Chapter: Conditionals

Conditional – if elif else

➤The syntax might feel a bit strange

➤Whitespaces matter – forced indentation

➤Leads to indented formatted code

➤I personally was not a huge fan of this in the beginning and then it grew on me and now it feels "so obvious"

Conditional – if elif else

```
>>> a = True
>>> type(a)
<type 'bool'>
>>> if a:
        print "Truthful"

Truthful
```

Conditional – if elif else

```
>>> a = 100
>>> if a<10 :
        print "Single Digit"
else:
        print "May be double digit"


May be double digit
```

Conditional – if elif else

```
>>> a = 100
>>> if a<10 :
        print "Single Digit"
elif a < 100:
        print "Double Digit"
else :
        print "2+ digits"

2+ digits
```

Conditionals

Description	Operator
Equality	==
Inequality	!=
less than	<
Less than or equal	<=
Greater than	>
Greater than or equal	>=
Membership	in

🐍 Conditional operators

```
>>> x = 5
>>> y = 10
>>> x < y
True
>>> x < 5 and y < 20
False
>>> x < 6 and y < 20
True
>>> x < 5 or y < 20
True
>>> x < 5 and not y < 6
False
>>> x < 6 and not y < 6
True
```

Conditional operators – Cool & readable

```
>>> x = 5
>>> y = 10
>>> 3 < x < 10
True
>>> 3 < x < y < 20
True
>>> if x > 3 and x < y and y < 20 :
        print "Truthful"


Truthful
```

Loops – while loop

```
>>> a = ['KA','TN','DL','AP','KL','PY']
>>> while count < len(a):
        print a[count]
        count+=1


KA
TN
DL
AP
KL
PY
```

Loops – for loop

A better Pythonic way of previous while loop

```
>>> a = ['KA','TN','DL','AP','KL','PY']
>>> for itr in a:
        print itr


KA
TN
DL
AP
KL
PY
```

Loops – break and continue

```
>>> a =
['KA','TN','DL','AP','KL','PY']
>>> for itr in a:
        if itr == 'UP':
                break
else:
        print 'did not find code'


did not find code
```

```
# not Pythonic. Sake of example
>>> found = False
>>> for itr in a:
        if itr == 'UP':
                found = True
>>> if not found:
        print "Did not find what I
was looking for"


Did not find what I was looking
for
```