Syntax of a while loop

while some_boolean_condition:

    #do something

else:

    #do something different

```python
In [1]: x = 0

while x < 5:
    print(f'The current value of x is {x}')
    x = x+1
```

```
The current value of x is 0
The current value of x is 1
The current value of x is 2
The current value of x is 3
The current value of x is 4
```

```python
In [2]: x = 50

while x < 5:
    print(f'The current value of x is {x}')
    x += 1
else:
    print("X IS NOT LESS THAN 5")
```

```
X IS NOT LESS THAN 5
```

break, continue, pass

We can use break, continue, and pass statements in our loops to add additional functionality for various cases. The three statements are defined by:

break: Breaks out of the current closest enclosing loop. continue: Goes to the top of the closest enclosing loop. pass: Does nothing at all.

```python
In [5]: x = [1,2,3]

for item in x:
    #comment

    pass

print('end of my script')
```

```
end of my script
```

```python
In [8]: mystring = 'sammy'
```

```python
In [9]: for letter in mystring:
    print(letter)
```

```
s
a
m
m
y
```

```python
In [10]: for letter in mystring:
    if letter == 'a':
        continue
    print(letter)
```

```
s
m
m
y
```

```python
In [11]: for letter in mystring:
    if letter == 'a':
        break
    print(letter)
```

```
s
```

```python
In [12]: x = 0

while x<5:
    print(x)
    x+=1
```

```
0
1
2
3
4
```

```python
In [13]: x = 0

while x<5:

    if x == 2:
        break
    print(x)
    x+=1
```

```
0
1
```

```python
In [ ]:
```