

1. I want to create a 2-dimension list, representing a matrix

```
1   -2   3   9   10
3   12  -9  12   9
-11 65   4   8   5
```

Each row will be represented by a list, and the matrix will be a list of lists.

Use some of the blocks below to define such a list Matrix (feel free to use Python interpreter for help).

```
Matrix = []
```

```
a = [1, -2, 3, 9, 10]
```

```
A = [[1], [-2], [3], [9], [10]]
```

```
Matrix.append(a)
```

```
Matrix = [ [1, -2, 3, 9, 10] ]
```

```
Matrix.append(A)
```

```
B = [[3], [12], [-9], [12], [9]]
```

```
b = [3, 12, -9, 12, 9]
```

```
Matrix.append(B)
```

```
Matrix.append(b)
```

```
C = [-11, 65, 4, 8, 5]
```

```
c = [[-11], [65], [4], [8], [5]]
```

```
Matrix.append(c)
```

```
Matrix.append(C)
```

Possible answers:

(a)

```
Matrix = []
Matrix.append(a)
Matrix.append(b)
Matrix.append(C)
```

(b)

```
Matrix = [ [1, -2, 3, 9, 10] ]
Matrix.append(b)
Matrix.append(C)
```

2. Assume we have the following list representing a matrix:

```
A = [ [1,2,3,4],
      [0,1,2,3],
      [8,7,6,5],
      [3,4,2,1],
      [1,2,9,3] ]
```

1) write the statement to display the third row

```
print(A[2])
```

2) write the code to print the elements of the 3rd column using a loop

for row in A:

```
    print(row[2])
```

3. Consider the following code

```
from random import randint

L = []

for i in range(10):
    a = randint(-100,0)
    b = randint(0,100)
    L.append((a,b))
```

It creates a list of 10 elements. Each element is a tuple, for example
`L = [(-6,8), (-2,0), (-1,9), (-5,7), (0,2), (-5,8), (0, 8), ...]`

Write the code to print out the list of squares of the second elements in the tuples if the first element in the tuple is not 0, separated by a comma with a space.

Here is what we should get for the list `L` above:
`64, 0, 81, 49, 2, 64, 8, ...`

A possible answer:

```
for t in L:
    if t[0] != 0:
        print(t[1]**2,end=" ")
    else:
        print(t[1], end=" ")
```