

I wrote 4 programs, including a randomized guesser with auto input, a guessing game with manual input, a list-making application for storing phone numbers.

The image shows a screenshot of a Python 2.7.10 Shell window and a Spyder IDE window. The Shell window displays the output of a program, including a welcome message, a fortune teller's response, and a list-making application's output. The Spyder IDE window shows the source code for the cat names application, which includes a random fortune teller and a list-making application.

```

Python 2.7.10 Shell*
File Edit Shell Debug Options Window Help
505-555-7777, t3
Register your cat phone number, Guest #4 of the Day. Please use comma between num and name, like: xxx-xxx-xxx, Tom
xxx, Tom

Traceback (most recent call last):
  File "C:\Users\thoma_000\Google Drive\_CCCreative\_school\1096_IoT\IoTpythonProgs\catNames.py", line 25, in <module>
    name = input()
  File "<string>", line 0
    ^
SyntaxError: unexpected EOF while parsing
>>> ===== RESTART =====
>>>
Welcome to this Good Fortune Page, and your fate reads that :

(2) Your day will be fast! Slow, and be sure to not make waste in the haste

Register all your cat phone numbers, Entry #1. After entering all numbers, hit ENTER to stop. Use comma between number and name, like: xxx-xxx-xxxx, Tom
'44,1'
The cat names are:

Ln: 151 Col

*catNames.py - C:\Users\thoma_000\Google Drive\_CCCreative\_school\1096_IoT\IoTpythonProgs\catNames.py ...
Spyder Editor
File Edit Format Run Options Window Help

#Author: Thomas Maestas
#CIS 1096-IoT
"""
# random
import random
def getAnswer(answerNumber):
    if answerNumber == 1:
        return '(1) Your day will be mundane, so look for novelty in little things'
    elif answerNumber == 2:
        return '(2) Your day will be fast! Slow, and be sure to not make waste in the haste'
    elif answerNumber == 3:
        return '(3) Your day portends great things: capture your fate!'
print('Welcome to this Good Fortune Page, and your fate reads that : ')
print
print(getAnswer(random.randint(1,4)))
print
# List-making app
catNames = []
name = ''
while True:
    print('Register all your cat phone numbers, Entry #' + str(len(catNames) + 1) + '. After entering all nu
    name = input()
    if name == '':
        break
    catNames = catNames + [name] # list concat
print('The cat names are:')
for name in catNames:
    print(' ' + name)

Automate the Boring Stuff with Python
to a return value. print() re
Ln: 6 Col

```

pe the following source code into the file editor, and save it as *essTheNumber.py*:

```
this is a guess the number game.
import random
secretNumber = random.randint(1, 20)
print('I am thinking of a number between 1 and 20.')

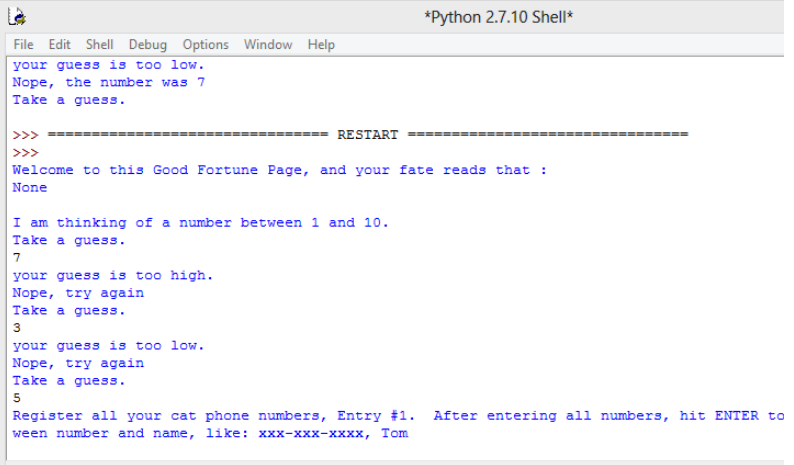
Ask the player to guess 6 times.
for guessesTaken in range(1, 7):
    print('Take a guess.')
    guess = int(input())

    if guess < secretNumber:
        print('Your guess is too low.')
    elif guess > secretNumber:
        print('Your guess is too high.')
    else:
        break # This condition is the correct guess!

guess == secretNumber:
    print('Good job! You guessed my number in ' + str(guessesTaken) + ' guesses!')
else:
    print('Nope. The number I was thinking of was ' + str(secretNumber))
```

Let's look at this code line by line, starting at the top.

```
this is a guess the number game.
import random
secretNumber = random.randint(1, 20)
```



```
*Python 2.7.10 Shell*
File Edit Shell Debug Options Window Help
your guess is too low.
Nope, the number was 7
Take a guess.

>>> ===== RESTART =====
>>>
Welcome to this Good Fortune Page, and your fate reads that :
None

I am thinking of a number between 1 and 10.
Take a guess.
7
your guess is too high.
Nope, try again
Take a guess.
3
your guess is too low.
Nope, try again
Take a guess.
5
Register all your cat phone numbers, Entry #1. After entering all numbers, hit ENTER to
between number and name, like: xxx-xxx-xxxx, Tom

C:\Users\thoma_000\Google Drive\_CCCreative\_school\1096_IoT\IoTpytho
File Edit Format Run Options Window Help
print
# 2 # Guessing Game
import random
secretNumber = random.randint(1, 11)
print('I am thinking of a number between 1 and 10.')
#
for guessesTaken in range(1,5):
    print('Take a guess.')
    guess = int(input())
    if guess < secretNumber:
        print('your guess is too low.')
    elif guess > secretNumber:
        print('your guess is too high.')
    else:
        break # condition correct!
if guess == secretNumber:
    print('Good job! You guessed my number in ' + str(guessesTaken) + ' guesses!')
else:
    print('Nope, the number was ' + str(secretNumber))

= int(input())
```

Now the `spam` and `cheese` variables refer to separate lists. When `cheese` is modified when you assign 42 at index 1. As the reference ID numbers are no longer the same for both variables, they refer to independent lists.

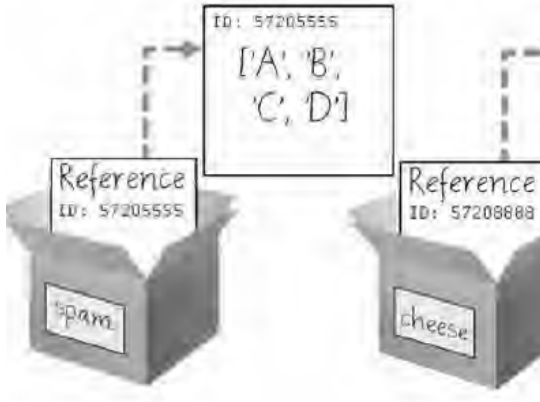


Figure 4-7. `cheese = copy.copy(spam)` creates a second list independently of the first.

If the list you need to copy contains lists, then use the `deepcopy()` function instead of `copy.copy()`. The `deepcopy()` function will copy

```

Python 2.7.10 Shell*
File Edit Shell Debug Options Window Help
>>> ===== RESTART =====
>>>
[1, 2]
[1, 2, 99, 98]
4
<type 'tuple'>
(1, 2, 99, 98, 'henry')

[1, 2, 3, 'Hi, Im appended']

['A', 'B', 'C', 'D']
['A', 123, 'C', 'D']

Welcome to this Good Fortune Page, and your fate reads that :

(3) Your day portends great things: capture your fate!

[1 2 3 ] + [t, o, m]
[1, 2, 3, 't', 'o', 'm']

I am thinking of a number between 1 and 10.
Take a guess.

```

```

catNames123guess.py - C:/Users/thoma_000/Google Drive/_CCCreative_
File Edit Format Run Options Window Help
#CIS 1096-IoT
"""
# 0 # eggs append !!
eggs = [1, 2, 3]
del eggs[2]
print eggs
eggs.append(99)
eggs.append(98)
print eggs
print len(eggs)
eggs.append('henry')
eggs = tuple(eggs) # Tuple Conversion
print type(eggs)
print eggs
print
def legos(someParameter):
    someParameter.append('Hi, Im appended')
spam = [1,2,3]
legos(spam)
print (spam)
print
import copy
spama = ['A', 'B', 'C', 'D']
cheese = copy.copy(spama)
cheese[1] = 123
print spama
print cheese
print

# 1 # randomized fortune:
import random

```

177 Automate the Boring Stuff with Python

SUMMARY