Welcome back to the
Python Course

Conditional Operators

When you want to verify a statement, you can use *if*

For example:

```
if True : print("it's true")
```

Conditional Operators

|if answer == "yes" :

| | print ("you said yes")

| | answer = input ("why yes?")

|else :

| | print ("you said no")

| | answer = input ("why no?")

Nested Conditionals

If the statement can have several values, you can do this:

```
if x==0 :

        print("it's 0")

else:

        if x==1 : print("it's 1")

        else : print("it's not 0 or 1")
```

Nested Conditionals

Be careful what belongs to what!

```
|if x==0 :
|    |print("it's 0")
|else:
|    |if x==1 : print("it's 1")
|    |else : print("it's not 0 or 1")
```

Nested Conditionals

Be careful what belongs to what!

```
|if x!=0 :
|    |if x==1 : print("it's 1")
|    |else : print("it's not 0 or 1")
|else:
|    |print("it's 0")
```

Let's practice!

Nested Conditionals

An alternative is *elif*

```
if x==0 : print("it's 0")

elif x==1 : print("it's 1")

else : print("it's not 0 or 1")
```

Nested Conditionals

Using elif you verify a statement, like an if.

But only if the previous statement is false, like an else.

Nested Conditionals

```
if x==0 : print("it's 0")

elif x==1 : print("it's 1")

elif x==2 : print("it's 2")

    …

else : print("it's unknown")
```

Let's practice!

Loop

A loop is a sequence of actions repeated a known or unknown number of times.

Loop

For example:

Have you eaten? → Chicken (loop back)

Yes / No

Great!   You should

While loop

When you don't know how many times you have to perform a command, you need the *while loop*

while True : action

While loop

```
while True :

    actions to repeat
```

While loop

while ans != 'yes' :

ans = input()

Let's practice!

Counter

The counter is a variable that stores the number of repetitions of a certain loop.

Counter

```
counter = 0

while True :

    action

    counter += 1
```

Counter

```
counter = 0

while counter < 5 :

        print(counter)

        counter += 1
```

Let's practice!

For loop

When you know how many times you have to perform a command, you need the *for loop*

for *element* in *sequence* : action

For loop

for e in sequence :

actions

For loop

for char **in** "hello" :

    print(char)

range()

range(n) $\longrightarrow$ 0, 1, 2, ... , n-1

range(n , m) $\longrightarrow$ n, n+1, n+2, ... , m-1

range()

range(5) $\longrightarrow$ 0, 1, 2, 3 , 4

range(6 , 10) $\longrightarrow$ 6, 7, 8, 9

For loop

```
for n in range(5) :
    print(n)
```

Let's practice!