Conditions

This is code braching. The code will take different paths depending on what happens, and the value of variables.

In [ ]:

```
# Here we check if you have enough money (more than 3.00)
money = 3.10

if money > 3.00:
  print("you can buy this")  #true
else:
  print("you dont have the right amount") #false
```

```
you can buy this
```

This will also work with strings. Eg. You may be looking up names in a database, or passwords.

In [ ]:

```
fname = "Paul"   #this wont give me access as it has a captial
sname = "smith"
#in this example we test 2 things at the same time with 'and'

if fname == "paul" and sname == "smith":   #both must be true  (see the capital lett
  print("access granted")
  #and gate - both must be true
else:
  print("please contact admin")
```

```
please contact admin
```

Note: fname = "Paul" is assignmet , its storing "paul" fname == "Paul" is a test of equality

In [ ]:

```
# This time we use or, this is 1 or the other.

raining = True   # boolean   True or False    1 / 0
temp = 6         # its cold

if raining == True or temp < 5:   #only 1 of them has to be true
  print("get a taxi to work")
else:
  print("you can walk")
```

```
get a taxi to work
```

Check if a person is over 18 , and has a clubcard - they get a discount

In [ ]:

```
age = 20
clubcard = True

if age > 18 and clubcard == True:  #both must be true
  print("you can buy a discounted drink")
else:
  print("sorry no discount")
```

```
you can buy a discounted drink
```

In the example below we will use a list of names and loop over them to see if the name is in the list.

In [ ]:

```
names = ["bob", "anne", "homer", "anakin"]
found = False
target = "paul"

for n in range(0,len(names)):
  if names[n] == target:
    print("Found " + target)
    found = True

if found == False:
    print(target + " not found")
```

```
paul not found
```

There is also a built in function 'in' to do this. But be carful how you use it.

In [ ]:

```
if "home" in names:
    print("found")
else:
    print("not found")
```