In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

In [3]:
```
the_file = pd.read_csv("metro.csv")
the_file.describe()
```

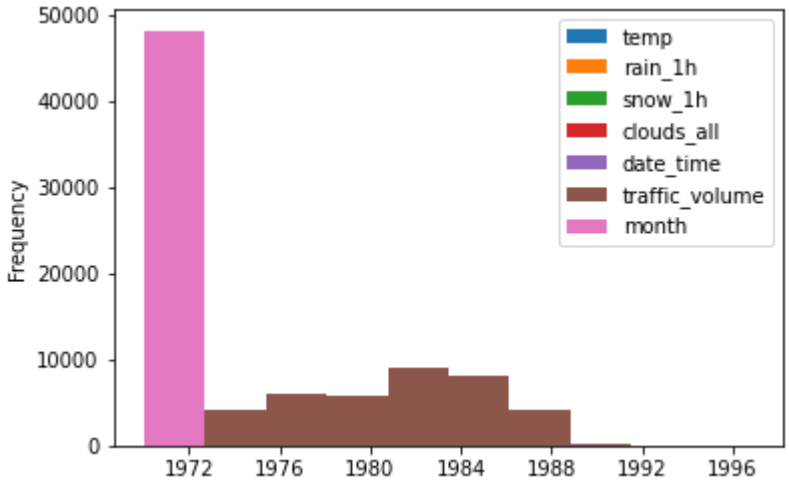
Out[3]:

	temp	rain_1h	snow_1h	clouds_all	traffic_volume
count	48204.000000	48204.000000	48204.000000	48204.000000	48204.000000
mean	281.205870	0.334264	0.000222	49.362231	3259.818355
std	13.338232	44.789133	0.008168	39.015750	1986.860670
min	0.000000	0.000000	0.000000	0.000000	0.000000
25%	272.160000	0.000000	0.000000	1.000000	1193.000000
50%	282.450000	0.000000	0.000000	64.000000	3380.000000
75%	291.806000	0.000000	0.000000	90.000000	4933.000000
max	310.070000	9831.300000	0.510000	100.000000	7280.000000

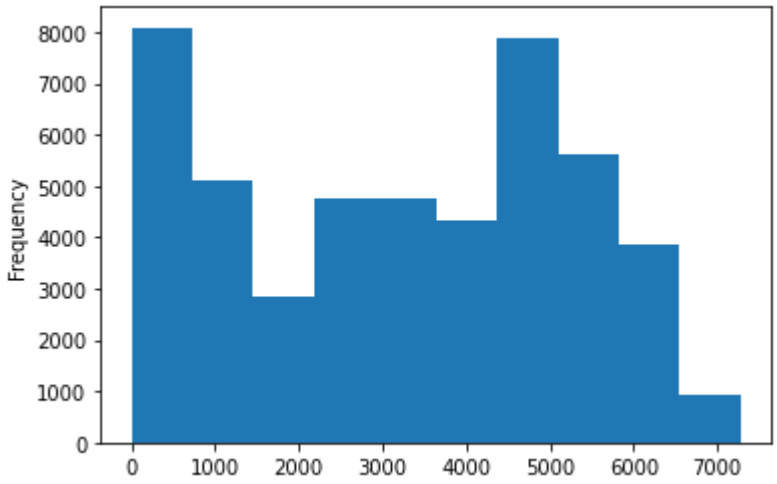
In [6]:
```
the_file.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 48204 entries, 0 to 48203
Data columns (total 10 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   holiday              48204 non-null  object
 1   temp                 48204 non-null  float64
 2   rain_1h              48204 non-null  float64
 3   snow_1h              48204 non-null  float64
 4   clouds_all           48204 non-null  int64
 5   weather_main         48204 non-null  object
 6   weather_description  48204 non-null  object
 7   date_time            48204 non-null  datetime64[ns]
 8   traffic_volume       48204 non-null  int64
 9   month                48204 non-null  int64
dtypes: datetime64[ns](1), float64(3), int64(3), object(3)
memory usage: 3.7+ MB
```

In [7]:
```
the_file.plot.hist();
```



In [8]:
```
the_file["traffic_volume"].plot.hist();
```



In [9]:
```
the_file["date_time"] = pd.to_datetime(the_file["date_time"])
```

In [10]:
```
the_file
```

Out[10]:

	holiday	temp	rain_1h	snow_1h	clouds_all	weather_main	weather_description	date_tim
0	None	288.28	0.0	0.0	40	Clouds	scattered clouds	2012-02 1(09:00:0(
1	None	289.36	0.0	0.0	75	Clouds	broken clouds	2012-02 1(10:00:0(
2	None	289.58	0.0	0.0	90	Clouds	overcast clouds	2012-02 1(11:00:0(
3	None	290.13	0.0	0.0	90	Clouds	overcast clouds	2012-02 1(12:00:0(
4	None	291.14	0.0	0.0	75	Clouds	broken clouds	2012-02 1(13:00:0(
...	...	...	...	...	...	...	...	..
48199	None	283.45	0.0	0.0	75	Clouds	broken clouds	2018-09 3(19:00:0(
48200	None	282.76	0.0	0.0	90	Clouds	overcast clouds	2018-09 3(20:00:0(
48201	None	282.73	0.0	0.0	90	Thunderstorm	proximity thunderstorm	2018-09 3(21:00:0(
48202	None	282.09	0.0	0.0	90	Clouds	overcast clouds	2018-09 3(22:00:0(
48203	None	282.12	0.0	0.0	90	Clouds	overcast clouds	2018-09 3(23:00:0(

48204 rows × 10 columns

In [11]:
```
day_night = the_file["date_time"]
day_bool = (day_night.dt.hour >= 7) & (day_night.dt.hour < 19)
night_bool = (day_night.dt.hour >= 19) | (day_night.dt.hour < 7)
day = the_file[day_bool]
night = the_file[night_bool]
```

In [12]:
```
day
```

Out[12]:

	holiday	temp	rain_1h	snow_1h	clouds_all	weather_main	weather_description	date_tim
0	None	288.28	0.00	0.0	40	Clouds	scattered clouds	2012-02 1 09:00:0
1	None	289.36	0.00	0.0	75	Clouds	broken clouds	2012-02 1 10:00:0
2	None	289.58	0.00	0.0	90	Clouds	overcast clouds	2012-02 1 11:00:0
3	None	290.13	0.00	0.0	90	Clouds	overcast clouds	2012-02 1 12:00:0
4	None	291.14	0.00	0.0	75	Clouds	broken clouds	2012-02 1 13:00:0
...	...	...	...	...	...	...	...	..
48194	None	283.84	0.00	0.0	75	Rain	proximity shower rain	2018-09 3 15:00:0
48195	None	283.84	0.00	0.0	75	Drizzle	light intensity drizzle	2018-09 3 15:00:0
48196	None	284.38	0.00	0.0	75	Rain	light rain	2018-09 3 16:00:0
48197	None	284.79	0.00	0.0	75	Clouds	broken clouds	2018-09 3 17:00:0
48198	None	284.20	0.25	0.0	75	Rain	light rain	2018-09 3 18:00:0

23877 rows × 10 columns

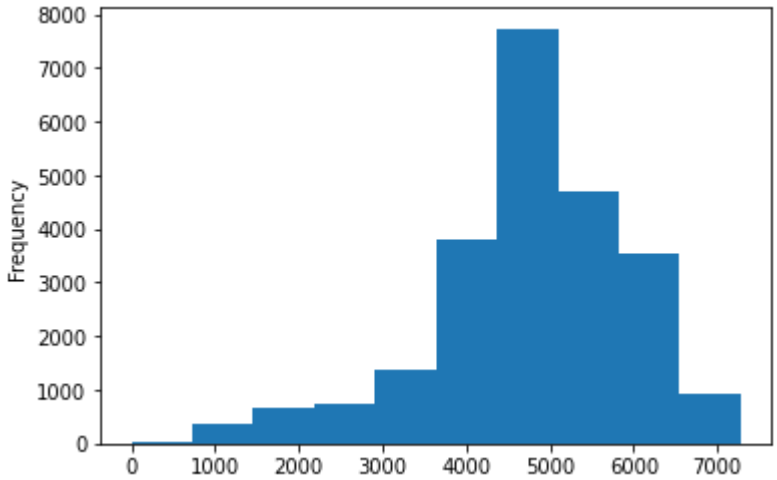
In [13]:
```
the_file
```

Out[13]:

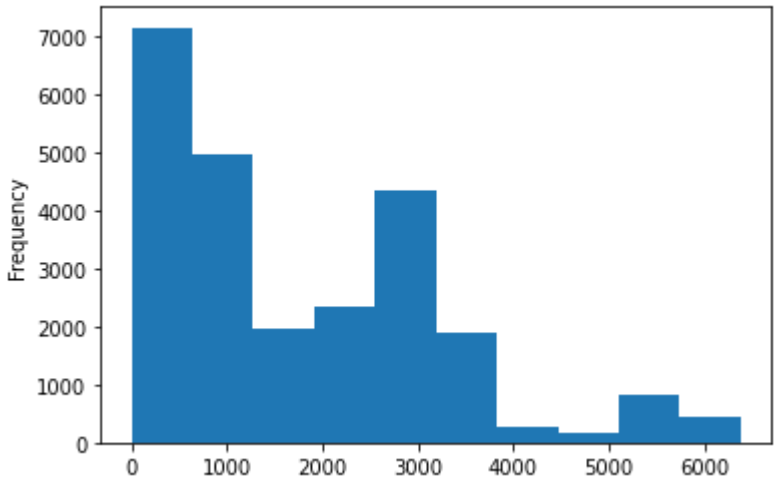
	holiday	temp	rain_1h	snow_1h	clouds_all	weather_main	weather_description	date_tim
0	None	288.28	0.0	0.0	40	Clouds	scattered clouds	2012-02 1 09:00:0
1	None	289.36	0.0	0.0	75	Clouds	broken clouds	2012-02 1 10:00:0
2	None	289.58	0.0	0.0	90	Clouds	overcast clouds	2012-02 1 11:00:0
3	None	290.13	0.0	0.0	90	Clouds	overcast clouds	2012-02 1 12:00:0
4	None	291.14	0.0	0.0	75	Clouds	broken clouds	2012-02 1 13:00:0
...	...	...	...	...	...	...	...	..
48199	None	283.45	0.0	0.0	75	Clouds	broken clouds	2018-09 3 19:00:0
48200	None	282.76	0.0	0.0	90	Clouds	overcast clouds	2018-09 3 20:00:0
48201	None	282.73	0.0	0.0	90	Thunderstorm	proximity thunderstorm	2018-09 3 21:00:0
48202	None	282.09	0.0	0.0	90	Clouds	overcast clouds	2018-09 3 22:00:0
48203	None	282.12	0.0	0.0	90	Clouds	overcast clouds	2018-09 3 23:00:0

48204 rows × 10 columns

In [14]:
```
day["traffic_volume"].plot.hist();
```



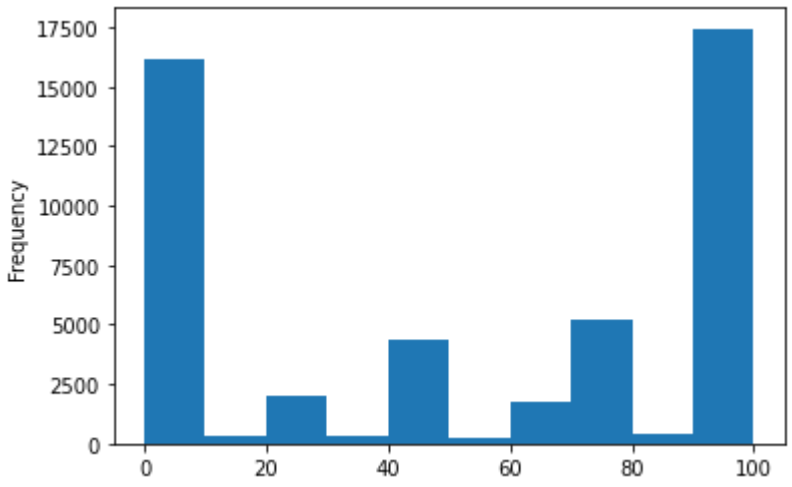
In [15]:
```
night["traffic_volume"].plot.hist();
```



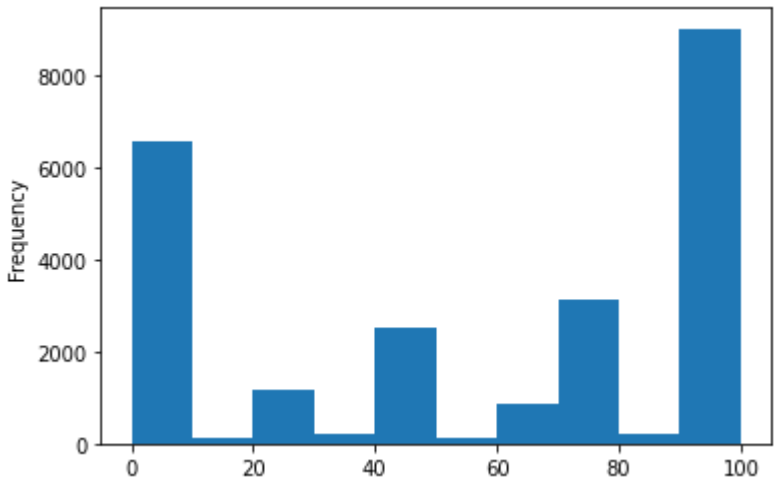
In [16]:
```
the_file["traffic_volume"].describe()
```

Out[16]:
```
count    48204.000000
mean      3259.818355
std       1986.860670
min          0.000000
25%       1193.000000
50%       3380.000000
75%       4933.000000
max       7280.000000
Name: traffic_volume, dtype: float64
```

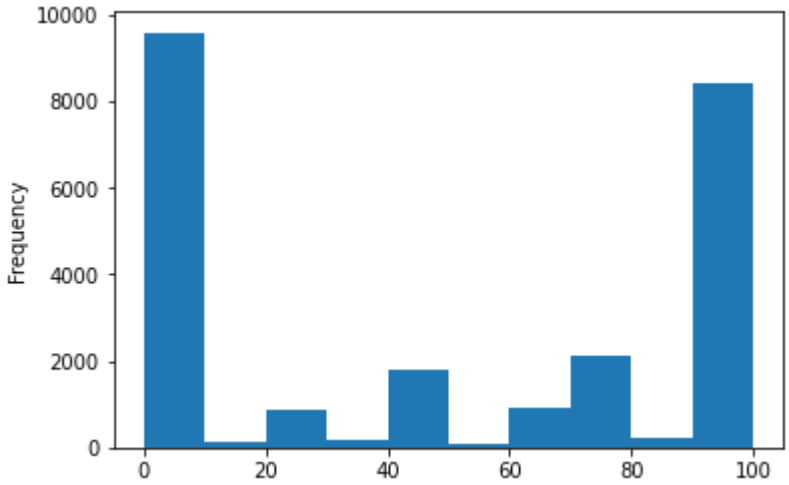
In [26]:
```
the_file["clouds_all"].plot.hist();
```



In [27]:
```
day["clouds_all"].plot.hist();
```



In [19]:
```
night["clouds_all"].plot.hist();
```



In [5]:
```
the_file["month"] = the_file["date_time"].dt.month
```

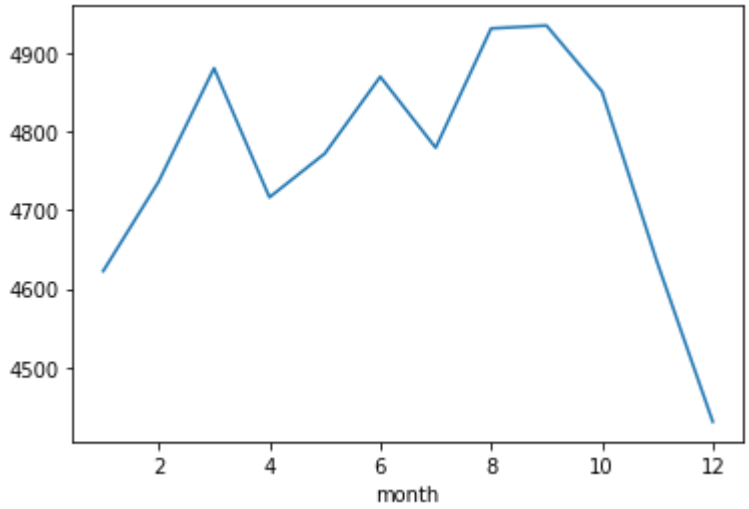
In [20]:
```
day
```

Out[20]:

	holiday	temp	rain_1h	snow_1h	clouds_all	weather_main	weather_description	date_tim
0	None	288.28	0.00	0.0	40	Clouds	scattered clouds	2012-02 1 09:00:0
1	None	289.36	0.00	0.0	75	Clouds	broken clouds	2012-02 1 10:00:0
2	None	289.58	0.00	0.0	90	Clouds	overcast clouds	2012-02 1 11:00:0
3	None	290.13	0.00	0.0	90	Clouds	overcast clouds	2012-02 1 12:00:0
4	None	291.14	0.00	0.0	75	Clouds	broken clouds	2012-02 1 13:00:0
...	...	...	...	...	...	...	...	..
48194	None	283.84	0.00	0.0	75	Rain	proximity shower rain	2018-09 3 15:00:0
48195	None	283.84	0.00	0.0	75	Drizzle	light intensity drizzle	2018-09 3 15:00:0
48196	None	284.38	0.00	0.0	75	Rain	light rain	2018-09 3 16:00:0
48197	None	284.79	0.00	0.0	75	Clouds	broken clouds	2018-09 3 17:00:0
48198	None	284.20	0.25	0.0	75	Rain	light rain	2018-09 3 18:00:0

23877 rows × 10 columns

In [39]:
```
month_day = day.groupby("month").mean()
month_day["traffic_volume"].plot.line()
#plt.xlim(0, 12)
plt.show()
```



In [43]:
```
day['year'] = day.copy()['date_time'].dt.year
```

```
C:\New\envs\snakes\lib\site-packages\ipykernel_launcher.py:1: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/sta
ble/user_guide/indexing.html#returning-a-view-versus-a-copy (https://pandas.pyd
ata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-c
opy)
  """Entry point for launching an IPython kernel.
```

In [ ]:
```
```