Errors and Exception Handling

PIERIAN DATA

Complete Python Bootcamp

- Errors are bound to happen in your code!
- Especially when someone else ends up using it in an unexpected way.
- We can use error handling to attempt to plan for possible errors.

Complete Python Bootcamp

- For example, a user may try to write to a file that was only opened in **mode='r'**
- Currently if there is any type of error in your code, the entire script will stop.
- We can use Error Handling to let the script continue with other code, even if there is an error.

Complete Python Bootcamp

- We use three keywords for this:
  - **try:** This is the block of code to be attempted (may lead to an error)
  - **except:** Block of code will execute in case there is an error in **try** block
  - **finally:** A final block of code to be executed, regardless of an error.

File Edit View Insert Cell Kernel Widgets Help Python 3

In [251]:
```
def add(n1,n2):
    print(n1+n2)
```

In [252]:
```
add(10,20)
```

```
30
```

In [253]:
```
number1 = 10
```

In [*]:
```
number2 = input("Please provide a number: ")
```

```
Please provide a number:
```

File Edit View Insert Cell Kernel Widgets Help Python 3

```
print(n1+n2)
```

In [252]:
```
add(10,20)
```

```
30
```

In [253]:
```
number1 = 10
```

In [254]:
```
number2 = input("Please provide a number: ")
```

```
Please provide a number: 20
```

In [255]:
```
add(number1,number2)
```

```
---------------------------------------------------------------------
TypeError                                 Traceback (most rece
nt call last)
```

File Edit View Insert Cell Kernel Widgets Help Python 3

```
In [255]: add(number1,number2)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most rece
nt call last)
<ipython-input-255-2ab48a5d5ad6> in <module>()
----> 1 add(number1,number2)

<ipython-input-251-59b429e2351f> in add(n1, n2)
      1 def add(n1,n2):
----> 2     print(n1+n2)

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

```
In [ ]:
```

File Edit View Insert Cell Kernel Widgets Help Python 3

In [257]:
```
try:
    # WANT TO ATTEMPT THIS CODE
    # MAY HAVE AN ERROR
    result = 10 + 10
except:
    print("Hey it looks like you aren't adding correctly!")
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [259]:
```
try:
    # WANT TO ATTEMPT THIS CODE
    # MAY HAVE AN ERROR
    result = 10 + '10'
except:
    print("Hey it looks like you aren't adding correctly!")
```

```
Hey it looks like you aren't adding correctly!
```

In [258]:
```
result
```

Out[258]:
```
20
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [260]:
```
try:
    # WANT TO ATTEMPT THIS CODE
    # MAY HAVE AN ERROR
    result = 10 + '10'
except:
    print("Hey it looks like you aren't adding correctly!")
else:
    print("Add went well!")
    print(result)
```

```
Hey it looks like you aren't adding correctly!
```

In [258]:
```
result
```

Out[258]:
```
20
```

File Edit View Insert Cell Kernel Widgets Help Python 3

In [261]:
```
try:
    # WANT TO ATTEMPT THIS CODE
    # MAY HAVE AN ERROR
    result = 10 + 10
except:
    print("Hey it looks like you aren't adding correctly!")
else:
    print("Add went well!")
    print(result)
```

```
Add went well!
20
```

In [258]:
```
result
```

Out[258]: 20

File Edit View Insert Cell Kernel Widgets Help Python 3

In [ ]:

In [267]:
```
try:
    f = open('testfile','w')
    f.write("Write a test line")
except TypeError:
    print("There was a type error!")
except OSError:
    print('Hey you have an OS Error')
finally:
    print("I always run")
```

```
I always run
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [ ]:

In [267]:
```
try:
    f = open('testfile','r')
    f.write("Write a test line")
except TypeError:
    print("There was a type error!")
except OSError:
    print('Hey you have an OS Error')
finally:
    print("I always run")
```

```
I always run
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [ ]:

In [269]:
```
try:
    f = open('testfile','r')
    f.write("Write a test line")
except TypeError:
    print("There was a type error!")
except:
    print('All other exceptions!')
finally:
    print("I always run")
```

```
All other exceptions!
I always run
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [272]:
```
def ask_for_int():
    try:
        result = int(input("Please provide number: "))
    except:
        print("Whoops! That is not a number")
    finally:
        print("End of try/except/finally")
```

In [ ]:
```
ask_for_int
```

File Edit View Insert Cell Kernel Widgets Help Python 3

```
    try:
        result = int(input("Please provide number: "))
    except:
        print("Whoops! That is not a number")
    finally:
        print("End of try/except/finally")
```

In [273]:
```
ask_for_int()
```

```
Please provide number: 20
End of try/except/finally
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [272]:
```
def ask_for_int():
    try:
        result = int(input("Please provide number: "))
    except:
        print("Whoops! That is not a number")
    finally:
        print("End of try/except/finally")
```

In [*]:
```
ask_for_int()
```

```
Please provide number:
word
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [272]:
```
def ask_for_int():
    try:
        result = int(input("Please provide number: "))
    except:
        print("Whoops! That is not a number")
    finally:
        print("End of try/except/finally")
```

In [274]:
```
ask_for_int()
```

```
Please provide number: word
Whoops! That is not a number
End of try/except/finally
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

In [281]:
```
def ask_for_int():

    while True:
        try:
            result = int(input("Please provide number: "))
        except:
            print("Whoops! That is not a number")
            continue
        else:
            print("Yes thank you")
            break
        finally:
            print("End of try/except/finally")
            print("I will always run at the end!")
```

In [279]:
```
ask_for_int()
```

File Edit View Insert Cell Kernel Widgets Help Python 3

```
            continue
        else:
            print("Yes thank you")
            break
        finally:
            print("End of try/except/finally")
            print("I will always run at the end!")
```

In [*]:
```
ask_for_int()
```

```
Please provide number:
20
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help | Python 3

```
            continue
        else:
            print("Yes thank you")
            break
        finally:
            print("End of try/except/finally")
            print("I will always run at the end!")
```

In [283]:
```
ask_for_int()
```

```
Please provide number: 20
Yes thank you
End of try/except/finally
I will always run at the end!
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

```
            continue
        else:
            print("Yes thank you")
            break
        finally:
            print("End of try/except/finally")
            print("I will always run at the end!")
```

In [*]:
```
ask_for_int()
```

```
Please provide number:
q
```

In [ ]:

File Edit View Insert Cell Kernel Widgets Help Python 3

```
            continue
        else:
            print("Yes thank you")
            break
        finally:
            print("End of try/except/finally")
            print("I will always run at the end!")
```

In [*]:
```
ask_for_int()
```

```
Please provide number: q
Whoops! That is not a number
End of try/except/finally
I will always run at the end!

Please provide number:
```

File Edit View Insert Cell Kernel Widgets Help Python 3

In [282]:
```
def ask_for_int():

    while True:
        try:
            result = int(input("Please provide number: "))
        except:
            print("Whoops! That is not a number")
            continue
        else:
            print("Yes thank you")
            break
        finally:
            print("I'm going to ask you again! \n")
```

In [284]:
```
ask_for_int()
```