|  |  |  |  |
| --- | --- | --- | --- |
| A picture containing drawing, stop, room  Description automatically generated | Applied Artificial Intelligence  Practical # 11 | | |
|  |  |  |  |
|  |  |  |  |
| **Name** | Ninad Karlekar | **Roll Number** | 22306A1012 |
| **Subject/Course:** | Applied Artificial Intelligence | **Class** | M.Sc. IT – Sem III |
| **Topic** | Language Parser | **Batch** | 1 |
|  |  |  |  |
| **Topic**: **Language Parser** | | | |
| 1. **AIM: Design an application to simulate language parser**   **Parsing is the process of analysing a sentence, breaking it down into smaller components, and identifying the grammatical structure of the sentence. It is a crucial component of NLP and helps machines understand human language.**  **Code:**  def sentenceSegment(text):  sentences = []  start = 0    for i in range(len(text)):  if text[i] == '.' or text[i] == '!' or text[i] == '?':  sentences.append(text[start:i+1].strip())  start = i + 1    return sentences    text = "Hello, NLP world!! In this example, we are going to do the basics of Text processing which will be used later."    print(sentenceSegment(text))    import nltk  nltk.download('punkt')    text = "Hello, NLP world!! In this example, we are going to do the basics of Text processing which will be used later."    sentences = nltk.sent\_tokenize(text)    print(sentences)    import string    def remove\_punctuation(input\_string):  # Define a string of punctuation marks and symbols  punctuations = string.punctuation    # Remove the punctuation marks and symbols from the input string  output\_string = "".join(char for char in input\_string if char not in punctuations)    return output\_string      text = "Hello, NLP world!! In this example, we are going to do the basics of Text processing which will be used later."  sentences = sentenceSegment(text)  puncRemovedText = remove\_punctuation(text)  print(puncRemovedText)    def convertToLower(s):  return s.lower()    text = "Hello, NLP world!! In this example, we are going to do the basics of Text processing which will be used later."  puncRemovedText = remove\_punctuation(text)    lowerText = convertToLower(puncRemovedText)  print(lowerText)    #in this code, we are not using any libraries  #tokenize without using any function from string or any other function.  #only using loops and if/else    def tokenize(s):  words = [] #token words should be stored here  i = 0  word = ""  while(i <len(s)):  if (s[i] != " "):  word = word+s[i]  else:  words.append(word)  word = ""    i = i + 1  words.append(word)  return words      text = "Hello, NLP world!! In this example, we are going to do the basics of Text processing which will be used later."  puncRemovedText = remove\_punctuation(text)  lowerText = convertToLower(puncRemovedText)    tokenizedText = tokenize(lowerText)  print(tokenizedText)    import nltk    # Define input text  text = "Hello, NLP world!! In this example, we are going to do the basics of Text processing which will be used later."    #sentence segmentation - removal of punctuations and converting to lowercase  sentences = nltk.sent\_tokenize(text)  puncRemovedText = remove\_punctuation(text)  lowerText = convertToLower(puncRemovedText)    # Tokenize the text  tokens = nltk.word\_tokenize(lowerText)    # Print the tokens  print(tokens)    import nltk    sentence = "We're going to John's house today."  tokens = nltk.word\_tokenize(sentence)    print(tokens) | | | |