```
"""
A really simple domino game.
"""

import boneyard as yard
# boneyard must have these functions:
#     1. tiles_remaining
#     2. draw()

import domino as doms
# domino must have these functions:
#     1. as_str()
#     2. get_right()
#     3. get_left()

the_yard = yard.create()
game_over = False

while not game_over:
    if yard.tiles_remaining(the_yard) == 0:
        print('Ran out of dominoes')
        game_over = True
    else:
        input('Press return to continue')
        tile = yard.draw(the_yard)
        print('Got tile %s' % (doms.as_str(tile)))
        if doms.get_left(tile) == 6 or doms.get_right(tile
) == 6:
            print('Got a SIX!!!')
            game_over = True

print("Game Over.")

```

```
 1 # 1. create()
 2 # 2. as_str()
 3 # 3. get_right()
 4 # 4. get_left()
 5
 6
 7 def create(left, right):
 8     """
 9      creates a domino and returns in tuple.
10     """
11     return (left, right)
12
13 def as_str(domino):
14     """
15      returns the domino values as a string
16     """
17     return "[%d | %d]" % (get_left(domino), get_right(
   domino))
18
19 def get_left(domino):
20     """
21      returns the left value of the domino
22     """
23     return domino[0]
24
25 def get_right(domino):
26     """
27      returns the right value of the domino
28     """
29     return domino[1]
30
31
32 # James Lin
33 # I affirm that I have carried out the attached academic
   endeavors with full academic honesty,
34 # in accordance with the Union College Honor Code and the
   course syllabus.
35
```

```
1  """
2  Models a boneyard -- a pile of dominoes.
3  """
4
5  import domino as d
6  # 1.  create()
7
8  import random
9
10 def create():
11     """
12     returns a pile of dominoes containing
13     one copy of every possible domino
14     """
15     yard = []
16     for i in range(0,7):
17         for j in range(0, 7):
18             tile = d.create(i, j)
19             yard.append(tile)
20     return yard
21
22 def draw(boneyard):
23     """
24     removes a random domino from the boneyard
25     and returns it
26     """
27     n = random.randint(0, len(boneyard)-1)
28     return boneyard.pop(n)
29
30 def tiles_remaining(boneyard):
31     """returns the number of tiles left in the yard"""
32     return len(boneyard)
33
```