Week7 lecture5.3

So now we're going to talk about what we're trying to accomplish with

a loop, right?

We're trying to, we know how they mechanically work but

what are we looking for?

What if we're looking for the largest value or

checking to see if 42 is a member of a set or something?

Or looking for the largest letter, like the max function?

And so we're going to construct loops sort of with an idea of doing something to each

value in the set that we're iterating through.

And then coming up with some kind of result.

And the pattern that we're going to do is we're going to write a for loop.

And actually, in this next two segments, I'm going to do the exact same for loop.

But we're going to do something before the loop starts,

set some variables to initial values.

And then we're going to do something to every one of the values in our list.

And then, we don't know what the largest value is while the loop is running.

And our goal is, when the loop finally finishes, that we have something.

Whether it's the maximum, the minimum, the average, the total,

how many things there are, how many things match.

And so, the iterations are getting us closer to knowing the answer.

But they don't instantly know the answer, so we we have to work towards the answer.

By setting something and then sort of checking it a bunch of times.

And then we have sort of have absolute, the truth comes out at the bottom.

And you'll see this in a second.

So here's the little loop that we're going to do.

And I'm going to do this over and over and over again, and these numbers,

you're going to get tired of these numbers.

For some variable, iteration variable in 9, 41, 12, 3, 74, 15.

We print out a before.

So we're going to do something before the loop.

We're going to do something during the loop and after the loop.

Right now, I'm just printing everything out.

So you kind of see how the loop kind of works, okay?

Okay, so here's a little problem.

I'm going to show you a number of numbers and maybe it'll be a million or

maybe it'll be 6, I don't know.

But you don't know how many I'm going to show you.

And I want you to tell me when it's all said and done what the largest number is.

And it's not going to be too hard.

But imagine that you had to do this for a million numbers as a human being.

That you were just sitting there and numbers going by and by and by and by.

How would your brain truly solve the problem?

So here, ready?

Here goes some numbers.

从 :2:40 开始播放视频并学习脚本2:40

So did you get it?

从 :2:42 开始播放视频并学习脚本2:42

Did you get the largest number?

从 :2:44 开始播放视频并学习脚本2:44

It's not so much interesting about what the largest number was.

It wasn't that many numbers.

What was your brain doing to sort of track that problem, to solve that problem?

What was your brain doing? And what if you really had to do that for a million numbers?

And they were not little tiny numbers either,

they were sort of medium-size numbers.

And you had to do a million of them, what would be your technique?

Other than running away, because humans aren't good at this.

But what would be the precise technique that you'd use?

从 :3:12 开始播放视频并学习脚本3:12

And so here are the numbers.

Now actually as a human, we love looking at these numbers.

Like oh, 74 . But then you ask, like how did your mind exactly find them, right?

Our minds go, like [SOUND] there's 74.

Our mind doesn't look at them the way a computer looks at them.

It just kind of zooms in on 74 and just kind of [SOUND].

But that's not how a computer looks them.

A computer has to look at them as 3, 41, 12, 9, 74, 15.

I conclude at the very end that 74 is the largest number, right?

从 :3:50 开始播放视频并学习脚本3:50

But a human's just [SOUND] 74.

So humans think about this differently.

And so we have to realize, the purpose of that last little exercise, was to think

when we construct loops how computers are going to attack this kind of a problem.

They attack it sequentially.

They don't attack it magically the way we humans do.

从 :4:13 开始播放视频并学习脚本4:13

And the way that you do it is you create in your head, and you probably did it,

some notion of what is the largest number I've seen so far.

Like a variable.

And we start and say oh, I haven't seen any numbers so far, so

we'll stick -1 in there.

And then you see 3 and you go like well, okay, that first number

is also the largest I've seen so far, so 3 is pretty good.

If we stop now, and I tell you that's the last number, the largest number is 3.

It's not the largest number until I tell you we're done.

从 :4:42 开始播放视频并学习脚本4:42

We see 41, it's like whoa, 3 is a bad number, 41 is even better.

It's way better, we'll keep 41 as our so far, so far, so good.

But we're not done yet, we've got to keep going.

12, whoa, the 12 is no good because 41 is larger, we'll keep 41.

9? 9's no good, we're in really great shape.

Is 41 the largest?

No, but it is largest so far, so we know that much.

Now we see 74, whoa, that's way better than 41, so keep track of that one.

And then we go like, 15, nah looks like 74 is pretty good.

And now we're all done, and we see that it's 74.

So what happens is at the end of the loop, the only thing we knew was

the largest we saw so far and when I tell you we're all done.

Then the largest we saw so far, like just poof, it is the largest, right?

because it's all that we're ever going to see.

So that's how you have to construct these loops in Python.

So here's a bit of code that does this logical bit here, okay?

And so I'm going to make a variable.

And remember that variables can have underscores in it, and

I'm going to call this variable the largest\_so\_far.

And I'm going to set it to -1 at the beginning.

And before the loop starts, remember I said there's stuff we do at

the beginning of the loop, and then stuff we do during the loop.

And then our payoff is after the loop, after the loop. Okay?

So before the loop we set the largest we've seen so far,

which we've seen nothing, so we say -1, then we print it out.

So before the loop largest is so far is -1.

Then we have an iteration variable,

the\_num is going to go through successively 9, 41, 12, 3, 74.

Run this block of code five times, six times, once for each of those numbers.

And we're going to ask, if the number we're looking at, in this first case 9,

is greater than the largest so far,

then the largest so far is the number we just looked at because it's

bigger than the one we saw before.

So largest\_so\_far is the largest up to this point.

从 :6:35 开始播放视频并学习脚本6:35

And then we kind of grab it basically.

So then we print out, for each iteration, the largest so far.

So the first time we get 9 and the largest we've seen so

far is 9 because that's the first one we saw and it's the largest we've seen.

Then we go up, we advance to 41, 41 is the\_num.

And then 9 is largest\_so\_far, 41 is greater than it, so

then we stick the 41 and stick in largest\_so\_far.

So the largest we've seen so far is 41.

The one we just saw was 41, go up again, now 12 is going to run.

And so we see 12 and the largest so far is still 41.

And so we don't do anything, go back up, run 3,

we get 3, largest so far is 41, nothing needs to change.

Then we run again, we get 74, 74 runs, and that is the new champion largest number.

And then we go back one more time, 15,

in this code we don't know how many we're going to do.

We could do we could do 5 or 6 or a million, but we don't care,

we just keep the best we're doing is largest so far.

So we get 15 and the largest so far is still 74.

And now the for loop says you're done and we come out.

Now, largest\_so\_far is actually

the largest in the sequence of numbers, which is 74.

So this is how, that's a pretty messy slide.

This is basically how Python can go through a list and looking at

the largest so far and then when we're done with the list, that is the largest.

So you get the idea.

We set something up before, we do something to each value, and then at

the end we kind of get the payoff of what we were looking for in the first place.

从 :8:16 开始播放视频并学习脚本8:16

So up next we're going to talk about more of these loop idioms and

how to find the smallest, and how to count things, and how to do averages and

sums and stuff like that.