Exercise 35: Branches and Functions

MAIN

PLAY VIDEO

You have learned `if-statements`, functions, and lists. Now it's time to bend your mind. Type this in, and see if you can figure out what it's doing.

PREVIOUS

NEXT

HELP

Follow

```python
1    from sys import exit
2
3    def gold_room():
4        print "This room is full of gold.  How much do you take?"
5
6        choice = raw_input("> ")
7        if "0" in choice or "1" in choice:
8            how_much = int(choice)
9        else:
10           dead("Man, learn to type a number.")
11
12       if how_much < 50:
13           print "Nice, you're not greedy, you win!"
14           exit(0)
15       else:
16           dead("You greedy bastard!")
17
18
19   def bear_room():
20       print "There is a bear here."
21       print "The bear has a bunch of honey."
22       print "The fat bear is in front of another door."
23       print "How are you going to move the bear?"
24       bear_moved = False
25
26       while True:
27           choice = raw_input("> ")
28
29           if choice == "take honey":
30               dead("The bear looks at you then slaps your face off.")
31           elif choice == "taunt bear" and not bear_moved:
32               print "The bear has moved from the door. You can go through it now."
33               bear_moved = True
34           elif choice == "taunt bear" and bear_moved:
35               dead("The bear gets pissed off and chews your leg off.")
36           elif choice == "open door" and bear_moved:
37               gold_room()
38           else:
39               print "I got no idea what that means."
40
41
42   def cthulhu_room():
43       print "Here you see the great evil Cthulhu."
44       print "He, it, whatever stares at you and you go insane."
45       print "Do you flee for your life or eat your head?"
46
47       choice = raw_input("> ")
48
49       if "flee" in choice:
50           start()
51       elif "head" in choice:
52           dead("Well that was tasty!")
53       else:
54           cthulhu_room()
55
56
57   def dead(why):
```

```
58        print why, "Good job!"
59        exit(0)
60
61   def start():
62        print "You are in a dark room."
63        print "There is a door to your right and left."
64        print "Which one do you take?"
65
66        choice = raw_input("> ")
67
68        if choice == "left":
69            bear_room()
70        elif choice == "right":
71            cthulhu_room()
72        else:
73            dead("You stumble around the room until you starve.")
74
75
76   start()
```

What You Should See

Here's me playing the game:

```
$ python ex35.py
You are in a dark room.
There is a door to your right and left.
Which one do you take?
>  left
There is a bear here.
The bear has a bunch of honey.
The fat bear is in front of another door.
How are you going to move the bear?
>  taunt bear
The bear has moved from the door. You can go through it now.
>  open door
This room is full of gold.  How much do you take?
>  1000
You greedy bastard! Good job!
```

Study Drills

1. Draw a map of the game and how you flow through it.
2. Fix all of your mistakes, including spelling mistakes.
3. Write comments for the functions you do not understand.
4. Add more to the game. What can you do to both simplify and expand it?
5. The `gold_room` has a weird way of getting you to type a number. What are all the bugs in this way of doing it? Can you make it better than what I've written? Look at how `int()` works for clues.

Common Student Questions

Q: Help! How does this program work!?

When you get stuck understanding a piece of code, simply write an English comment above *every* line explaining what that line does. Keep your comments short and similar to the code. Then either diagram how the code works or write a paragraph describing it. If you do that you'll get it.

Q: Why did you write `while True`?

That makes an infinite loop.

Q: What does `exit(0)` do?

On many operating systems a program can abort with `exit(0)`, and the number passed in will indicate an error or not. If you do `exit(1)` then it will be an error, but `exit(0)` will be a good exit. The reason it's backward from normal boolean logic (with `0==False` is that you can use different numbers to indicate different error results. You can do `exit(100)` for a different error result than `exit(2)` or `exit(1)`.

Q: Why is `raw_input()` sometimes written as `raw_input('> ')`?

The parameter to `raw_input` is a string that it should print as a prompt before getting the user's input.

Video

Purchase The Videos For $29.59

For just $29.59 you can get access to all the videos for Learn Python The Hard Way, **plus** a PDF of the book and no more popups all in this one location. For $29.59 you get:

- All 52 videos, 1 per exercise, almost 2G of video.
- A PDF of the book.
- Email help from the author.
- See a list of everything you get before you buy.

When you buy the videos they will immediately show up **right here** without any hassles.

Already Paid? Reactivate Your Purchase Right Now!

Buying Is Easy

Buying is easy. Just fill out the form below and we'll get started.

Full Name

Full Name

Email Address

Email Address

◉ VISA MasterCard AMERICAN EXPRESS   ○ PayPal Use
Pay With Credit Card   your PayPal™
(by Stripe™)   account.

Buy Learn Python The Hard
Way, 3rd Edition

Zed Shaw	**Amazon**	**Amazon**	**B&N**	**B&N**
PDF + Videos + Updates	Paper + DVD	Kindle	Paper + DVD	Nook (No Video)
$29.95	**$22.74**	**$17.27**	**$22.96**	**$17.27**
	InformIT	**Interested In Ruby?**		
	eBook + Paper	Ruby is also a great language.		
	$43.19	**Learn Ruby The Hard Way**		