

# Lab Assignment - 1

## Reg ex

```
In [1]: a=5
```

```
In [13]: import re
```

```
In [14]: allstr= "sat,hat,mat,pat"
```

```
In [22]: re.findall(r'[a-m]',allstr)
```

```
Out[22]: ['a', 'h', 'a', 'm', 'a', 'a']
```

```
In [23]: a = "vai abc bdh irs ola ubol vid asc"
```

```
In [37]: re.findall(r'\w\w',a)
```

```
Out[37]: ['va', 'ab', 'bd', 'ir', 'ol', 'ub', 'ol', 'vi', 'as']
```

```
In [50]: re.findall(r'\b\w\w',a)
```

```
Out[50]: ['va', 'ab', 'bd', 'ir', 'ol', 'ub', 'vi', 'as']
```

```
In [38]: re.findall(r'\b(\w{1,2})\w*\b',a)
```

```
Out[38]: ['va', 'ab', 'bd', 'ir', 'ol', 'ub', 'vi', 'as']
```

```
In [51]: s = '''abc@gmail.com , pqr@gmail.com , step@gmail.in , vaibhav@email.edu'''
```

```
In [54]: re.findall(r'^[a-zA-Z0-9]+@[a-zA-Z0-9]+\.[a-zA-Z]{2,}$',s)
```

```
Out[54]: []
```

```
In [55]: re.findall(r'^\w+@\w+\.[a-zA-Z]{2,}$',"abc@gmail.com")
```

```
Out[55]: ['abc@gmail.com']
```

```
In [59]: re.findall(r'^\w+@\w+\.[a-zA-Z]{2,}$')
```

```
Out[59]: []
```

```
In [67]: re.findall(r'\@w+\.(\w+)',s)
```

```
Out[67]: ['com', 'com', 'in', 'edu']
```

```
In [69]: b= 'virat kohli is scoring centuries like flowing water in river'
```

```
In [70]: b.split(" ")
```

```
Out[70]: ['virat',  
          'kohli',  
          'is',  
          'scoring',  
          'centuries',  
          'like',  
          'flowing',  
          'water',  
          'in',  
          'river']
```

```
In [74]: p= "This is a sample paragraph. It contains multiple sentences. Each sentence ends with a p
```

```
In [75]: re.split(r'(?<!\w\.\w.)(?<![A-Z][a-z]\.)(?<=\.|\\?)\s', p)
```

```
Out[75]: ['This is a sample paragraph.',  
          'It contains multiple sentences.',  
          'Each sentence ends with a period.',  
          'mr.x is very happy!']
```

```
In [76]: re.split(r'(?<=[.!?])\s+', p)
```

```
Out[76]: ['This is a sample paragraph.',  
          'It contains multiple sentences.',  
          'Each sentence ends with a period.',  
          'mr.x is very happy!']
```

```
In [80]: re.split(r'(?<=[.!?])', p)
```

```
Out[80]: ['This is a sample paragraph.',  
          ' It contains multiple sentences.',  
          ' Each sentence ends with a period.',  
          ' mr.',  
          'x is very happy!',  
          '']
```

```
In [81]: re.split(r'(?<!\w\.\w.)(?<![A-Z][a-z]\.)(?<=\.|\\?)', p)
```

```
Out[81]: ['This is a sample paragraph.',  
          ' It contains multiple sentences.',  
          ' Each sentence ends with a period.',  
          ' mr.',  
          'x is very happy!']
```

```
In [82]: re.split(r'([.!?])', p)
```

```
Out[82]: ['This is a sample paragraph',  
          '.',  
          ' It contains multiple sentences',  
          '.',  
          ' Each sentence ends with a period',  
          '.',  
          ' mr',  
          '.',  
          'x is very happy',  
          '!',  
          '']
```

```
In [83]: re.split(r'[^.!?]', p)
```

```
Out[83]: ['This is a sample paragraph',  
          ' It contains multiple sentences',  
          ' Each sentence ends with a period',  
          ' mr',  
          'x is very happy',  
          '']
```

```
In [84]: re.split(r'[^.!?]\s', p)
```

```
Out[84]: ['This is a sample paragraph',  
          'It contains multiple sentences',  
          'Each sentence ends with a period',  
          'mr.x is very happy!']
```

```
In [105]: d = "This is a sample paragraph. It contains multiple sentences. Each sentence ends with a
```

```
In [106]: re.split(r'[^.!?]\s+', d)
```

```
Out[106]: ['This is a sample paragraph',  
          'It contains multiple sentences',  
          'Each sentence ends with a period',  
          'Mr',  
          'Doctor is very happy!']
```

```
In [107]: re.split(r'(?<!\w\.\w.)(?<![A-Z][a-z]\.)(?<=\.|\?|!)\s', d)
```

```
Out[107]: ['This is a sample paragraph.',  
          'It contains multiple sentences.',  
          'Each sentence ends with a period.',  
          'Mr. Doctor is very happy!']
```

```
In [108]: d.split(r'(?<=\.|!|?)\s+|\s+(?=[A-Z](?<![a-z]))')
```

```
Out[108]: ['This is a sample paragraph. It contains multiple sentences. Each sentence ends with a pe  
riod. Mr. Doctor is very happy!']
```

```
In [109]: re.split(r'(?<!\w\.\w.)(?<![A-Z][a-z]\.)(?<=\.|\?|!)\s', d)
```

```
Out[109]: ['This is a sample paragraph.',  
          'It contains multiple sentences.',  
          'Each sentence ends with a period.',  
          'Mr. Doctor is very happy!']
```

```
In [110]: re.split(r'(?<=\.|!|?)\s+', d)
```

```
Out[110]: ['This is a sample paragraph.',  
          'It contains multiple sentences.',  
          'Each sentence ends with a period.',  
          'Mr.',  
          'Doctor is very happy!']
```

```
In [111]: import os
```

```
In [ ]:
```

```
In [115]: with open(r"C:\Users\raval\Downloads\stopwords-en.txt", 'r', encoding='utf-8') as file:
          st = file.read()
          print(st)
```

```
'll
'tis
'twas
've
10
39
a
a's
able
ableabout
about
above
abroad
abst
accordance
according
accordingly
across
act
+ ..
```

```
In [118]: # Sample paragraph
paragraph = """
Natural language processing (NLP) is a subfield of artificial intelligence content improve
"""

stopwords_list = st.split('\n')

stopwords_list = [stopword.strip() for stopword in stopwords_list if stopword.strip()]

words_in_paragraph = paragraph.split()

non_stopwords_list = [word for word in words_in_paragraph if word.lower() not in stopwords_

print("\nNon-Stopwords:", non_stopwords_list)
```

```
Non-Stopwords: ['Natural', 'language', 'processing', '(NLP)', 'subfield', 'artificial', 'i
ntelligence', 'content', 'improve', 'efficiency', 'algorithms', 'highlight', 'informatio
n.', 'create', 'list', 'stopwords', 'filter', 'sample', 'paragraph.']
```

```
In [117]: # Sample paragraph
paragraph = """
Natural language processing (NLP) is a subfield of artificial intelligence content improve
"""

# Split the content into a list of stopwords
stopwords_list = st.split('\n')

# Remove any empty strings or whitespaces from the list
stopwords_list = [stopword.strip() for stopword in stopwords_list if stopword.strip()]

# Split the paragraph into words
words_in_paragraph = paragraph.split()

# Create a list of non-stopwords
non_stopwords_list = [word for word in words_in_paragraph if word.lower() not in stopwords_]

# Print the list of stopwords and non-stopwords
print("Stopwords:", stopwords_list)
print("\nNon-Stopwords:", non_stopwords_list)
```

Stopwords: ['ll', 'tis', 'twas', 've', '10', '39', 'a', 'a's', 'able', 'ableabout', 'a  
bout', 'above', 'abroad', 'abst', 'accordance', 'according', 'accordingly', 'across', 'ac  
t', 'actually', 'ad', 'added', 'adj', 'adopted', 'ae', 'af', 'affected', 'affecting', 'aff  
ects', 'after', 'afterwards', 'ag', 'again', 'against', 'ago', 'ah', 'ahead', 'ai', 'ai  
n't', 'aint', 'al', 'all', 'allow', 'allows', 'almost', 'alone', 'along', 'alongside', 'al  
ready', 'also', 'although', 'always', 'am', 'amid', 'amidst', 'among', 'amongst', 'amouns  
t', 'amount', 'an', 'and', 'announce', 'another', 'any', 'anybody', 'anyhow', 'anymore',  
'anyone', 'anything', 'anyway', 'anyways', 'anywhere', 'ao', 'apart', 'apparently', 'appea  
r', 'appreciate', 'appropriate', 'approximately', 'aq', 'ar', 'are', 'area', 'areas', 'are  
n', 'aren't', 'arent', 'arise', 'around', 'arpa', 'as', 'aside', 'ask', 'asked', 'asking',  
'asks', 'associated', 'at', 'au', 'auth', 'available', 'aw', 'away', 'awfully', 'az', 'b',  
'ba', 'back', 'backed', 'backing', 'backs', 'backward', 'backwards', 'bb', 'bd', 'be', 'be  
came', 'because', 'become', 'becomes', 'becoming', 'been', 'before', 'beforehand', 'bega  
n', 'begin', 'beginning', 'beginnings', 'begins', 'behind', 'being', 'beings', 'believe',  
'below', 'beside', 'besides', 'best', 'better', 'between', 'beyond', 'bf', 'bg', 'bh', 'b  
i', 'big', 'bill', 'billion', 'biol', 'bj', 'bm', 'bn', 'bo', 'both', 'bottom', 'br', 'bri  
ef', 'briefly', 'bs', 'bt', 'but', 'buy', 'bv', 'bw', 'by', 'bz', 'c', 'c'mon', 'c's', 'c  
a', 'call', 'came', 'can', 'can't', 'cannot', 'cant', 'caption', 'case', 'cases', 'cause',  
'causes', 'cc', 'cd', 'certain', 'certainly', 'cf', 'cg', 'ch', 'changes', 'ci', 'ck', 'c  
l', 'clear', 'clearly', 'click', 'cm', 'cmon', 'cn', 'co', 'co.', 'com', 'come', 'comes',  
'computer', 'con', 'concerning', 'consequently', 'consider', 'considering', 'contain', 'co  
ntaining', 'contains', 'copy', 'corresponding', 'could', 'could've', 'couldn', 'couldn't',  
'couldnt', 'course', 'cr', 'cry', 'cs', 'cu', 'currently', 'cv', 'cx', 'cy', 'cz', 'd', 'd  
are', 'daren't', 'darent', 'date', 'de', 'dear', 'definitely', 'describe', 'described', 'd  
espite', 'detail', 'did', 'didn', 'didn't', 'didn't', 'differ', 'different', 'differently',  
'directly', 'dj', 'dk', 'dm', 'do', 'does', 'doesn', 'doesn't', 'doesnt', 'doing', 'don',  
'don't', 'done', 'dont', 'doubtful', 'down', 'downed', 'downing', 'downs', 'downwards', 'd  
ue', 'during', 'dz', 'e', 'each', 'early', 'ec', 'ed', 'edu', 'ee', 'effect', 'eg', 'eh',  
'eight', 'eighty', 'either', 'eleven', 'else', 'elsewhere', 'empty', 'end', 'ended', 'endi  
ng', 'ends', 'enough', 'entirely', 'er', 'es', 'especially', 'et', 'et-al', 'etc', 'even',  
'evenly', 'ever', 'evermore', 'every', 'everybody', 'everyone', 'everything', 'everywher  
e', 'ex', 'exactly', 'example', 'except', 'f', 'face', 'faces', 'fact', 'facts', 'fairly',  
'far', 'farther', 'felt', 'few', 'fewer', 'ff', 'fi', 'fifteen', 'fifth', 'fifty', 'fify',  
'fill', 'find', 'finds', 'fire', 'first', 'five', 'fix', 'fj', 'fk', 'fm', 'fo', 'followe  
d', 'following', 'follows', 'for', 'forever', 'former', 'formerly', 'forth', 'forty', 'for  
ward', 'found', 'four', 'fr', 'free', 'from', 'front', 'full', 'fully', 'further', 'furthe  
red', 'furthering', 'furthermore', 'furthers', 'fx', 'g', 'ga', 'gave', 'gb', 'gd', 'ge',  
'general', 'generally', 'get', 'gets', 'getting', 'gf', 'gg', 'gh', 'gi', 'give', 'given',  
'gives', 'giving', 'gl', 'gm', 'gmt', 'gn', 'go', 'goes', 'going', 'gone', 'good', 'good  
s', 'got', 'gotten', 'gov', 'gp', 'gq', 'gr', 'great', 'greater', 'greatest', 'greetings',  
'group', 'grouped', 'grouping', 'groups', 'gs', 'gt', 'gu', 'gw', 'gy', 'h', 'had', 'had  
n't', 'hadnt', 'half', 'happens', 'hardly', 'has', 'hasn', 'hasn't', 'hasnt', 'have', 'hav  
en', 'haven't', 'havent', 'having', 'he', 'he'd', 'he'll', 'he's', 'hed', 'hell', 'hello',  
'help', 'hence', 'her', 'here', 'here's', 'hereafter', 'hereby', 'herein', 'heres', 'hereu  
pon', 'hers', 'herself', 'herse', 'hes', 'hi', 'hid', 'high', 'higher', 'highest', 'him',  
'himself', 'himse', 'his', 'hither', 'hk', 'hm', 'hn', 'home', 'homepage', 'hopefully',  
'how', 'how'd', 'how'll', 'how's', 'howbeit', 'however', 'hr', 'ht', 'htm', 'html', 'htt  
p', 'hu', 'hundred', 'i', 'i'd', 'i'll', 'i'm', 'i've', 'i.e.', 'id', 'ie', 'if', 'ignore  
d', 'ii', 'il', 'ill', 'im', 'immediate', 'immediately', 'importance', 'important', 'in',  
'inasmuch', 'inc', 'inc.', 'indeed', 'index', 'indicate', 'indicated', 'indicates', 'infor  
mation', 'inner', 'inside', 'insofar', 'instead', 'int', 'interest', 'interested', 'intere  
sting', 'interests', 'into', 'invention', 'inward', 'io', 'iq', 'ir', 'is', 'isn', 'is  
n't', 'isnt', 'it', 'it'd', 'it'll', 'it's', 'itd', 'itll', 'its', 'itself', 'itse', 'iv  
e', 'j', 'je', 'jm', 'jo', 'join', 'jp', 'just', 'k', 'ke', 'keep', 'keeps', 'kept', 'key  
s', 'kg', 'kh', 'ki', 'kind', 'km', 'kn', 'knew', 'know', 'known', 'knows', 'kp', 'kr', 'k  
w', 'ky', 'kz', 'l', 'la', 'large', 'largely', 'last', 'lately', 'later', 'latest', 'latte  
r', 'latterly', 'lb', 'lc', 'least', 'length', 'less', 'lest', 'let', 'let's', 'lets', 'l  
i', 'like', 'liked', 'likely', 'likewise', 'line', 'little', 'lk', 'll', 'long', 'longer',  
'longest', 'look', 'looking', 'looks', 'low', 'lower', 'lr', 'ls', 'lt', 'ltd', 'lu', 'l  
v', 'ly', 'm', 'ma', 'made', 'mainly', 'make', 'makes', 'making', 'man', 'many', 'may', 'm  
aybe', 'mayn't', 'maynt', 'mc', 'md', 'me', 'mean', 'means', 'meantime', 'meanwhile', 'mem  
ber', 'members', 'men', 'merely', 'mg', 'mh', 'microsoft', 'might', 'might've', 'might  
n't', 'mightnt', 'mil', 'mill', 'million', 'mine', 'minus', 'miss', 'mk', 'ml', 'mm', 'm  
n', 'mo', 'more', 'moreover', 'most', 'mostly', 'move', 'mp', 'mq', 'mr', 'mrs', 'ms', 'ms  
ie', 'mt', 'mu', 'much', 'mug', 'must', 'must've', 'mustn't', 'mustnt', 'mv', 'mw', 'mx',  
'my', 'myself', 'myse', 'mz', 'n', 'na', 'name', 'namely', 'nay', 'nc', 'nd', 'ne', 'nea  
r', 'nearly', 'necessarily', 'necessary', 'need', 'needed', 'needing', 'needn't', 'needn  
t', 'needs', 'neither', 'net', 'netscape', 'never', 'neverf', 'neverless', 'nevertheless',  
'new', 'newer', 'newest', 'next', 'nf', 'ng', 'ni', 'nine', 'ninety', 'nl', 'no', 'no-on  
e', 'nobody', 'non', 'none', 'nonetheless', 'noone', 'nor', 'normally', 'nos', 'not', 'not  
ed', 'nothing', 'notwithstanding', 'novel', 'now', 'nowhere', 'np', 'nr', 'nu', 'null', 'n  
umber', 'numbers', 'nz', 'o', 'obtain', 'obtained', 'obviously', 'of', 'off', 'often', 'o

h', 'ok', 'okay', 'old', 'older', 'oldest', 'om', 'omitted', 'on', 'once', 'one', 'one's', 'ones', 'only', 'onto', 'open', 'opened', 'opening', 'opens', 'opposite', 'or', 'ord', 'order', 'ordered', 'ordering', 'orders', 'org', 'other', 'others', 'otherwise', 'ought', 'oughtn't', 'oughtnt', 'our', 'ours', 'ourselves', 'out', 'outside', 'over', 'overall', 'owing', 'own', 'p', 'pa', 'page', 'pages', 'part', 'parted', 'particular', 'particularly', 'parting', 'parts', 'past', 'pe', 'per', 'perhaps', 'pf', 'pg', 'ph', 'pk', 'pl', 'place', 'placed', 'places', 'please', 'plus', 'pm', 'pmid', 'pn', 'point', 'pointed', 'pointing', 'points', 'poorly', 'possible', 'possibly', 'potentially', 'pp', 'pr', 'predominantly', 'present', 'presented', 'presenting', 'presents', 'presumably', 'previously', 'primarily', 'probably', 'problem', 'problems', 'promptly', 'proud', 'provided', 'provides', 'pt', 'put', 'puts', 'pw', 'py', 'q', 'qa', 'que', 'quickly', 'quite', 'qv', 'r', 'ran', 'rather', 'rd', 're', 'readily', 'really', 'reasonably', 'recent', 'recently', 'ref', 'refs', 'regarding', 'regardless', 'regards', 'related', 'relatively', 'research', 'reserved', 'respectively', 'resulted', 'resulting', 'results', 'right', 'ring', 'ro', 'room', 'rooms', 'round', 'ru', 'run', 'rw', 's', 'sa', 'said', 'same', 'saw', 'say', 'saying', 'says', 'sb', 'sc', 'sd', 'se', 'sec', 'second', 'secondly', 'seconds', 'section', 'see', 'seeing', 'seem', 'seemed', 'seeming', 'seems', 'seen', 'sees', 'self', 'selves', 'sensible', 'sent', 'serious', 'seriously', 'seven', 'seventy', 'several', 'sg', 'sh', 'shall', 'shan't', 'shant', 'she', 'she'd', 'she'll', 'she's', 'shed', 'shell', 'shes', 'should', 'should've', 'shouldn', 'shouldn't', 'shouldnt', 'show', 'showed', 'showing', 'shown', 'shows', 'shows', 'si', 'side', 'sides', 'significant', 'significantly', 'similar', 'similarly', 'since', 'sincere', 'site', 'six', 'sixty', 'sj', 'sk', 'sl', 'slightly', 'sm', 'small', 'smaller', 'smallest', 'sn', 'so', 'some', 'somebody', 'someday', 'somehow', 'someone', 'somethan', 'something', 'sometime', 'sometimes', 'somewhat', 'somewhere', 'soon', 'sorry', 'specifically', 'specified', 'specify', 'specifying', 'sr', 'st', 'state', 'states', 'still', 'stop', 'strongly', 'su', 'sub', 'substantially', 'successfully', 'such', 'sufficiently', 'suggested', 'sup', 'sure', 'sv', 'sy', 'system', 'sz', 't', 't's', 'take', 'taken', 'taking', 'tc', 'td', 'tell', 'ten', 'tends', 'test', 'text', 'tf', 'tg', 'th', 'than', 'thank', 'thanks', 'thanx', 'that', 'that'll', 'that's', 'that've', 'that'll', 'thats', 'thatve', 'the', 'their', 'theirs', 'them', 'themselves', 'then', 'thence', 'there', 'there'd', 'there'll', 'there're', 'there's', 'there've', 'thereafter', 'thereby', 'thered', 'therefore', 'theren', 'therell', 'thereof', 'therere', 'theres', 'thereto', 'thereupon', 'thereve', 'these', 'they', 'they'd', 'they'll', 'they're', 'they've', 'theyd', 'theyll', 'theyre', 'theyve', 'thick', 'thin', 'thing', 'things', 'think', 'thinks', 'third', 'thirty', 'this', 'thorough', 'thoroughly', 'those', 'thou', 'though', 'thoughh', 'thought', 'thoughts', 'thousand', 'three', 'through', 'through', 'throughout', 'thru', 'thus', 'til', 'till', 'tip', 'tis', 'tj', 'tk', 'tm', 'tn', 'to', 'today', 'together', 'too', 'took', 'top', 'toward', 'towards', 'tp', 'tr', 'tried', 'tries', 'trillion', 'truly', 'try', 'trying', 'ts', 'tt', 'turn', 'turned', 'turning', 'turns', 'tv', 'tw', 'twas', 'twelve', 'twenty', 'twice', 'two', 'tz', 'u', 'ua', 'ug', 'uk', 'um', 'un', 'under', 'underneath', 'undoing', 'unfortunately', 'unless', 'unlikely', 'unlikely', 'until', 'unto', 'up', 'upon', 'ups', 'upwards', 'us', 'use', 'used', 'useful', 'usefully', 'usefulness', 'uses', 'using', 'usually', 'uucp', 'uy', 'uz', 'v', 'va', 'value', 'various', 'vc', 've', 'versus', 'very', 'vg', 'vi', 'via', 'viz', 'vn', 'vol', 'vols', 'vs', 'vu', 'w', 'want', 'wanted', 'wanting', 'wants', 'was', 'wasn', 'wasn't', 'wasnt', 'way', 'ways', 'we', 'we'd', 'we'll', 'we're', 'we've', 'web', 'webpage', 'website', 'wed', 'welcome', 'well', 'wells', 'went', 'were', 'weren', 'weren't', 'werent', 'weve', 'wf', 'what', 'what'd', 'what'll', 'what's', 'what've', 'whatever', 'whatll', 'whats', 'whatve', 'when', 'when'd', 'when'll', 'when's', 'whence', 'whenever', 'where', 'where'd', 'where'll', 'where's', 'whereafter', 'whereas', 'whereby', 'wherein', 'wheres', 'whereupon', 'wherever', 'whether', 'which', 'whichever', 'while', 'whilst', 'whim', 'whither', 'who', 'who'd', 'who'll', 'who's', 'whod', 'whoever', 'whole', 'wholl', 'whom', 'whomever', 'whos', 'whose', 'why', 'why'd', 'why'll', 'why's', 'widely', 'width', 'will', 'willing', 'wish', 'with', 'within', 'without', 'won', 'won't', 'wonder', 'wont', 'words', 'work', 'worked', 'working', 'works', 'world', 'would', 'would've', 'wouldn', 'wouldn't', 'wouldnt', 'ws', 'www', 'x', 'y', 'ye', 'year', 'years', 'yes', 'yet', 'you', 'you'd', 'you'll', 'you're', 'you've', 'youd', 'youll', 'young', 'younger', 'youngest', 'your', 'you're', 'yours', 'yourself', 'yourselves', 'youve', 'yt', 'yu', 'z', 'za', 'zero', 'zm', 'zr']

Non-Stopwords: ['Natural', 'language', 'processing', '(NLP)', 'subfield', 'artificial', 'intelligence', 'content', 'improve', 'efficiency', 'algorithms', 'highlight', 'information.', 'create', 'list', 'stopwords', 'filter', 'sample', 'paragraph.']

In [ ]:

