In [1]:

```
import numpy as np
import pandas as pd
# import nltk
```

In [2]:

```
df_sms=pd.read_csv('spam.csv')
df_sms.head()
```

Out[2]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN

In [3]:

```
df_sms=df_sms.drop(["Unnamed: 2", "Unnamed: 3", "Unnamed: 4"] , axis=1)
df_sms=df_sms.rename(columns={"v1":"label","v2":"sms-text"})
```

In [4]:

```
df_sms.head()
```

Out[4]:

	label	sms-text
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

In [5]:

```
print(len(df_sms))
```

```
5572
```

In [6]:

```
df_sms.shape
```

Out[6]:

```
(5572, 2)
```

In [7]:

```
df_sms.tail(5)
```

Out[7]:

	label	sms-text
5567	spam	This is the 2nd time we have tried 2 contact u...
5568	ham	Will �_ b going to esplanade fr home?
5569	ham	Pity, * was in mood for that. So...any other s...
5570	ham	The guy did some bitching but I acted like i'd...
5571	ham	Rofl. Its true to its name

In [8]:

```
df_sms.label.value_counts()
```

Out[8]:

```
ham     4825
spam     747
Name: label, dtype: int64
```

In [9]:

```
df_sms.head()
```

Out[9]:

	label	sms-text
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...

In [10]:

```
df_sms.describe()
```

Out[10]:

	label	sms-text
count	5572	5572
unique	2	5169
top	ham	Sorry, I'll call later
freq	4825	30

In [11]:

```
df_sms['length']=df_sms['sms-text'].apply(len)
```

In [12]:

```
df_sms.head(3)
```

Out[12]:

	label	sms-text	length
0	ham	Go until jurong point, crazy.. Available only ...	111
1	ham	Ok lar... Joking wif u oni...	29
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	155

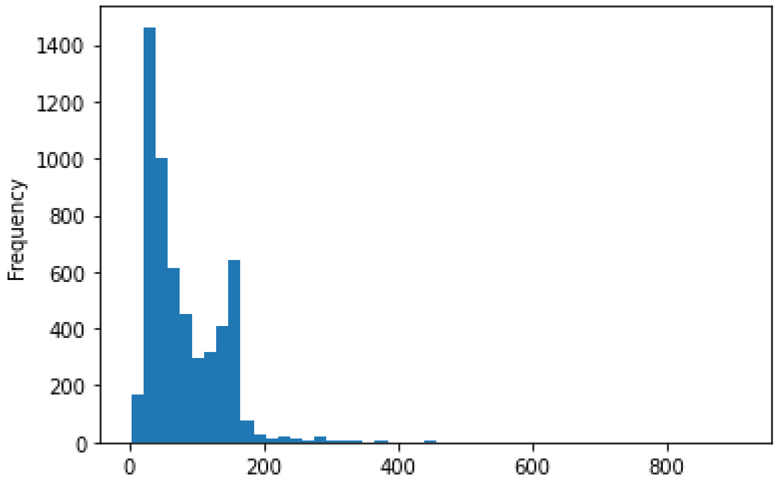
In [13]:

```
import matplotlib.pyplot as plt
import seaborn as sns

%matplotlib inline
df_sms['length'].plot(bins=50,kind='hist')
```

Out[13]:

```
<AxesSubplot:ylabel='Frequency'>
```



In [14]:

```
#Implemenation of Bag of words approach
#step1:Convert all strings to their lower case form
```

In [15]:

```
documents=['Hello,how are you!',
           'Win money ,win from home.',
           'Call me now.'
           'Hello,Call hello you tomorrow? ']
lower_case_documents=[]
lower_case_documents=[d.lower() for d in documents]
print(lower_case_documents)
```

```
['hello,how are you!', 'win money ,win from home.', 'call me now.hello,call
hello you tomorrow? ']
```

In [16]:

```
# Step 2:Removing all punctuations
sans_punctuation_documents=[]
import string
for i in lower_case_documents:
    sans_punctuation_documents.append(i.translate(str.maketrans("","",string.punctuation)))
```

In [17]:

```
sans_punctuation_documents
```

Out[17]:

```
['hellohow are you',
 'win money win from home',
 'call me nowhellocall hello you tomorrow ']
```

In [18]:

```
#Step 3:Tokenization
preprocessed_documents=[[w for w in d.split()] for d in sans_punctuation_documents]
preprocessed_documents
```

Out[18]:

```
[['hellohow', 'are', 'you'],
 ['win', 'money', 'win', 'from', 'home'],
 ['call', 'me', 'nowhellocall', 'hello', 'you', 'tomorrow']]
```

In [19]:

```
#Step 4:Count frequencies
frequency_list=[]
import pprint
from collections import Counter

frequency_list=[Counter (d)for d in preprocessed_documents ]
pprint.pprint(frequency_list)
```

```
[Counter({'hellohow': 1, 'are': 1, 'you': 1}),
 Counter({'win': 2, 'money': 1, 'from': 1, 'home': 1}),
 Counter({'call': 1,
          'me': 1,
          'nowhellocall': 1,
          'hello': 1,
          'you': 1,
          'tomorrow': 1})]
```

In [20]:

```
doc_array=frequency_list
doc_array
```

Out[20]:

```
[Counter({'hellohow': 1, 'are': 1, 'you': 1}),
 Counter({'win': 2, 'money': 1, 'from': 1, 'home': 1}),
 Counter({'call': 1,
          'me': 1,
          'nowhellocall': 1,
          'hello': 1,
          'you': 1,
          'tomorrow': 1})]
```

In [21]:

```
from sklearn.feature_extraction.text import CountVectorizer
count_vector=CountVectorizer()
```

In [22]:

```
count_vector.fit(documents)
count_vector.get_feature_names_out()
```

Out[22]:

```
array(['are', 'call', 'from', 'hello', 'home', 'how', 'me', 'money',
       'now', 'tomorrow', 'win', 'you'], dtype=object)
```

In [23]:

```
doc_array=count_vector.transform(documents).toarray()
doc_array
```

Out[23]:

```
array([[1, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1],
       [0, 0, 1, 0, 1, 0, 0, 1, 0, 0, 2, 0],
       [0, 2, 0, 2, 0, 0, 1, 0, 1, 1, 0, 1]], dtype=int64)
```

In [24]:

```
frequency_matrix=pd.DataFrame(doc_array,columns=count_vector.get_feature_names_out())
frequency_matrix
```

Out[24]:

	are	call	from	hello	home	how	me	money	now	tomorrow	win	you
0	1	0	0	1	0	1	0	0	0	0	0	1
1	0	0	1	0	1	0	0	1	0	0	2	0
2	0	2	0	2	0	0	1	0	1	1	0	1

In [25]:

```
from sklearn.model_selection import train_test_split

X_train,X_test,y_train,y_test=train_test_split(df_sms['sms-text'],df_sms['label'],test_size
```

In [26]:

```
count_vector=CountVectorizer()
```

In [27]:

```
training_data=count_vector.fit_transform(X_train)
```

In [28]:

```
training_data
```

Out[28]:

```
<4457x7733 sparse matrix of type '<class 'numpy.int64'>'
	with 59215 stored elements in Compressed Sparse Row format>
```

In [29]:

```
testing_data=count_vector.transform(X_test)
```

In [30]:

```
from sklearn.naive_bayes import MultinomialNB
naive_bayes=MultinomialNB()
naive_bayes.fit(training_data,y_train)
```

Out[30]:

```
MultinomialNB()
```

In [31]:

```
predictions=naive_bayes.predict(testing_data)
```

In [32]:

```
predictions
```

Out[32]:

```
array(['ham', 'ham', 'ham', ..., 'ham', 'ham', 'ham'], dtype='<U4')
```

In [33]:

```
from sklearn.metrics import accuracy_score,precision_score,recall_score,f1_score
print('Accuracy score:{}'.format(accuracy_score(y_test,predictions)))
```

```
Accuracy score:0.9856502242152466
```

In [34]:

```
print('Precision score:{}'.format(precision_score(y_test,predictions,pos_label='spam')))
print('Recall score:{}'.format(recall_score(y_test,predictions,pos_label='spam')))
print('f1 score:{}'.format(f1_score(y_test,predictions,pos_label='spam')))
```

```
Precision score:0.9424460431654677
Recall score:0.9424460431654677
f1 score:0.9424460431654677
```

In [35]:

```
from sklearn.metrics import classification_report
print(classification_report(predictions,y_test))
```

```
              precision    recall  f1-score   support

         ham       0.99      0.99      0.99       976
        spam       0.94      0.94      0.94       139

    accuracy                           0.99      1115
   macro avg       0.97      0.97      0.97      1115
weighted avg       0.99      0.99      0.99      1115
```

In [44]:

```
from sklearn.naive_bayes import MultinomialNB
spam_filter=MultinomialNB()
predictions=spam_filter.fit(training_data,y_train)
```

In [45]:

```
predictions=spam_filter.predict(testing_data)
```

In [46]:

```
count=0
for i in range(len(y_test)):
    if y_test.iloc[i] !=predictions[i]:
        count+=1
print('Total number of test cases',len(y_test))
print('Number of wrong predictions',count)
```

```
Total number of test cases 1115
Number of wrong predictions 16
```

In [47]:

```
from sklearn.model_selection import cross_val_score
model=MultinomialNB()
scores=cross_val_score(model,X_train,y_train,scoring='accuracy',cv=5,n_jobs=-1)
```

```
c:\users\dell\appdata\local\programs\python\python39\lib\site-packages\sklea
rn\model_selection\_validation.py:372: FitFailedWarning:
5 fits failed out of a total of 5.
The score on these train-test partitions for these parameters will be set to
nan.
If these failures are not expected, you can try to debug them by setting err
or_score='raise'.

Below are more details about the failures:
----------------------------------------------------------------------------
----
1 fits failed with the following error:
Traceback (most recent call last):
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\model_selection\_validation.py", line 681, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\naive_bayes.py", line 663, in fit
    X, y = self._check_X_y(X, y)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\naive_bayes.py", line 523, in _check_X_y
    return self._validate_data(X, y, accept_sparse="csr", reset=reset)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\base.py", line 572, in _validate_data
    X, y = check_X_y(X, y, **check_params)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\utils\validation.py", line 956, in check_X_y
    X = check_array(
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\utils\validation.py", line 738, in check_array
    array = np.asarray(array, order=order, dtype=dtype)
  File "C:\Users\DELL\AppData\Roaming\Python\Python39\site-packages\numpy\co
re\_asarray.py", line 83, in asarray
    return array(a, dtype, copy=False, order=order)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\pandas\core\series.py", line 768, in __array__
    return np.asarray(self.array, dtype)
  File "C:\Users\DELL\AppData\Roaming\Python\Python39\site-packages\numpy\co
re\_asarray.py", line 83, in asarray
    return array(a, dtype, copy=False, order=order)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\pandas\core\arrays\numpy_.py", line 203, in __array__
    return np.asarray(self._ndarray, dtype=dtype)
  File "C:\Users\DELL\AppData\Roaming\Python\Python39\site-packages\numpy\co
re\_asarray.py", line 83, in asarray
    return array(a, dtype, copy=False, order=order)
ValueError: could not convert string to float: 'Free Msg: get Gnarls Barkley
s \\Crazy\\" ringtone TOTALLY FREE just reply GO to this message right no
w!"'

----------------------------------------------------------------------------
----
4 fits failed with the following error:
Traceback (most recent call last):
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
```

```
es\sklearn\model_selection\_validation.py", line 681, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\naive_bayes.py", line 663, in fit
    X, y = self._check_X_y(X, y)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\naive_bayes.py", line 523, in _check_X_y
    return self._validate_data(X, y, accept_sparse="csr", reset=reset)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\base.py", line 572, in _validate_data
    X, y = check_X_y(X, y, **check_params)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\utils\validation.py", line 956, in check_X_y
    X = check_array(
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\sklearn\utils\validation.py", line 738, in check_array
    array = np.asarray(array, order=order, dtype=dtype)
  File "C:\Users\DELL\AppData\Roaming\Python\Python39\site-packages\numpy\co
re\_asarray.py", line 83, in asarray
    return array(a, dtype, copy=False, order=order)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\pandas\core\series.py", line 768, in __array__
    return np.asarray(self.array, dtype)
  File "C:\Users\DELL\AppData\Roaming\Python\Python39\site-packages\numpy\co
re\_asarray.py", line 83, in asarray
    return array(a, dtype, copy=False, order=order)
  File "c:\users\dell\appdata\local\programs\python\python39\lib\site-packag
es\pandas\core\arrays\numpy_.py", line 203, in __array__
    return np.asarray(self._ndarray, dtype=dtype)
  File "C:\Users\DELL\AppData\Roaming\Python\Python39\site-packages\numpy\co
re\_asarray.py", line 83, in asarray
    return array(a, dtype, copy=False, order=order)
ValueError: could not convert string to float: 'Sleeping nt feeling well'

  warnings.warn(some_fits_failed_message, FitFailedWarning)
```

In [48]:

```
scores
```

Out[48]:

```
array([nan, nan, nan, nan, nan])
```

In [ ]:

```
```