09/13/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|MANHATTAN|           INSIDE|N.Y. POLICE
DEPT|    984623|    209859| 40.74269929900004|-73.99865537999993|(40.74
26992990000...|      UNKNOWN|            UNKNOWN|      F|2019|   09|
4|
| 108984487|  09/08/2019|2023-05-19 03:00:00|09/11/2019|  104|      RAP
E|       COMPLETED|     FELONY|   QUEENS|           INSIDE|N.Y. POLICE
DEPT|   1018699|    215043| 40.75686097700003|-73.87565666399996|(40.75
68609770000...|        25-44|      WHITE HISPANIC|      F|2019|   09|
3|
| 706732006|  09/09/2019|2023-05-19 11:04:00|09/09/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|    BRONX|         FRONT OF|N.Y. POLICE
DEPT|   1006434|    244344| 40.83732351100008|-73.91983075699994|(40.83
73235110000...|          <18|              BLACK|      F|2019|   09|
11|
| 157890980|  07/12/2019|2023-05-19 09:20:00|07/12/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|    BRONX|          UNKNOWN|N.Y. POLICE
DEPT|   1006434|    244344| 40.83732351100008|-73.91983075699994|(40.83
73235110000...|        18-24|      WHITE HISPANIC|      F|2019|   07|
9|
| 534846644|  07/18/2019|2023-05-19 03:15:00|07/18/2019|  116|SEX CRIME
S|       COMPLETED|     FELONY|   QUEENS|           INSIDE|N.Y. POLICE
DEPT|   1041749|    196938| 40.70704747500002|-73.79261190399995|(40.70
70474750000...|        25-44|      WHITE HISPANIC|      F|2019|   07|
3|
| 195904001|  07/21/2019|2023-05-19 04:30:00|07/22/2019|  116|SEX CRIME
S|       COMPLETED|     FELONY|    BRONX|           INSIDE|N.Y. POLICE
DEPT|   1011779|    246746| 40.84390125500005|     -73.900504632|(40.84
39012550000...|        25-44|      WHITE HISPANIC|      F|2019|   07|
4|
| 189492814|  07/06/2019|2023-05-19 02:00:00|07/06/2019|  116|SEX CRIME
S|       COMPLETED|     FELONY|    BRONX|          UNKNOWN|N.Y. POLICE
DEPT|   1004926|    234532| 40.81039601900005|-73.92531074499993|(40.81
```

```
03960190000...|        25-44|               WHITE|      F|2019|   07|
2|
| 178042438|  10/27/2014|2023-05-19 06:44:00|10/27/2014|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|MANHATTAN|          UNKNOWN|N.Y. POLICE
DEPT|    992411|    215025|      40.756874911|     -73.970544057|(40.75
6874911, -7...|        25-44|               WHITE|      F|2014|   10|
6|
| 967247975|  06/03/2019|2023-05-19 09:00:00|06/26/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|   QUEENS|           INSIDE|N.Y. POLICE
DEPT|   1020255|    210816| 40.74525274100006|-73.87006286999997|(40.74
52527410000...|          <18|      WHITE HISPANIC|      F|2019|   06|
9|
| 398809294|  06/19/2019|2023-05-19 14:00:00|06/20/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|    BRONX|           INSIDE|N.Y. POLICE
DEPT|   1006434|    244344| 40.83732351100008|-73.91983075699994|(40.83
73235110000...|          <18|      BLACK HISPANIC|      M|2019|   06|
14|
| 713337798|  05/06/2012|2023-05-19 03:50:00|05/06/2012|  104|      RAP
E|       COMPLETED|     FELONY|    BRONX|          REAR OF|N.Y. POLICE
DEPT|   1020316|    239179|      40.823101299|     -73.869690461|(40.82
3101299, -7...|        18-24|      WHITE HISPANIC|      F|2012|   05|
3|
| 760747009|  01/07/2015|2023-05-19 05:00:00|01/07/2015|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR| BROOKLYN|           INSIDE|N.Y. POLICE
DEPT|   1003606|    185050|      40.674583308|     -73.930221541|(40.67
4583308, -7...|        45-64|               BLACK|      M|2015|   01|
5|
| 311323786|  07/31/2012|2023-05-19 05:00:00|07/31/2012|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|   QUEENS|         FRONT OF|N.Y. POLICE
DEPT|   1020255|    210816|      40.745252741|      -73.87006287|(40.74
5252741, -7...|      UNKNOWN|             UNKNOWN|      F|2012|   07|
5|
| 182795315|  06/18/2019|2023-05-19 16:20:00|06/18/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|   QUEENS|           INSIDE|N.Y. POLICE
DEPT|   1032198|    217060| 40.76233421800004|-73.82691730799998|(40.76
23342180000...|          <18|ASIAN / PACIFIC I...|      F|2019|   06|
16|
| 534042624|  06/20/2019|2023-05-19 22:05:00|06/21/2019|  116|SEX CRIME
S|       COMPLETED|     FELONY|    BRONX|           INSIDE|N.Y. POLICE
DEPT|   1006434|    244344| 40.83732351100008|-73.91983075699994|(40.83
73235110000...|        25-44|               BLACK|      F|2019|   06|
22|
| 118056951|  05/09/2019|2023-05-19 20:30:00|05/12/2019|  104|      RAP
E|       COMPLETED|     FELONY|MANHATTAN|           INSIDE|N.Y. POLICE
DEPT|   1001936|    245282| 40.83990820100007|-73.93608358699998|(40.83
99082010000...|        25-44|               WHITE|      F|2019|   05|
20|
| 578590261|  05/15/2019|2023-05-19 06:00:00|05/15/2019|  116|SEX CRIME
S|       COMPLETED|     FELONY|MANHATTAN|           INSIDE|N.Y. POLICE
DEPT|    988353|    217918|40.764818269000045|-73.98518977299993|(40.76
48182690000...|        18-24|               WHITE|      F|2019|   05|
6|
| 757861293|  07/28/2017|2023-05-19 00:01:00|04/30/2019|  233|SEX CRIME
S|       COMPLETED|MISDEMEANOR|   QUEENS|           INSIDE|N.Y. POLICE
DEPT|   1007654|    219564| 40.76930608700008|-73.91550817999997|(40.76
93060870000...|        18-24|             UNKNOWN|      F|2019|   04|
0|
```

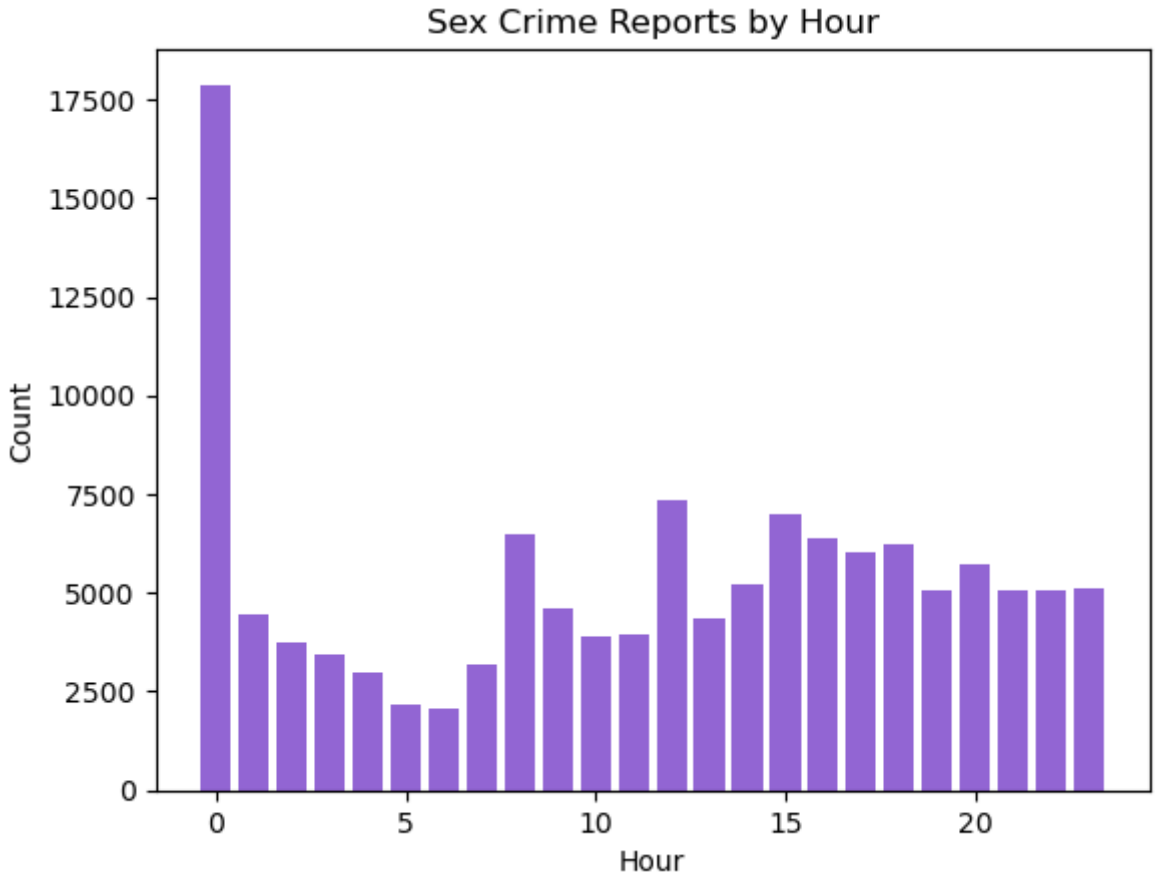
```
+----------+------------+--------------------+----------+-----+---------
-+----------------+-----------+---------+-----------------+------------
----+----------+----------+-------------------+------------------+------
--------------+-------------+--------------------+-------+----+-----+--
--+
only showing top 20 rows
```

In [20]:
```
# Group by year and count the number of occurrences
crime_counts = sex_crimes.groupBy("year").agg(count("*").alias("count")).

# Convert the DataFrame to Pandas for plotting
crime_counts_list = crime_counts.collect()

# Extract year and count values from the DataFrame
years = [row["year"] for row in crime_counts_list]
counts = [row["count"] for row in crime_counts_list]

# Plot the bar graph using Matplotlib
plt.barh(years, counts, color='#C8102E')
plt.title("Sex Crime Reports by Year")
plt.xlabel("Count")
plt.ylabel("Year")
plt.show()

# Calculate average sex crimes per year
mean = crime_counts.selectExpr("avg(count)").first()[0]
print(round(mean, 2))
```

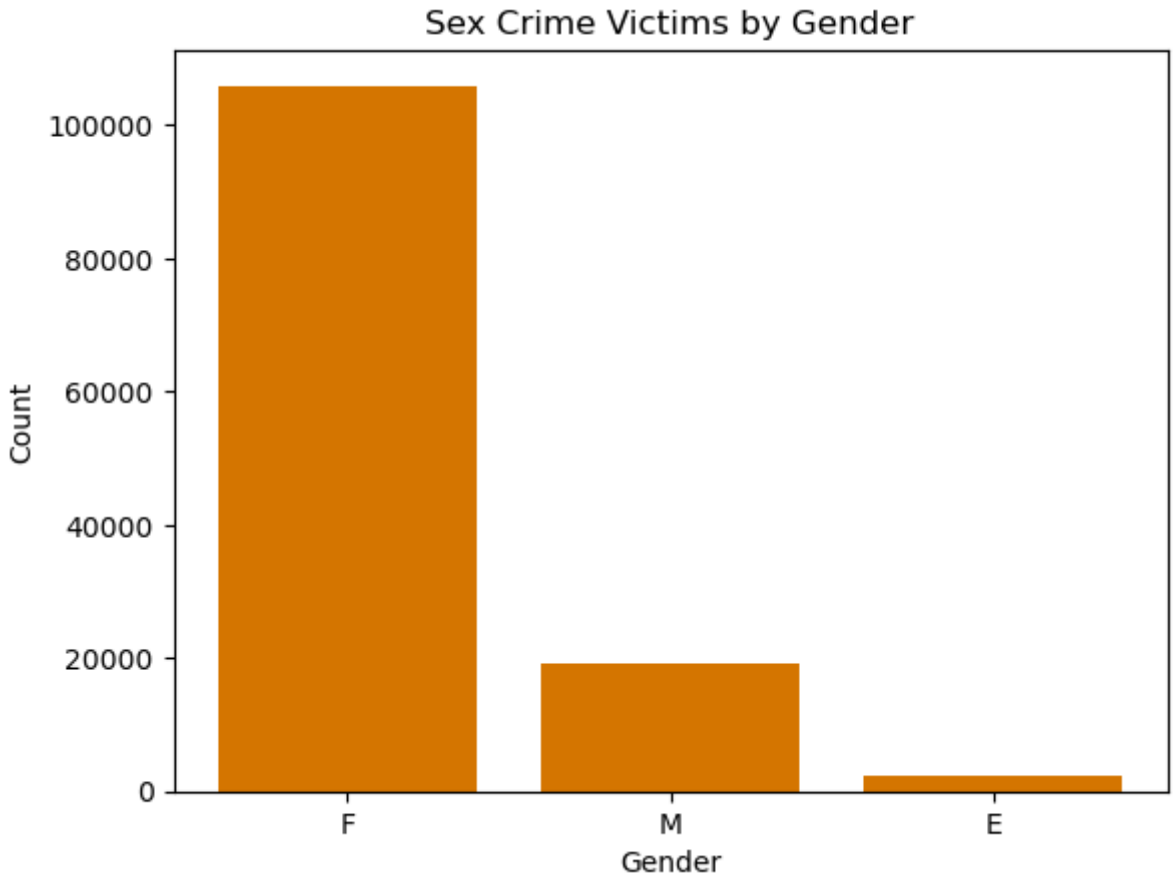



```
[Stage 129:================================================>     (19 +
2) / 21]

7494.18
```

In [21]:
```
# Extract the hour from the 'CMPLNT_FR_TM' column
sex_crimes = sex_crimes.withColumn("hour", hour("CMPLNT_FR_TM"))

# Group by hour and count the number of occurrences
crime_counts = sex_crimes.groupBy("hour").agg(count("*").alias("count")).

# Convert the DataFrame to Pandas for plotting
crime_counts_list = crime_counts.collect()

# Extract hour and count values from the DataFrame
hours = [row["hour"] for row in crime_counts_list]
counts = [row["count"] for row in crime_counts_list]

# Plot the bar graph using Matplotlib
plt.bar(hours, counts, color='#9265D3')
plt.title("Sex Crime Reports by Hour")
plt.xlabel("Hour")
plt.ylabel("Count")
plt.show()
```

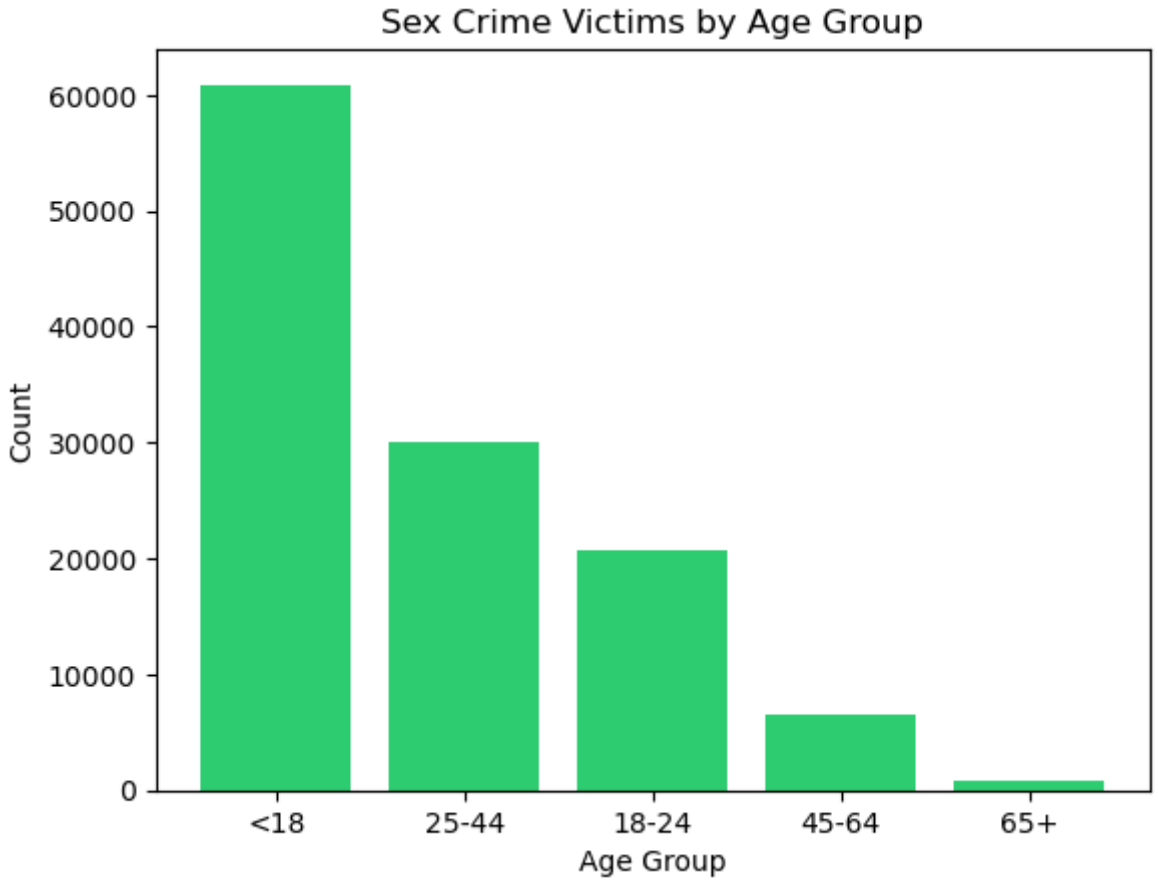


In [22]:
```
# Group by 'VIC_SEX' and count the number of occurrences
crime_counts = sex_crimes.groupBy("VIC_SEX").agg(count("*").alias("count"

# Sort the counts in descending order and limit to the top 3
top_victims = crime_counts.orderBy(col("count").desc()).limit(3)

# Convert the DataFrame to Pandas for plotting
top_victims_list = top_victims.collect()

# Extract VIC_SEX and count values from the DataFrame
vic_sex = [row["VIC_SEX"] for row in top_victims_list]
counts = [row["count"] for row in top_victims_list]

# Plot the bar graph using Matplotlib
plt.bar(vic_sex, counts, color='#D47500')
plt.title("Sex Crime Victims by Gender")
plt.xlabel("Gender")
plt.ylabel("Count")
plt.show()

# Calculate the percentage of victims by gender
total_count = sex_crimes.count()
vic_sex_per = crime_counts.withColumn("percentage", (col("count") / lit(t
vic_sex_per.show()
```



```
[Stage 149:=====================================================>  (20 + 
1) / 21]

+-------+--------------------+
|VIC_SEX|          percentage|
+-------+--------------------+
|      F|   83.13749499611463|
|      E|  1.7072079497021215|
|      M|   15.04383796045557|
|      D| 0.10988924733714807|
|      L|7.849231952653434E-4|
|UNKNOWN|7.849231952653434E-4|
+-------+--------------------+
```

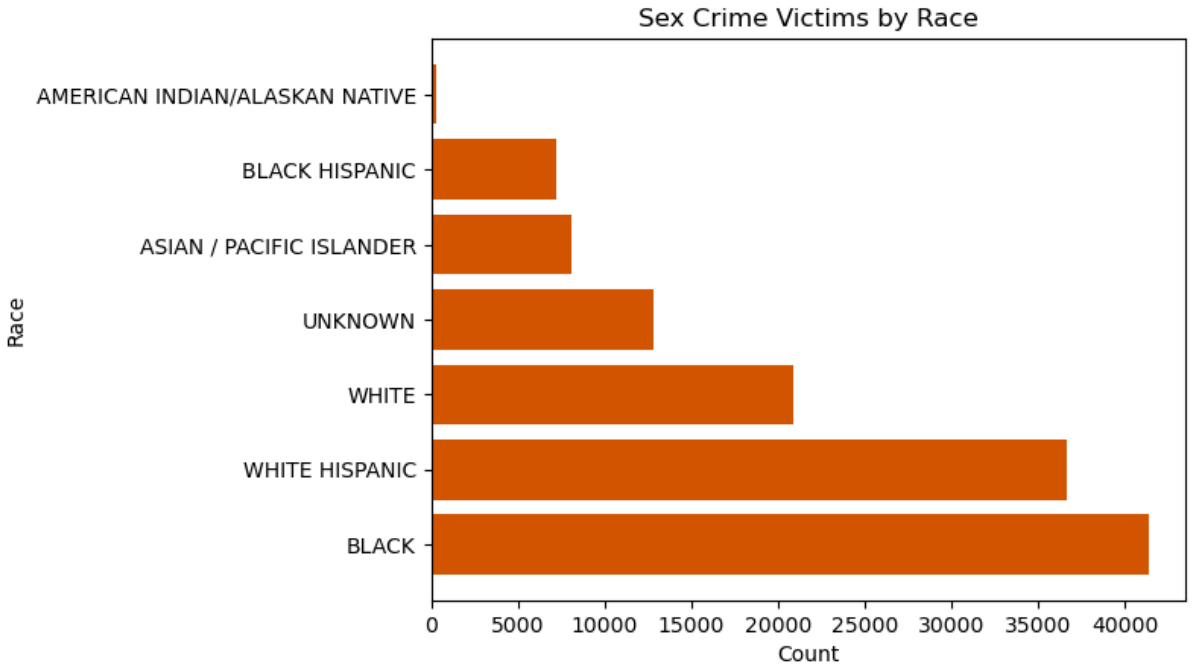
In [15]:
```
# Group by 'VIC_AGE_GROUP' and count the number of occurrences
crime_counts = sex_crimes.groupBy("VIC_AGE_GROUP").agg(count("*").alias("

# Select the desired age groups and sort the counts in descending order
selected_age_groups = ["<18", "18-24", "25-44", "45-64", "65+"]

filtered_counts = crime_counts.filter(col("VIC_AGE_GROUP").isin(selected_
filtered_counts = filtered_counts.orderBy(col("count").desc())

# Convert the DataFrame to Pandas for plotting
filtered_counts_list = filtered_counts.collect()

# Extract VIC_AGE_GROUP and count values from the DataFrame
vic_age_group = [row["VIC_AGE_GROUP"] for row in filtered_counts_list]
counts = [row["count"] for row in filtered_counts_list]

# Plot the bar graph using Matplotlib
plt.bar(vic_age_group, counts, color='#2ECC71')
plt.title("Sex Crime Victims by Age Group")
plt.xlabel("Age Group")
plt.ylabel("Count")
plt.show()

# Calculate the percentage of victims by age group
total_count = sex_crimes.count()
vic_age_per = filtered_counts.withColumn("percentage", (col("count") / li
vic_age_per.show()
```




```
[Stage 85:=====================================================>  (20 +
1) / 21]
```

```
+-------------+------------------+
|VIC_AGE_GROUP|        percentage|
+-------------+------------------+
|          <18| 47.79632812929255|
|        25-44| 23.53906170281238|
|        18-24|16.290295994536937|
|        45-64| 5.093366614076813|
|          65+|0.6200893242596212|
+-------------+------------------+
```

In [16]:
```
# Group by 'VIC_RACE' and count the number of occurrences
crime_counts = sex_crimes.groupBy("VIC_RACE").agg(count("*").alias("count

# Sort the counts in descending order and limit to the top 7
top_victims = crime_counts.orderBy(col("count").desc()).limit(7)

# Convert the DataFrame to Pandas for plotting
top_victims_list = top_victims.collect()

# Extract VIC_RACE and count values from the DataFrame
vic_race = [row["VIC_RACE"] for row in top_victims_list]
counts = [row["count"] for row in top_victims_list]

# Plot the bar graph using Matplotlib
plt.barh(vic_race, counts, color='#D35400')
plt.title("Sex Crime Victims by Race")
plt.xlabel("Count")
plt.ylabel("Race")
plt.show()

# Calculate the percentage of victims by race
total_count = sex_crimes.count()
vic_race_per = top_victims.withColumn("percentage", (col("count") / lit(t
vic_race_per.show()
```




```
[Stage 94:=================================================>      (19 +
2) / 21]
```

```
+--------------------+------------------+
|            VIC_RACE|        percentage|
+--------------------+------------------+
|               BLACK|32.487186128837294|
|      WHITE HISPANIC|28.745457257007402|
|               WHITE|16.397830472288284|
|             UNKNOWN| 10.09175752152652|
|ASIAN / PACIFIC I...| 6.330405569814994|
|      BLACK HISPANIC| 5.672639932182636|
|AMERICAN INDIAN/A...|0.2723683487570741|
+--------------------+------------------+
```