jupyter Kovan_Eth Last Checkpoint: 5 minutes ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Code

In [1]:
```
import math
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:
```
result = pd.read_csv("kovanEth.csv")
result.shape
```

Out[2]:
```
(1457, 10)
```

In [3]:
```
kk = result[["block_confirmations","timestamp"]]
kk.head(10)
```

Out[3]:

	block_confirmations	timestamp
0	51 Block Confirmations	5 mins ago (Jun-10-2022 05:40:56 AM +UTC)
1	44 Block Confirmations	4 mins ago (Jun-10-2022 05:43:12 AM +UTC)
2	49 Block Confirmations	5 mins ago (Jun-10-2022 05:40:24 AM +UTC)
3	47 Block Confirmations	4 mins ago (Jun-10-2022 05:42:24 AM +UTC)
4	47 Block Confirmations	4 mins ago (Jun-10-2022 05:39:56 AM +UTC)
5	50 Block Confirmations	5 mins ago (Jun-10-2022 05:40:16 AM +UTC)
6	43 Block Confirmations	4 mins ago (Jun-10-2022 05:43:40 AM +UTC)
7	50 Block Confirmations	5 mins ago (Jun-10-2022 05:42:00 AM +UTC)
8	48 Block Confirmations	5 mins ago (Jun-10-2022 05:40:32 AM +UTC)
9	49 Block Confirmations	4 mins ago (Jun-10-2022 05:42:08 AM +UTC)

In [4]:
```
def solve(k):
    i=0
    r=""
    while k[i]!=" ":
        r+=k[i]
        i+=1
    return int(r)
```

In [5]:
```
arr = []
for i in range(result.shape[0]):
    d = solve(result['block_confirmations'].iloc[i])
    t = solve (result['timestamp'].iloc[i]) * 60
    arr.append( 7 * t / d )
```

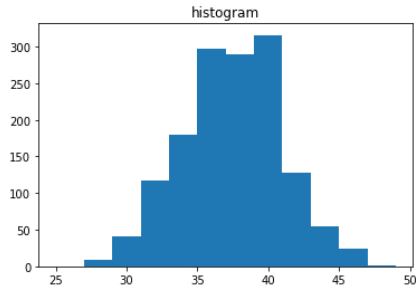
In [11]:
```
df = pd.DataFrame(arr)
df.describe()

# 7 CONFIRMATIONS
```

Out[11]:

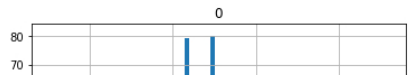
	0
count	1457.000000
mean	37.414094
std	3.531674
min	27.540984
25%	35.000000
50%	37.333333
75%	40.000000
max	48.000000

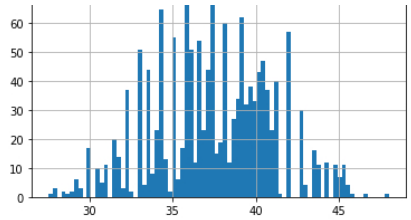
In [12]:
```
plt.hist(arr, bins = [25,27,29,31,33,35,37,39,41,43,45,47,49])
plt.title("histogram")
plt.show()
```



In [13]:
```
df.hist(bins=80)
```

Out[13]:
```
array([[<AxesSubplot:title={'center':'0'}>]], dtype=object)
```





In [16]:
```
ar = []
for i in range(result.shape[0]):
    d = solve(result['block_confirmations'].iloc[i])
    t = solve (result['timestamp'].iloc[i]) * 60
    ar.append( 12 * t / d )
```

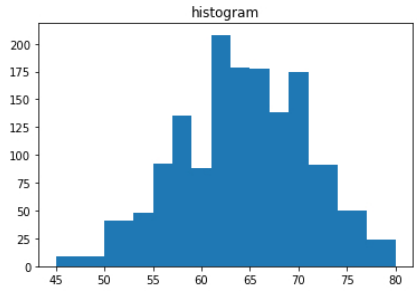
In [17]:
```
df = pd.DataFrame(ar)
df.describe()

# 12 CONFIRMATIONS
```

Out[17]:

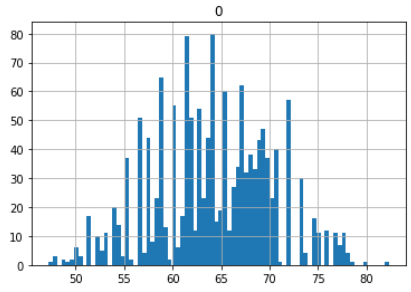
	0
count	1457.000000
mean	64.138447
std	6.054298
min	47.213115
25%	60.000000
50%	64.000000
75%	68.571429
max	82.285714

In [20]:
```
plt.hist(ar, bins = [45,50,53,55,57,59,61,63,65,67,69,71,74,77,80])
plt.title("histogram")
plt.show()
```



In [21]:
```
df.hist(bins=80)
```

Out[21]:
```
array([[<AxesSubplot:title={'center':'0'}>]], dtype=object)
```



In [ ]: