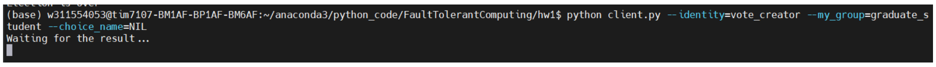
Fault Tolerant Computing Project 2

Implementation

In my implementation, there are three parts:

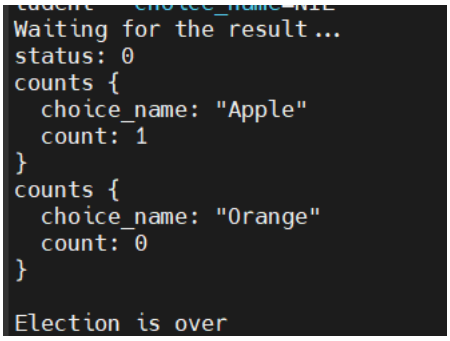
1. Server
   The server is the platform which client can make vote system and voting.

2. Vote creator
   This is the creator of the vote system, and print the result of each candidate after the time expired. Note that the time expired is the only one way to closed the vote system validly.
   After the vote system be created successfully, we can see the information:



```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=vote_creator --my_group=graduate_s
tudent --choice_name=NIL
Waiting for the result...
```

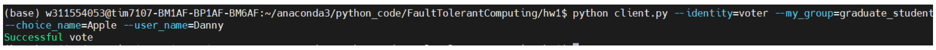
   And after the time expired, close the vote system and show the total ballots of each candidate:



```
Waiting for the result...
status: 0
counts {
  choice_name: "Apple"
  count: 1
}
counts {
  choice_name: "Orange"
  count: 0
}

Election is over
```

3. Voter
   The only one function of the Voter is vote to the candidate and print ‘successful’ while voting done successfully. Note that every voter has only one ballot.
   If the voter act successfully, we can see the output:



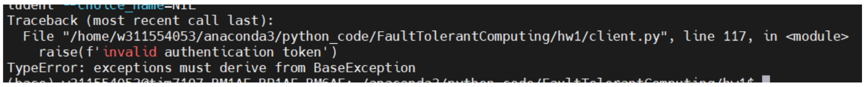
```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=voter --my_group=graduate_student
--choice_name=Apple --user_name=Danny
Successful vote
```

Notice that Voter and Vote creator are one kind of client, so they must login to the server before doing action.

judge the correctness of each status

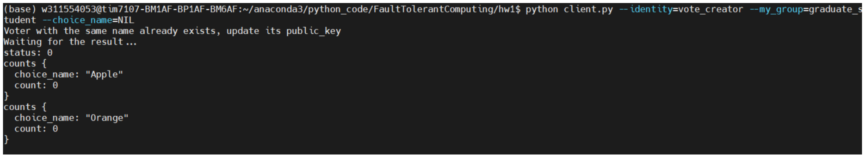
- CreateElection

1. Status.code=1 : Invalid authentication token
   For example we set the auth_token by 0, then got the error:



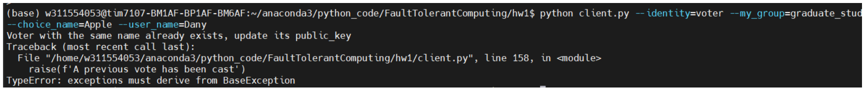
```
tudent --choice_name=NIL
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 117, in <module>
    raise(f'invalid authentication token')
TypeError: exceptions must derive from BaseException
```

- RegisterVoter
  1. Status.code=1 : Voter with the same name already exists
     This case will happened while we use the same user name, in my implementation, the registered public_key will be updated to the new one, like the row 3 in the figure below.



```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=vote_creator --my_group=graduate_s
tudent --choice_name=NIL
Voter with the same name already exists, update its public_key
Waiting for the result...
status: 0
counts {
  choice_name: "Apple"
  count: 0
}
counts {
  choice_name: "Orange"
  count: 0
}
```

- UnregisterVoter
  1. Status.code=1 : No voter with the name exists on the server
     If we registered name “Danny” but desired to unregistered name “Dann”, the error will occur:

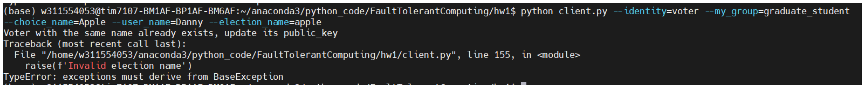


```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=voter --my_group=graduate_stud
--choice_name=Apple --user_name=Dany
Voter with the same name already exists, update its public_key
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 158, in <module>
    raise(f'A previous vote has been cast')
TypeError: exceptions must derive from BaseException
```

- CastVote

  1. Status.code=1 : Invalid authentication token
     Same as the modify mention above, we could get the same result, because they use same code snippet

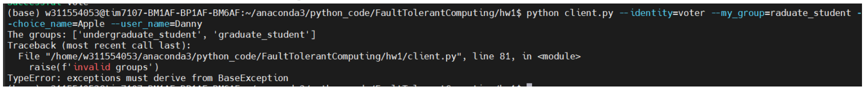
  2. Status.code=2 : Invalid election name
     When we try to vote on the “Apple” vote system, we will get error because there are no vote named “Apple”.



```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=voter --my_group=graduate_student
--choice_name=Apple --user_name=Danny --election_name=apple
Voter with the same name already exists, update its public_key
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 155, in <module>
    raise(f'Invalid election name')
TypeError: exceptions must derive from BaseException
```

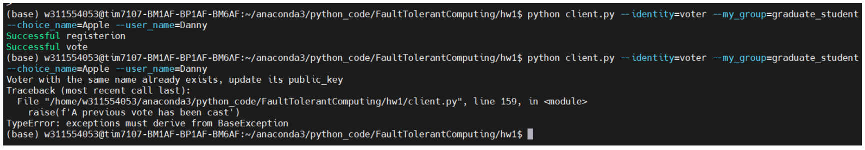
  3. Status.code=3 : The voter’s group is not allowed in the election
     By modify the graduate_student to

raduate_student, got the result



```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=voter --my_group=raduate_student -
-choice_name=Apple --user_name=Danny
The groups: ['undergraduate_student', 'graduate_student']
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 81, in <module>
    raise(f'invalid groups')
TypeError: exceptions must derive from BaseException
```

4. Status.code=4 : A previous vote has been cast
   When we want to vote 2 times, system will raise an error because everyone has one vote in current design.

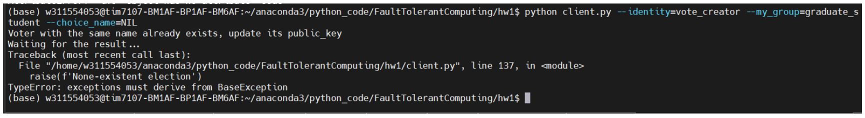


```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=voter --my_group=graduate_student
--choice_name=Apple --user_name=Danny
Successful registerion
Successful vote
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=voter --my_group=graduate_student
--choice_name=Apple --user_name=Danny
Voter with the same name already exists, update its public_key
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 159, in <module>
    raise(f'A previous vote has been cast')
TypeError: exceptions must derive from BaseException
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$
```

   we can see that the first time, it is vote successfully, but raise an error at the second time.

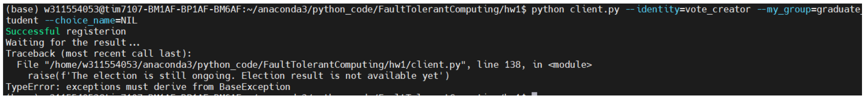
- GetResult

  1. ElectionResult.status = 1: Non-existent election
     After the system is waiting for the result (i.e. the system is start up), we change the “election_name” to “Apple”, then an error happened.



```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=vote_creator --my_group=graduate_s
tudent --choice_name=NIL
Voter with the same name already exists, update its public_key
Waiting for the result...
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 137, in <module>
    raise(f'None-existent election')
TypeError: exceptions must derive from BaseException
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$
```

  2. ElectionResult.status = 2: The election is still ongoing. Election result is not available yet
     If the end_date on the server was be extended, but the vote_creator did not know, this error will happened.



```
(base) w311554053@tim7107-BM1AF-BP1AF-BM6AF:~/anaconda3/python_code/FaultTolerantComputing/hw1$ python client.py --identity=vote_creator --my_group=graduate_
tudent --choice_name=NIL
Successful registerion
Waiting for the result...
Traceback (most recent call last):
  File "/home/w311554053/anaconda3/python_code/FaultTolerantComputing/hw1/client.py", line 138, in <module>
    raise(f'The election is still ongoing. Election result is not available yet')
TypeError: exceptions must derive from BaseException
```

So far, all status can be output correctly.