

Substitution cipher instructions

1. First, import the modules `translit` and `string`.

```
1 import translit, string
```

2. **Define string of encrypted text.** Note there are issues if you try to copy-paste the text below into Python, the same string is available from the website, where copy-pasting should not be a problem.

```
1 s = "UQ UZ D EMDRQUNRX GDB. TPB DYM TM ZUQQUIO UI D GDYW YJJH  
    YMDGUIO QPUZ HMZZDOM? EMFDRZM UQ UZ NRI, ERQ UN TM GJ IJQ OMQ  
    HJYM M'Z TM TUXX ZQDB PMYM NJYMSMY"
```

3. **Define the encryption alphabet.** Convention: the encrypted message is in capital letters.

```
1 enc="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
```

4. **Define a decryption alphabet.** For now, it is the same as the encryption alphabet.

```
1 dec="ABCDEFGHIJKLMNOPQRSTUVWXYZ"
```

5. **Perform a frequency analysis.** Count the number of occurrence of single letters and group of two/three letters.

```
1 translit.count_letters(s)  
  translit.count_word2(s)  
3 translit.count_word3(s)
```

Use the frequency table (below) to make an educated guess for one letter of the decryption alphabet.

Letter	a	b	c	d	e	f	g	h	i	j	k	l	m
Percentage	8.2	1.5	2.8	4.3	12.7	2.2	2	6.1	7	0.2	0.7	4	2.4
Letter	n	o	p	q	r	s	t	u	v	w	x	y	z
Percentage	6.7	7.5	1.9	0.1	6	6.3	9.1	2.8	1	2.4	0.2	2	0.1

Say our guess is that “M” in the cypher is “e” in the clear text (because “M” is the most common letter in the cypher, may be it also corresponds to the most common letter in English?)

```
1 # modify the line where dec is assigned to  
dec="ABCDEFGHIKLeNOPQRSTUVWXYZ"
```

6. Decrypt the message with the new alphabet.

```
m=translit.translit(s,dec)  
2 print "Decrypted message?\n" + m
```

The output you should be getting looks like:

```
Decrypted message?  
2 UQ UZ D EeDRQUNRX GDB. TPB DY e Te ZUQQUIO UI D GDYW YJJH YeDGUIO  
  QPUZ HeZZDoe? EeFDRZe UQ UZ NRI, ERQ UN Te GJ IJQ OeQ HJYe e'Z  
  Te TUXX ZQDB PeYe NJYeSeY
```

7. Repeat for another letter.