**School of Computer Science Engineering and Technology**

|  |  |
| --- | --- |
| Course- M. Tech / MCA | Type- Elective |
| Course Code- CSET546 / CMCA546 | Course Name: Natural Language Processing |
| Year- 2025 | Semester- odd |
| Date: 29-07-2025 | Batch - **B2** |

**Lab Assignment – 01: Extracting the patterns**

The main objective of this assignment is to write regular expressions that extract the patterns given in each question.

You will explore the utility of ***re*** package- regular expression package.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

NOTE: Try to write function and call it. Don’t do hardcode. See the below example.

example: **Find hashtags from a sentence.**

**Method 1:** import re

sentence = "I am from #Bennett University"

result = re.findall(r'#\w+', sentence)

print(result)

**Method 2:** import re

def extract\_hashtags(text):

    return re.findall(r'#\w+', text)

sentence = "I am from #Bennett University"

h = extract\_hashtags(sentence)

print("hash tag found:", h)

Both methods will give same Output. But always try method 2.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. **Extract all domain names from a sentence (not full URL).**  
   **Example Input:** "Visit our portal at https://www.iitbhu.ac.in for details."  
   **Expected Output:** ['iitbhu.ac.in']
2. **Extract all Gmail and Yahoo email addresses.**  
   **Example Input:** "Send updates to abc123@yahoo.com and x.y.z@gmail.com"  
   **Expected Output:** ['abc123@yahoo.com', 'x.y.z@gmail.com']
3. **Find hashtags that include digits or underscores.**  
   **Example Input:** "Today's highlights: #IndvsPak\_2025 #Go\_India #Cricket123"  
   **Expected Output:** ['#IndvsPak\_2025', '#Go\_India', '#Cricket123']
4. **Extract all Twitter-style mentions ignoring special characters after @.**  
   **Example Input:** "@dr\_singh! you were amazing @Minister@India"  
   **Expected Output:** ['@dr\_singh', '@Minister']
5. **Extract all decimal numbers from a sentence.**  
   **Example Input:** "The price increased from 45.60 to 99.99 in 3 days"  
   **Expected Output:** ['45.60', '99.99', '3']
6. **Extract all special characters from a sentence.**  
   **Example Input:** "Whoa!! COVID-19 has changed the world @2020 #History"  
   **Expected Output:** ['!', '!', '-', '@', '#']
7. **Validate vehicle registration numbers (format: 2 uppercase letters, 2 digits, 2 uppercase letters, 4 digits).**  
   **Example Input 1:** "UP32GH1234" (Valid)  
   **Example Input 2:** "up32gh1234" (Invalid)  
   **Expected Output:** True or False
8. **Normalize text by replacing repeated characters with 1 occurrence.**  
   **Example Input:** "Ammmaaazing wooorrrkkk!!!"  
   **Expected Output:** "Amazing work!"
9. **Find 10-digit mobile numbers starting with 6, 7, 8, or 9 only if preceded by a '+' or '0'.**  
   **Example Input:** "Call me at +919990001234 or 09990004567"  
   **Expected Output:** ['+919990001234', '09990004567']
10. **Extract all proper nouns (capitalized words not at the beginning of a sentence).**  
    **Example Input:** "Delhi is the capital of India. Narendra Modi is the Prime Minister."  
    **Expected Output:** ['India', 'Narendra', 'Modi', 'Prime', 'Minister']