

[Courseware \(/courses/MITx/6.00.1_4x/3T2014/courseware\)](/courses/MITx/6.00.1_4x/3T2014/courseware)[Updates & News \(/courses/MITx/6.00.1_4x/3T2014/info\)](/courses/MITx/6.00.1_4x/3T2014/info)[Calendar \(/courses/MITx/6.00.1_4x/3T2014/89309559b0414f6d8cbef9e48ca19f4b/\)](/courses/MITx/6.00.1_4x/3T2014/89309559b0414f6d8cbef9e48ca19f4b/)[Wiki \(/courses/MITx/6.00.1_4x/3T2014/course_wiki\)](/courses/MITx/6.00.1_4x/3T2014/course_wiki)[Discussion \(/courses/MITx/6.00.1_4x/3T2014/discussion/forum\)](/courses/MITx/6.00.1_4x/3T2014/discussion/forum)[Progress \(/courses/MITx/6.00.1_4x/3T2014/progress\)](/courses/MITx/6.00.1_4x/3T2014/progress)

L12 PROBLEM 3 (5/5 points)

In this problem, you'll be asked to read through an object-oriented implementation of the hand from the word game problem of Problem Set 4. You'll then be asked to implement one of its methods. Note that the implementation of the object-oriented version of the hand is a bit different than how we did things with the functional implementation; pay close attention to doc strings and read through the implementation carefully.

To begin: Download [L12_hand.py \(/c4x/MITx/6.00.1_4x/asset/files_finger_exercises_L12_hand.py\)](/c4x/MITx/6.00.1_4x/asset/files_finger_exercises_L12_hand.py) and read through the file. Be sure to understand what's going on in the file. Make a few instances of the `Hand` class, and play around with the existing methods.

When you have completed reading through the file, implement the `update` method.

Paste the entire `Hand` class in the box below.

The `__str__` method is this:

```
def __str__(self):
    """
    Display a string representation of the hand.
    """
    output = ''
    hand_keys = self.hand.keys()
    hand_keys.sort()
    for letter in hand_keys:
        for j in range(self.hand[letter]):
            output += letter
    return output
```

A more concise version of this code might be

```
def __str__(self):
    """
    Display a string representation of the hand.
    """
    output = ''
    for letter in sorted(self.hand.keys()):
        output += letter * self.hand[letter]
    return output
```

Use whichever `__str__` method you like. This will make sure the grading of the hand's display is consistent.

Correct

```
# This is the solution for the update method only.

def update(self, word):
    """
    Does not assume that self.hand has all the letters in word.

    Updates the hand: if self.hand does have all the letters to make
    the word, modifies self.hand by using up the letters in the given word.

    Returns True if the word was able to be made with the letter in
    the hand; False otherwise.

    word: string
    returns: Boolean (if the word was or was not made)
    """
    # Make a copy of the hand, and try to update it
    new_hand = self.hand.copy()
    for letter in word:
        try:
            new_hand[letter] -= 1
        except KeyError:
            # if 'letter' isn't in the hand, we can't make the word from this hand.
            return False
    for letter in new_hand.keys():
        # If any of the letter counts of the new hand are less than zero after the
        # update, then we can't make the word from this hand.
        if new_hand[letter] < 0:
            return False
    # If we've gotten to here, we must be able to make the word from this hand.
    # Set self.hand to the new, updated hand and return True.
    self.hand = new_hand
    return True
```

Test results

CORRECT

[See full output](#)[See full output](#)

Check

Hide Answer

Show Discussion

 [New Post](#)

About (<https://www.edx.org/about-us>) Jobs (<https://www.edx.org/jobs>)
Press (<https://www.edx.org/press>) FAQ (<https://www.edx.org/student-faq>)
Contact (<https://www.edx.org/contact>)



EdX is a non-profit created by founding partners Harvard and MIT whose mission is to bring the best of higher education to students of all ages anywhere in the world, wherever there is Internet access. EdX's free online MOOCs are interactive and subjects include computer science, public health, and artificial intelligence.



(<http://www.meetup.com/YourMeetup>)



(<http://www.facebook.com/EdxOnline>)



(<https://twitter.com/YourPlatformTwitterAcco>)



(<https://plus.google.com/YourGooglePlusAcco>)



(<http://youtube.com/user/edxonline>)

© 2014 edX, some rights reserved.

Terms of Service and Honor Code -
Privacy Policy (<https://www.edx.org/edx-privacy-policy>)