:[5] In

```
from platform import python_version
print(python_version())
```

```
3.7.4
```

:[6] In

```
import numpy
numpy.version.version
```

Out[6]:

```
'1.16.5'
```

:[7] In

```
import gensim as gs
print(gs.__version__)
```

```
3.8.3
```

!Exercise 2: Tokenize the following sentence and write down the resu you obtain

:[8] In

```
gs.utils.tokenize
help(gs.utils.tokenize)
```

```
:Help on function tokenize in module gensim.utils

tokenize(text, lowercase=False, deacc=False, encoding='utf8', errors='stric
(t', to_lower=False, lower=False
 Iteratively yield tokens as unicode strings, optionally removing accent
.marks and lowercasing it

Parameters
----------
text : str or bytes
.Input string
deacc : bool, optional
?`Remove accentuation using :func:`~gensim.utils.deaccent
encoding : str, optional
Encoding of input string, used as parameter for :func:`~gensim.util
.`s.to_unicode
errors : str, optional
Error handling behaviour, used as parameter for :func:`~gensim.util
.`s.to_unicode
lowercase : bool, optional
?Lowercase the input string
to_lower : bool, optional
.Same as `lowercase`. Convenience alias
lower : bool, optional
.Same as `lowercase`. Convenience alias

Yields
------
str
Contiguous sequences of alphabetic characters (no digits!), using :f
`unc:`~gensim.utils.simple_tokenize

Examples
--------
sourcecode:: pycon ..

from gensim.utils import tokenize <<<
list(tokenize('Nic nemůže letět rychlostí vyšší, než 300 tisíc k <<<
((ilometrů za sekundu!', deacc=True
u'Nic', u'nemuze', u'letet', u'rychlosti', u'vyssi', u'nez', u'tisi]
['c', u'kilometru', u'za', u'sekundu
```

:[9] In

```
Sentence= '''Tokenization is the process of breaking
down text documentapart into those pieces'''
import gensim as gs
tokenizedWord= list(gs.utils.tokenize(Sentence))
tokenizedWord
```

Out[9]:

```
             ,'Tokenization']
             ,'is'
             ,'the'
             ,'process'
             ,'of'
             ,'breaking'
             ,'down'
             ,'text'
             ,'documentapart'
             ,'into'
             ,'those'
             ['pieces'
```

.Exercise 3: Count frequency of each word

:[10] In

```
document = ''' In computer science, artificial
intelligence (AI), sometimes called machine
intelligence, is intelligence demonstrated by
machines, in contrast to the natural intelligence
displayed by humans and animals. Computer science
defines AI research as the study of intelligent
agents: any device that perceives its environment and
takes actions that maximize its chance of successfully
achieving its goals.”
'''
import gensim
from gensim import corpora
from pprint import pprint
text = [document]
tokens = [[token for token in sentence.split()] for sentence
in text]
gensim_dictionary = corpora.Dictionary()
gensim_corpus = [gensim_dictionary.doc2bow(token,
allow_update=True) for token in tokens]
print(gensim_corpus)
```

```
 ,(1 ,8) ,(2 ,7) ,(1 ,6) ,(1 ,5) ,(1 ,4) ,(1 ,3) ,(1 ,2) ,(1 ,1) ,(1 ,0)]]
 ,17) ,(1 ,16) ,(1 ,15) ,(1 ,14) ,(1 ,13) ,(2 ,12) ,(1 ,11) ,(1 ,10) ,(1 ,9)
 ,(3 ,25) ,(1 ,24) ,(1 ,23) ,(1 ,22) ,(1 ,21) ,(1 ,20) ,(1 ,19) ,(1 ,18) ,(1
 ,34) ,(1 ,33) ,(1 ,32) ,(1 ,31) ,(1 ,30) ,(3 ,29) ,(1 ,28) ,(1 ,27) ,(1 ,26)
 ,(1 ,42) ,(1 ,41) ,(1 ,40) ,(1 ,39) ,(1 ,38) ,(1 ,37) ,(1 ,36) ,(1 ,35) ,(2
[[(1 ,45) ,(2 ,44) ,(2 ,43)
```

:[11] In

```
word_frequencies = [[(gensim_dictionary[id], frequence)
 for id, frequence in couple]
 for couple in gensim_corpus]
print(word_frequencies)
```

```
AI),', 1), ('AI', 1), ('Computer', 1), ('In', 1), ('achieving', 1), ('a)')]]
ctions', 1), ('agents:', 1), ('and', 2), ('animals.', 1), ('any', 1), ('arti
ficial', 1), ('as', 1), ('by', 2), ('called', 1), ('chance', 1), ('compute
 r', 1), ('contrast', 1), ('defines', 1), ('demonstrated', 1), ('device', 1),
 ('displayed', 1), ('environment', 1), ('goals.”', 1), ('humans', 1), ('in',
 1), ('intelligence', 3), ('intelligence,', 1), ('intelligent', 1), ('is',
1), ('its', 3), ('machine', 1), ('machines,', 1), ('maximize', 1), ('natura
l', 1), ('of', 2), ('perceives', 1), ('research', 1), ('science', 1), ('scie
 nce,', 1), ('sometimes', 1), ('study', 1), ('successfully', 1), ('takes',
[[(1), ('that', 2), ('the', 2), ('to', 1
```

:[17] In

```
from gensim.utils import simple_preprocess
from smart_open import smart_open
import os

tokens = [simple_preprocess(sentence, deacc=True) for sentence in open(r'D:\\file1.txt', er

gensim_dictionary = corpora.Dictionary()
gensim_corpus = [gensim_dictionary.doc2bow(token, allow_update=True) for token in tokens]
word_frequencies = [[(gensim_dictionary[id], frequence) for id, frequence in couple] for cc

print(word_frequencies)
```

```
[[(my', 1), ('teacher', 1), ('wolcame', 1')]]
```

:[ ] In

```

```