What code will define a function called “multTwo" to accept two numbers, compute their product, and return it to the calling program?

A:

```
def multTwo (a, b):
    return a * b
```

What code will define a function called “multTwo" to accept two numbers, compute their product, and display it?

A:

```
def multTwo (a, b):
    print ( a * b )
```

The code below defines a function to do data validation. What will be the value of the variable 'name' after the code is executed?

```
def validState (a):
    states = (('AK', 'Alaska'), ('CA', 'California'),
              ('MD', 'Montana'), ('VA', 'Vermont'))
    for s in states:
        if a == s[0]:
            return  s[1]
    print ('Invalid abbreviation')

name = validState ('VA')
```

A:

Vermont

What will be the display output of the following code?

```
def addNums (w, x, y = 10, z = 20):
    sum = w + x + y + z
    if sum > 100:
        return 'huge'
    if sum%2 == 0:
        return 'even'
    return 'odd'
    print ('End of function')

print (addNums(30, 50))
```

A. huge

What will be the display output of the following code?

```
def addNums (w, x, y = 10, z = 20):
    sum = w + x + y + z
    if sum > 100:
        return 'huge'
    if sum%2 == 0:
        return 'even'
    return 'odd'
    print ('End of function')

print (addNums(3, 5, 7))
```

A. odd

What will be the display output of the following code?

```
def addNums (w, x, y = 10, z = 20):
    sum = w + x + y + z
    if sum > 100:
        return 'huge'
    if sum%2 == 0:
        return 'even'
    return 'odd'
    print ('End of function')

print (addNums(1, 3, z = 90))
```

A. huge

What will be the display output of the following code?

```
def addNums (w, x, y = 10, z = 20):
    sum = w + x + y + z
    if sum > 100:
        return 'huge'
    if sum%2 == 0:
        return 'even'
    return 'odd'
    print ('End of function')

print (addNums(1, 3, 5, 6))
```

A. odd

```
def addNums (inNums):
    sum = 0
    for n in inNums:
        sum += n
    return sum

numList = [1, 3, 5, 7, 9]
print (addNums(numList))
```

A. 25

```
def printReceipt (items):
    for n in range (len(items)):
        print(n + 1, items[n])
    return True

items = ['milk', 'eggs', 'bread', 'ale']
printReceipt(items)
```

A:

1 milk

2 eggs

3 bread

4 ale

```
indata = fi.read()
```

A: indata will be a string containing the entire file

```
indata = fi.readlines()
```

A: indata will be a list consisting of one string for each line in the file

A text file has the following data:

Washington
Adams
Jefferson

After the above file is opened as "fi", what will be the display output of the following code?

```
x = fi.readlines()
for line in x:
    print(line)
```

A:
Washington
Adams
Jefferson

A text file 'states.txt' has the following data:

AL,Alabama,Montgomery

What code that uses the **split** method will open and read the file, create the list below, and assign it the variable name "L":

['AL', 'Alabama', 'Montgomery'] ?

The code must work for all possible U.S. states, not just the 'AL' example above.

A:
```
f = open('states.txt', 'r')
x = f.read()
L = x.split(',')
L[2] = L[2].strip( )
```

A list of continents:

continents = ['Africa', 'Antarctica', 'Asia', 'Australia', 'Europe', 'North America']

What code will write the names of the continents to a file called 'continents.txt' so that they will appear as lines 1 – 6 in the file when a text editor like Notepad is used to display the file?

A:
```
f = open('continents.txt', 'w')
for c in continents:
    f.write(c + '\n')
f.close()
```

You're writing code to maintain a dictionary of US presidents and the year they were elected. The key is the year, the name is the value. The dictionary looks like this:

PD = {1788:'Washington', … 2020:'Biden'}, but doesn't have all presidents. Prompt the user for a year. If it's in PD, print a message stating 'already there'. If not there, prompt again for the president's name and add it to the dictionary. Detect when a non-integer year is entered using an exception handler and re-prompt until a valid year is entered.

```
while True:
    try:
        year = int(input('Enter year: '))
        if year in PD:
            print('already there')
        else:
            name = input('Enter presidents name: ')
            PD [year] = name
            break
    except:
        print('year must be an integer – please re-enter')
```

'hat code would convert the above list (assume more than the ur shown) into a dictionary called 'prez' using the year elected as ey and the name of the president as the value?

```
prez = { }
for p in presidents:
    prez[p[1]] = p[0]
```

A list of grocery items purchased is in this list:

items = [['eggs', 2.50], ['Vie de France baguette', 1.95], ['Pinot Noir', 11.0]]

What code is needed to print the receipt as in the previous question, but add a total cost at the bottom labelled "Total" and line it up with the printed items?

A:
```
def receipt (inlist):
    total = 0.0
    for item in inlist:
        print (' {0:20s}  ${1:8.2f}'.format(item[0], item[1]))
        total += item[1]
    print (' {0:20s}  $(1:8.2f}'.format("Total", total))
```