

# Assignment

1. Use regular expression to change the input string to the output forms given below:

**Input text:** The quick brown fox jumps over the lazy dog.

**Output1:** The quick brown **wolf** jumps over the lazy dog.

**Output2:** **THE** quick brown **FOX** jumps over the lazy **DOG**.

2. Write a Python program with regex to remove the parenthesis area in a string.

**Input data :** ["example (.com)", "w3resource", "github (.com)", "stackoverflow (.com)"]

**Output:**

example

w3resource

github

stackoverflow

# Assignment

3. Write a regex script to split a string with multiple delimiters and output as a **list**.

**Input text:** The quick brown\nfox jumps\*over the lazy dog.

4. Write a regex script to find all three, four, five characters long words in a string and output it in a **list**.

**Input text:** The quick brown fox jumps over the lazy dog.

5. Write a Python program to find all adverbs and their positions in a given sentence and give its start and end position.

Use atleast re.findall and re.search. Is there any other way execute this?

**Input text :** text = "He was carefully disguised but captured quickly by police."

**Expected Output:**0-7: carefully (Hint: 2 adverbs)

# Assignment

6. In a imaginary Phone list

Not all entries contain a phone number,

If a phone number exists it is the first part of an entry, then separated by a blank a surname, which is followed by first names. Surname and first name are separated by a comma.

```
Input = ["555-8396 Neu, Allison",  
         "Burns, C. Montgomery",  
         "555-5299 Putz, Lionel",  
         "555-7334 Simpson, Homer Jay"]
```

**Output:**

Allison Neu 555-8396

C. Montgomery Burns

Lionel Putz 555-5299

Homer Jay Simpson 555-7334

# Assignment

7. Do the following:
  1. Use Biopython to read the DNA sequence from the file “Ciliate\_gene.fasta”,
  2. Perform translation using Biopython, by using the correct codon table for ciliates and report the final output (i.e., protein sequence) in the same .txt file.
  3. For the DNA sequence (read in step1): **Using regular expression**, count the frequency of occurrence of each codon and give its start and end positions in a **separate** table in the same .txt file.
  4. For the translated amino acid sequence: **Using regular expression**, count the frequency of occurrence of each codon and give positions in a **separate** table in the same .txt file.

**Note:** the table should also be generated by program (not copy pasted)

Eg.

Separate table

Separate table

Codon/Amino acid	position
ATG	0-2, 105-107
M	0, 35