Dictionaries

Dictionaries are *associative* structures -- they associate *keys* with *values*. You look up something in a dictionary by giving it the key; the dictionary gives you the value stored for that key. The values can be any data you can represent in Python - numbers, strings, lists, and so forth. The keys must be members of an *immutable* type -- such as numbers, strings, or tuples. You can't use lists for keys because lists are mutable. Dictionaries themselves are another mutable type, so we can write functions that change the data stored in a dictionary.

There aren't a lot of operations for dictionaries:

D = {} creates an empty dictionary.
D[key] looks up the value associated with the key.
D[key] = value creates an association between the key and the value.
D.keys() is the list of keys currently set for the dictionary.

Note that your program will crash if you try to look up something that isn't a key, so we usually do lookups in 2 steps. For example, to print the value associated with key x:

```
if x in D.keys():
        print(x, D[x])
else:
        print( "no value found." )
```

You can't use a for-loop to run directly through the values in a dictionary; instead, work through the keys, as in

```
for x in D.keys():
        print( x, D[x] )
```

What will this print?

```
def change(D):
        D["bob"] = 63

def main():
        Ages = {}
        change(Ages)
        for person in Ages.keys():
                print( "%s: %d" ) %(person, Ages[person])
main()
```

A: nothing: D and Ages are different variables
B: It will generate an error and not run
C: It will run and crash because D isn't defined
D: It will print

```
bob 63
```

What will this print?

```
def change(D):
        D = { }
        D["bob"] = 63

def main():
        Ages = {}
        change(Ages)
        for person in Ages.keys():
                print( "%s: %d" ) %(person, Ages[person])
main()
```

A: nothing: D and Ages are different variables

B: It will generate an error and not run

C: It will run and crash because D isn't defined

D: It will print

bob 63