



# SKILLPILLS

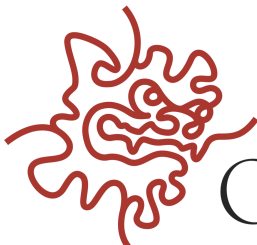
## Skill Pill: Basic Programming with Python

### Lecture 3: Loops, conditions, and functions, oh my!

**James Schloss**

Okinawa Institute of Science and Technology  
[james.schloss@oist.jp](mailto:james.schloss@oist.jp)

February 13, 2016



OIST

## 1 Conditions

- Assignments vs Comparison
- if, else, elif

## 2 Loops

- while
- for

## 3 Functions

- passing

Let's assign some values to some variables:

---

```
x = 5  
y = 17  
z = 5
```

---

In these cases, we are setting the values of *x*, *y*, and *z* to be the values 5, 17, and 5 respectively. Now, let's ask some questions. Type the following into your console:

---

<code>x == y;</code>	<code>x &gt;= y</code>
<code>x == z;</code>	<code>x &gt;= z</code>
<code>x &gt; y;</code>	<code>x &lt;= y</code>
<code>x &gt; z;</code>	<code>x &lt;= z</code>
<code>x != y;</code>	<code>x != z</code>

---

Now, let's see if we can do something with this information!

Define  $x$ ,  $y$ , and  $z$  as before:

---

```
x = 5
y = 17
z = 5
```

---

Now, let's write a code:

---

```
# our first if statement!
if x == z:
    print("x is equal to z")
elif x == y:
    print("x is equal to y")
else:
    print("Nothing is equal. What the heck, man?")
```

---

It might be best to do this in a file, rather than on the console.

## PROBLEM

Type the following into the your console:

```
a = 2**32
```

Now, using exclusively the comparisons we mentioned before and the if statement, find the number. **Do not do the following:**

```
print a
```

Aight. We can tell if things are true or false. Let's do something with that: **while loops**. Type the following into a file:

---

```
a = 0
while a < 5:
    print(a)
    a = a + 1
```

---

The while loop evaluates the condition you give it ( $a < 10$ ) and executes the commands underneath continuously until that statement is no longer true.

you should see:

---

```
0
1
2
3
4
```

Type the following into your console:

```
while True:  
    print("True")
```

# That was an awful idea!

Now press **CTRL + C** or click the **red square**.



We just created an **endless loop**.

What does this mean? Well, it means that our loop will go on forever and ever because we gave it a condition that was always true. Try the following:

---

```
a = 0
while True:
    print("True")
    a = a + 1
    if a >= 4:
        break
```

---

You should see:

---

```
True
True
True
True
```

---

## PROBLEM

Type the following into the your console:

```
a = 2**32
```

Now, using exclusively the comparisons we mentioned before, the if statement, **and a while loop**, find the number. **Do not do the following:**

```
print a
```

Note: Keep this code when you are done.

While loops are cool, but what if we don't want to manually assign a stop condition? for this, we use a **for** loop:

```
for i in range(0,5):  
    print(i)
```

You should see:

```
0  
1  
2  
3  
4
```

The for loop is powerful. It can create lists:

```
lst = [i for i in range(10)]  
print(lst)
```

... and it can iterate through them:

```
for i in lst:  
    print(i)
```

Now let's iterate through a list and keep  $i$ :

---

```
# Creates initial list
```

```
lst = ["apple", "banana", "cherry", "berry", "tomato"]
```

```
# loops through list and creates a number id for each element
```

```
for i, j in enumerate(lst):
```

```
    print(i,j)
```

```
    if j == "tomato":
```

```
        print("What the heck, man? This isn't a fruit!")
```

---

## PROBLEM

Aight. Look. You're a cop, and you know what that means: you love doughnuts. I mean, you really, really love doughnuts. Throughout your day, you stop by every doughnut shop in town and you pick up one doughnut more than you picked up in the previous shop. For example: at the first shop, you pick up one doughnut. At the second shop, you pick up two. At the third shop you pick up three, and so on.

Given the following list of doughnut shops:

---

```
shops = ["Krispy Kreme", "dunkin' donuts", "Nuts of Dough",  
         "Dough-natto", "Doug's Donuts", "Dough"]
```

---

How many donuts will you eat in a single day?

Let's write ourselves a code that finds the area in two different arbitrarily-sized rectangles...

```
# Defining the first square's properties
```

```
r1_l = 10; r1_h = 10  
r2_l = 20; r2_h = 300
```

```
# We also need some sort of counter for both r1 and r2
```

```
count_1 = 0; count_2 = 0
```

```
# Add a bunch of smaller blocks together for rectangle 1:
```

```
for i in range(r1_l):  
    for j in range(r1_h):  
        count_1 += 1
```

```
# Add a bunch of smaller blocks together for rectangle 2:
```

```
for i in range(r2_l):  
    for j in range(r2_h):  
        count_2 += 1
```

```
print(count_1, count_2)
```



You should see:

---

100 6000

---

But that was kinda cumbersome. We had to write the loops twice. Let's turn the code into a **function** and avoid that in the future...

---

```
def find_area(width, height):  
    # Initializes counter  
    count = 0  
  
    # Adds all of our smaller blocks together  
    for i in range(width):  
        for j in range(height):  
            count += 1  
  
    # Sends "count" outside of the function  
    return count
```

---

If we want to use these functions in main, we need to call them like so:

---

```
area = find_area(3,4)
```

---

```
def find_area(width, height):  
    # Initializes counter  
    count = 0  
  
    # Adds all of our smaller blocks together  
    for i in range(width):  
        for j in range(height):  
            count += 1  
  
    # Sends "count" outside of the function  
    return count  
  
print(find_area(10,10), find_area(20,300))
```

you should see:

100 6000

## PROBLEM

Remember that code I told you to keep back in slide 10? Let's make a function out of that. I want to be able to read in an arbitrary equation.

- Use Google
- Ask Questions
- Programmers are Grumpy. Be grumpy back.
- If you don't understand something, go back to basic principles!
- You have the tools. Make them shine!

**PROBLEM**

I have a ball 5 meters above the ground. If I drop it, how long will it take to hit the ground? Formula:

$$x = x_0 + v_0 t + \frac{1}{2} a t^2$$

Hint: Use a while loop

**PROBLEM**

Two lovebugs are 10 steps away from each other. If they race towards each other with all of their hearts (at one step per second), how long will it take the bugs to meet?

## PROBLEM

Make a list of colors:

```
colors = ["yellow", "green", "red", "purple"]
```

Move each color up one index. If you are at the first index, copy it to the last element. This will “rotate” the colors.

## PROBLEM

Fizzbuzz: in a range from 0 to 100, print out “fizz” whenever

```
i % 3 == 0
```

and “buzz” whenever

```
i % 5 == 0
```