If Else and control flow.

Due Feb 21, 2022. 200pts.

Goals

Create a program that when input the weight of a package prints out the cost for shipping the package.

From Class, and example of the age/ticket price:

```
 1: print ( "Enter Age" )
 2: age_str = input()
 3: age = int(age_str)
 4:
 5: ticket_price = 0
 6: if age >= 0 and age <= 4:
 7:     ticket_price = 0
 8: elif age >= 5 and age <= 12:
 9:     ticket_price = 38
10: elif age >= 13 and age <= 17:
11:     ticket_price = 48
12: elif age >= 18:
13:     ticket_price = 62
14: else:
15:     print ( "Impossile Age" )
16:
17: print ( "Price = {}".format(ticket_price) )
```

Built as a function that you can test:

File: age2.py

```
 1: def calc_ticket_price(age):
 2:     ticket_price = 0
 3:     if age >= 0 and age <= 4:
 4:         ticket_price = 0
 5:     elif age >= 5 and age <= 12:
 6:         ticket_price = 38
 7:     elif age >= 13 and age <= 17:
 8:         ticket_price = 48
 9:     elif age >= 18:
10:         ticket_price = 62
11:     else:
12:         print ( "Impossile Age" )
13:     return(ticket_price)
14:
15:
16: # Automated Test
17: if __name__ == "__main__":
18:     n_err = 0
19:
20:     got = calc_ticket_price(13)
```

```
21:     expect = 48
22:     if got != expect:
23:         n_err = n_err + 1
24:         print ( "Error: Test 1: expected {} got {}".
25:                 format (  expect, got ) )
26:
27:     got = calc_ticket_price(18)
28:     expect = 62
29:     if got != expect:
30:         n_err = n_err + 1
31:         print ( "Error: Test 1: expected {} got {}".
32:                 format (  expect, got ) )
33:
34:     if n_err == 0 :
35:         print ( "PASS" )
36:     else:
37:         print ( "FAILED" )
38:
```

and as a main program that calls the function:

File: age_main.py

```
1: import age2
2:
3: print ( "Enter Age" )
4: age_str = input()
5: age = int(age_str)
6:
7: ticket_price = age2.calc_ticket_price(age)
8:
9: print ( "Price = {}".format(ticket_price) )
```

Using Visual Studio Code (VSCode)

Create a new directory, lets say `hw4` .

Start VS Code. Open the directory that you created.

Create a new file called `shipping_cost.py`

Create the lookup with if/else as a function. You will want to copy the code from `age2.py` and modify it.

Run it. Test it.

Copy the main program, `age_main.py` , to a new file name, lets say `shipping_main.py` .

Modify it to call your new program. Run it. Check to see if 999 is returned. If it is then print “Too Heavy”. If the value is not 999 then print out the shipping cost. Alternatively test some cases (run your program a few times).

Modify The Code for the rest of the table.

Weight	Cost
0 to 2	4
3 to 4	6
5 to 7	10
8 to 10	14
11 to 15	18
16 to 25	22
26 to 35	28
36 and over	999

Testing your code

Run your code with different weights as input and see if your code is printing the output according to the values in the table above.

Turn in

Your python two python files.