Problem Set #1

1. Given the following variables, what will print when the single lines of code below are executed? If a line crashes, provide a description of the error.

```
a_list = [10, 2, 5, 2, [0,1], [ [30,40], 30] ]
a_string = "More practice problems!"
a_dictionary = {"cat":4, "dog":[2,5], "foobar":"bar", "!":-1}


print ( a_list[2] )

print ( a_list.index(2) )

print ( a_list[a_list[2]] )

print ( a_string[9:13] )

print ( a_string.split("a") )

print ( a_dictionary["!"] )

print ( "bar" in a_dictionary )

print ( a_list[a_dictionary["cat"]] )

print ( a_dictionary.keys() )

print ( a_list[ a_dictionary[ a_string[-1] ] ][0][1] )
```

2. Read the code in the 1st column. Write the output of the code in the 2nd column, and answer the question in the 3rd column. **No output** and **error** are both possible.

Code	Output	Question

`def cheer(a, b, c, d):`
`  s = "3"`
`  for i in [a,b,c,d]:`
`  s *= i`
`  return s`
 	
`a = 1`
`b = 2`
`c = 3`
`d = 4`
 	
`e = cheer(d, c, b, a)`
	What is the output of this program?	What value is in **e**?

`a = {'sound': 'beep', 1:200}`
`def make_some_noise():`
`  noise = (a['sound'] + '!')`
`  print(noise)`
 	
`b = make_some_noise()`
	What is the output of this program?	What value is in **b**?

`def make_some_noise(noise):`
`  return noise + '!' * 4`
 	
`result = make_some_noise('pop')`
`print(noise)`
	What is the output of this program?	What value is in **result**?

3. Edit the following program. If the user enters 'first', we want to print 'Alice'. Use **exception handling** to prevent an error from happening when the user types in 'foo'. Instead of an error, the program should write out, "I don't have that data!".

```
person = {'first':'Alice', 'last':'Abel', 'middle':'A'}
response = input('What data would you like?')
```

4. Use the following dictionary to answer the questions below.

```
d = {'x':100, 'y':200}
```

a) Write two ways of **retrieving** the value at key, **'x'** from the dictionary, **d**.

b) Write the correct way of **adding** a new key value pair, **'z':300**, to the dictionary, **d**.

c) Write the correct way of **removing** the element, **'x'**, from the dictionary **d**.

d) Write 3 ways to iterate over the dictionary, **d.**

5. Trace the output of the following program. **The order of the key value pairs will not be taken into account**.

```
d1 = {'first_name':'Stanley', 'year_born':1928}
d2 = {'first_name':'Ridley', 'year_born':1937}
d2['last_name'] = 'Scott'
s = ''
for k, v in d2.items():
    s += str(v)


print(d1)
print(d2)
print(s)
```

6. Trace the output of the following program.

```
code = { 6:["t", "g"], 14:["p","q"], 15:["c", "d"], 9:["o", "a"] }
data = "5,0:3,1:2,0"

splitdata = data.split(":")
for item in splitdata:
    splititem = item.split(",")
    print ( code[ int(splititem[0])*3 ][ int(splititem[1]) ], end="")
```

7. You've been hired to write an English to Spanish translator by someone that loves cats and dogs. Write a **function** called **en_espanol** that translates English words into Spanish words. You'll **write this two ways**, but both will do the following:

- Your function should take one argument – the word to be translated. It should return the translated word.
- If the word is 'cat' in any casing ('Cat', 'CAT', 'cAt', etc.) return 'gato', and if the word is 'dog' in any casing return 'perro'.

- For all other words, return 'no se'.

Example usage: Resulting Output:

```
print(en_espanol('Cat'))              gato
print(en_espanol('Final Exam'))       no se
```

Part 1: Implement **using conditionals**.

Part 2: Implement **using a dictionary to "look up" translations** instead of conditionals.