Question 1

>>> t = 'Hello all'

... What does this expression evaluate to?

A: 'lo a'
B: 'lo'
C: 'lo '
D: 'o '
E: I do not know
Question 1, solved

A: 'lo a'
B: 'lo'
C: 'lo ' CORRECT
D: 'o '
E: I do not know

What does this expression evaluate to?

```python
t = 'Hello all'
t[3:6]  
>>> 'lo '
```
Question 2

What does this expression evaluate to?

A: 'all'
B: 'l'
C: 'Hel'
D: Error!
E: I do not know
What does this expression evaluate to?

A: 'all'
B: 'l'
C: 'Hel' CORRECT
D: Error!
E: I do not know
Want to write function `middle`, which returns the middle 3\textsuperscript{rd} of a string (length divisible by 3).

How we want it to behave:

```python
>>> middle('abc')
'b'
>>> middle('aabbcc')
'bb'
>>> middle('aaabbbccc')
'bbb'
```

Important Questions:

1. What are the parameters?
2. What is the return value?
3. What goes in the body?

```python
def middle(text):
    ???
    return middle_third
```
def middle(text):
    """Returns: middle 3\textsuperscript{rd} of text
Param text: a string with length divisible by 3""

    # Get length of text
    size = len(text)

    # Start of middle third
    start2 = size//3

    # Start of last third
    start3 = (2*size)//3

    # Get the substring
    middle_third = text[start2:start3]

    return middle_third
def firstparens(text):
    """Returns: substring in ()
    Uses the first set of parens
    Param text: a string with ()"""

>>> s = 'One (Two) Three'
>>> firstparens(s)
'Two'

>>> t = '(A) B (C) D'
>>> firstparens(t)
'A'
def firstparens(text):
    """Returns: substring in ()
    Uses the first set of parens
    Param text: a string with ()"""

    # Find the open parenthesis
    start = text.index('(')

    # Find the close parenthesis
    end = text.index(')')

    inside = text[start+1:end]

    return inside
String Extraction, a better version

```python
def firstparens(text):
    """Returns: substring in ()
    Uses the first set of parens
    Param text: a string with ()"""

    # Find the open parenthesis
    start = text.index('(')

    # Store part AFTER paren
    substr = text[start+1:]

    # Find the close parenthesis
    end = substr.index(')')

    inside = substr[:end]
    return inside

>>> s = 'One (Two) Three'
>>> firstparens(s)
'Two'

>>> t = '(A) B (C) D'
>>> firstparens(t)
'A'

>>> v = 'A) B (C) D'
>>> firstparens(v)
```