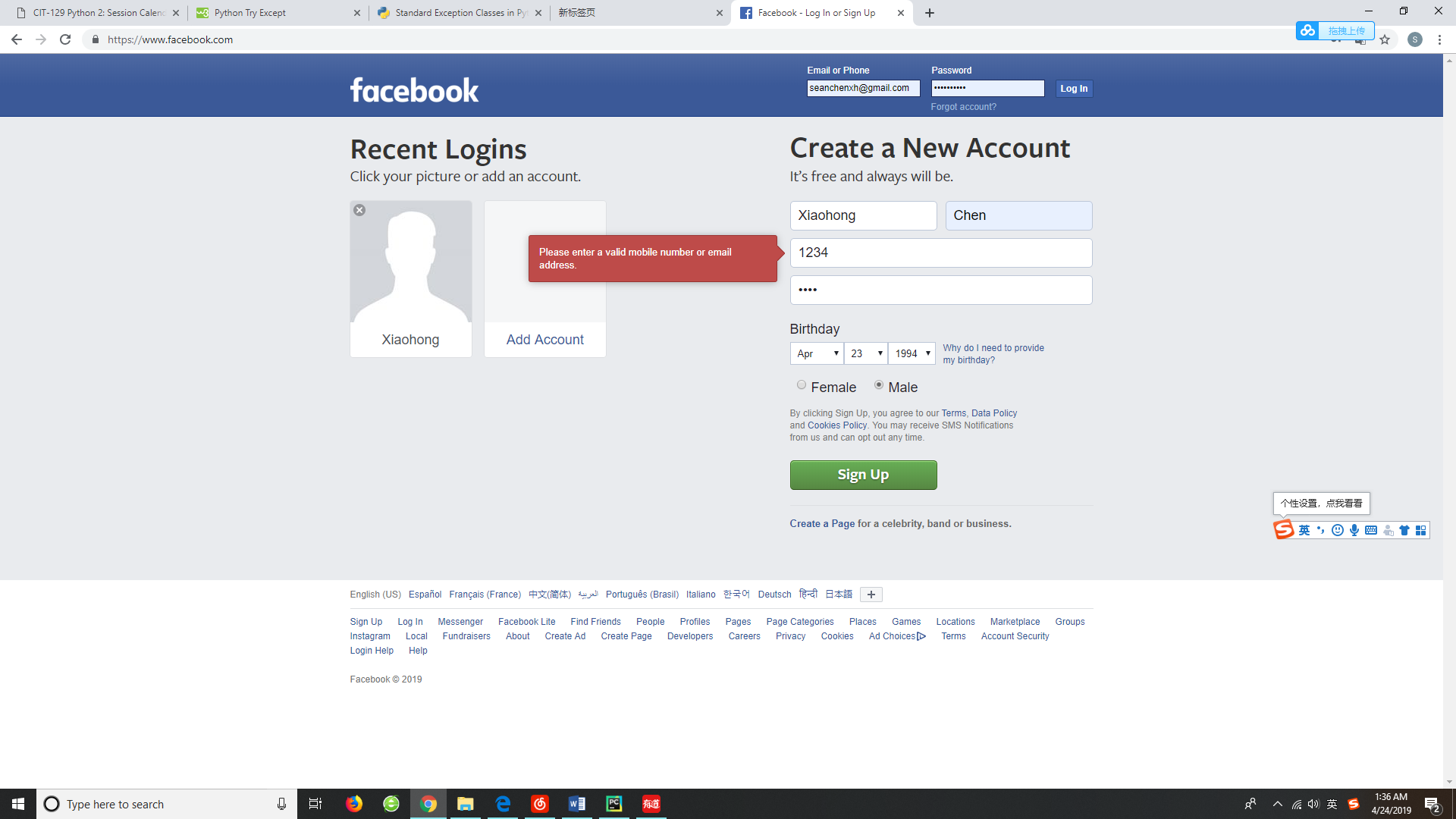
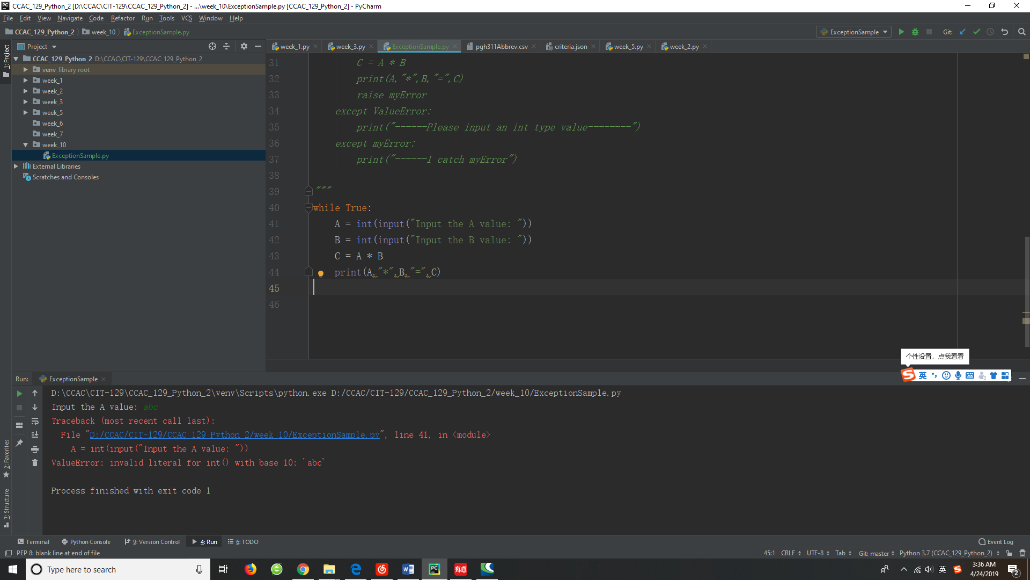
**Why use Exceptions?**

**Case 1**: sign up in Facebook

****

**Case 2**: calculate A \* B in Python (What if A=abd or B=efg?)

**while True:  
 a = int(input("Input the a value: "))  
 b = int(input("Input the b value: "))  
 c=a\*b  
 print(c)**

****

**Q:When an error or exception occurs, Python will normally stop and generate an error message. However, what should we do if we hope that python or website work as usual instead of discontinuing?**

**A: Using an exception handler(try statement) to catch and respond to the errors or ignore the exceptions.**

**1.try/except Catch and recover from exceptions raised by Python, or by you.**

**while True:  
 try:  
 A = int(input("Input the A value: "))  
 B = int(input("Input the B value: "))  
 C = A \* B  
 print(A,"\*",B,"=",C)  
 except ValueError:  
 print("--Please input an int type value--")**

**2.try/finally finally block gets executed no matter if the try block raises any errors or not**

**while True:  
 try:  
 A = int(input("Input the A value: "))  
 B = int(input("Input the B value: "))  
 C = A \* B  
 print(A,"\*",B,"=",C)  
 except ValueError:  
 print("--Please input an int type value--")  
 finally:  
 print("—Next calculation will begin! --")**

**3.raise (extension exercise) Trigger an exception manually in your code.**

**class myError(Exception): pass  
try:  
 raise myError  
except myError:  
 print(“—executed or not?--”)**