

Common Fixes For Failing Python Scripts

- **Have you indented your code correctly?** - Commonly used items like functions and if statements require their contents to be indented from the first line, for example:

Wrong:

```
def conversion(a, b, d):
ca = math.cos(a/2)
cb = math.cos(b/2)
cd = math.cos(d/2)
sa = math.sin(a/2)
sb = math.sin(b/2)
sd = math.sin(d/2)

w = (ca*cb*cd)-(sa*sb*sd)
x = (sa*sb*cd)+(ca*cb*sd)
y = (sa*cb*cd)+(ca*sb*sd)
z = (ca*sb*cd)-(sa*cb*sd)

return Quaternion(w, x, y, z)
```

Correct:

```
def conversion(a, b, d):
    ca = math.cos(a/2)
    cb = math.cos(b/2)
    cd = math.cos(d/2)
    sa = math.sin(a/2)
    sb = math.sin(b/2)
    sd = math.sin(d/2)

    w = (ca*cb*cd)-(sa*sb*sd)
    x = (sa*sb*cd)+(ca*cb*sd)
    y = (sa*cb*cd)+(ca*sb*sd)
    z = (ca*sb*cd)-(sa*cb*sd)

    return Quaternion(w, x, y, z)
```

Wrong:

```
if dx1 < 0:
return 1000
```

Correct:

```
if dx1 < 0:
    return 1000
```

- **Have you imported the libraries that contain all of the functions that you have used in your script?** - If you have used a function from a particular python library you must ensure you have imported that library in the script otherwise the computer will not know what this function is. For example if I used `math.sin(0)` in my script I would have to include `import math` in my script otherwise the computer won't know what `math.sin` is.
- **Are there brackets and a colon after your function name?** - Even if your function doesn't require any inputs it must contain a set of brackets after your function name but in this case they would be empty. Also you must include a colon after your function name but before the contents of your function. For example

```
def function():  
    blah = 5  
    return blah
```

- **Do you have the same amount of arguments in the defined function as when you call the function?** - If you define a function with say 6 arguments you must have some sort of import for each of those 6 arguments when you call the function otherwise it will not work.