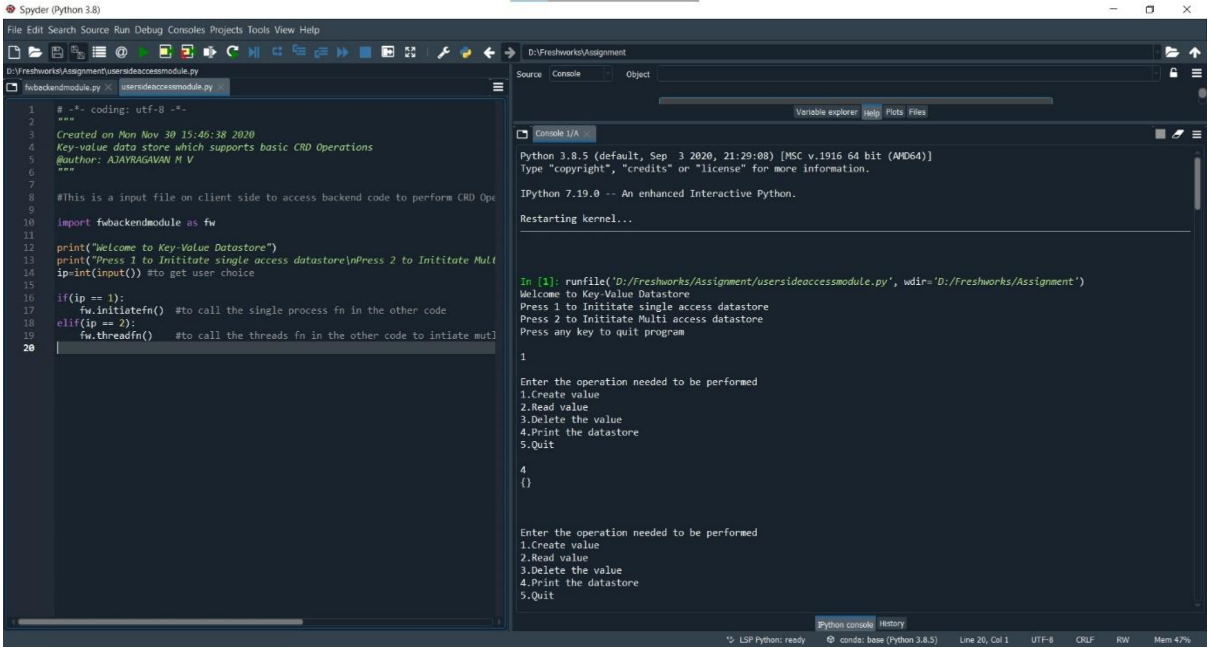
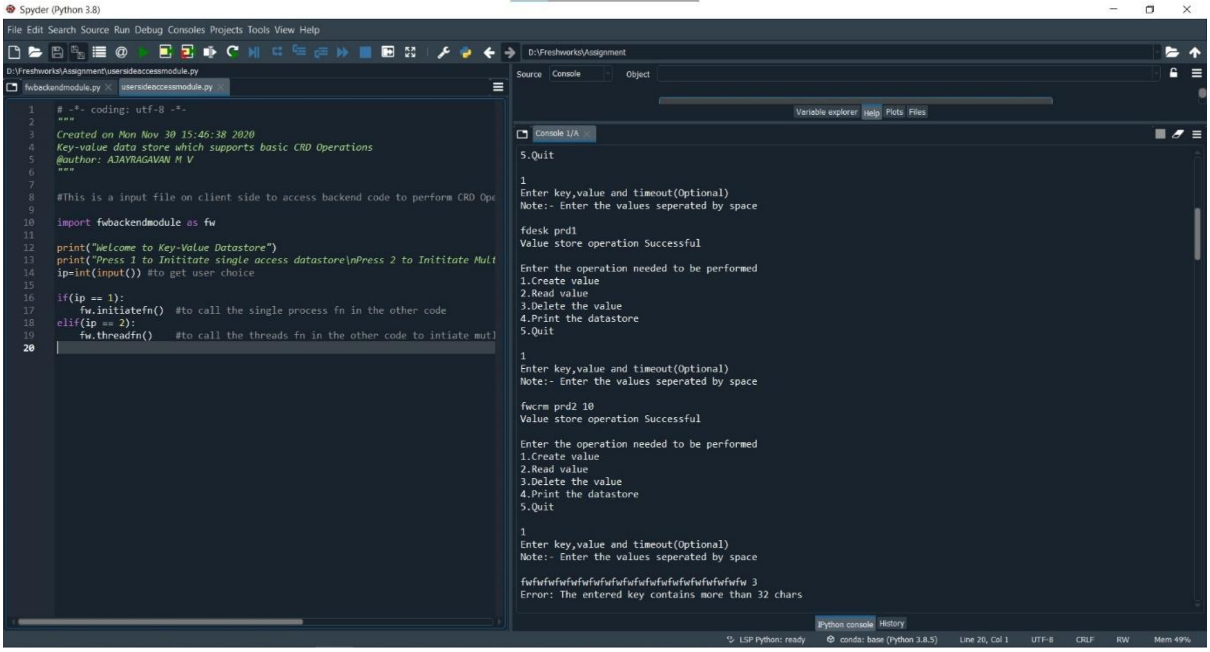
Key value datastore – Basic CRD Operations O/P

Single Process operation



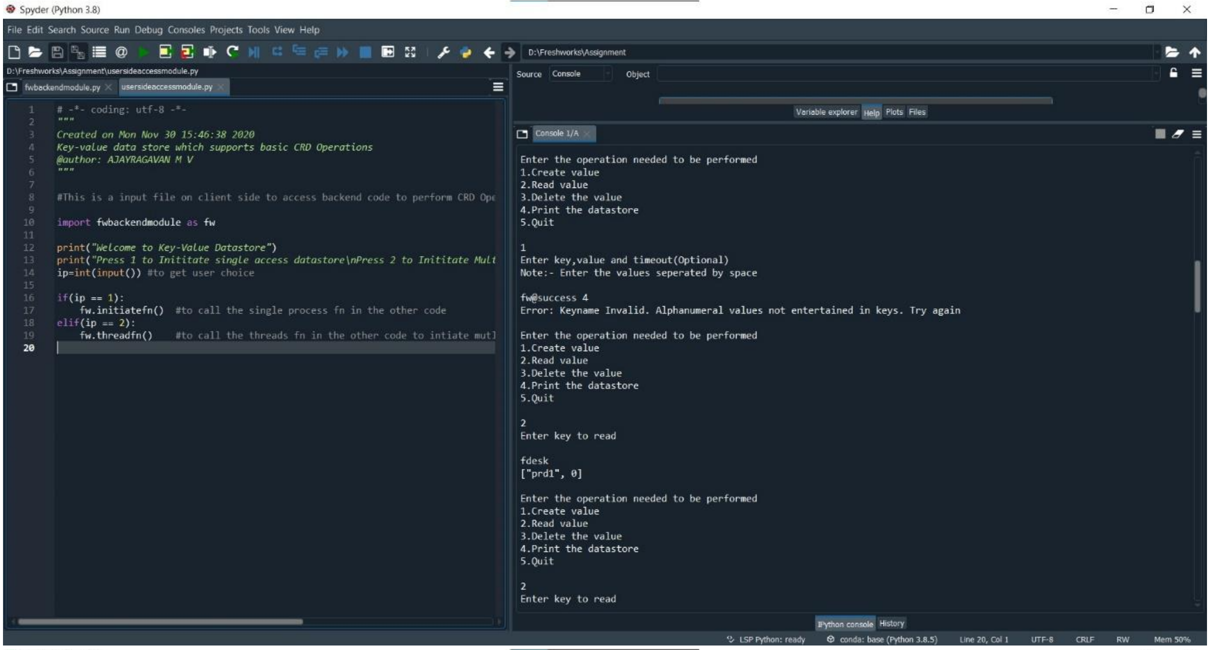
```python
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl
```

```
Python 3.8.5 (default, Sep  3 2020, 21:29:08) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.19.0 -- An enhanced Interactive Python.

Restarting kernel...



In [1]: runfile('D:/Freshworks/Assignment/usersideaccessmodule.py', wdir='D:/Freshworks/Assignment')
Welcome to Key-Value Datastore
Press 1 to Inititate single access datastore
Press 2 to Inititate Multi access datastore
Press any key to quit program

1

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

4
{}



Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit
```



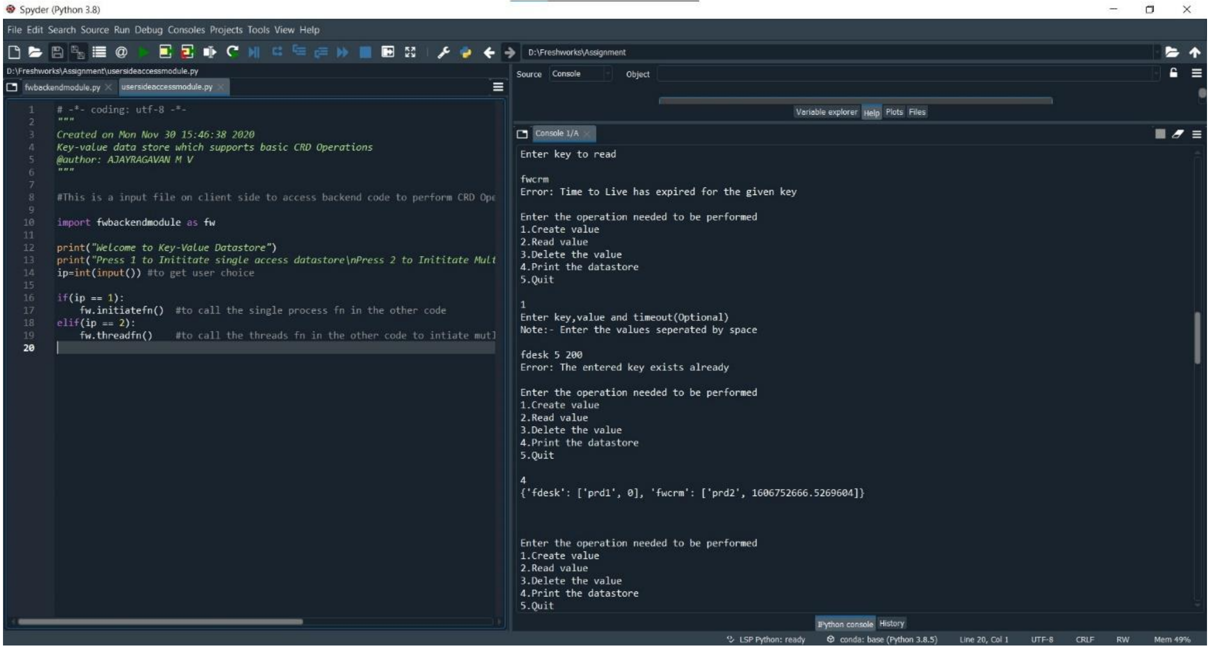
```python
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl
```

```
5.Quit

1
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fdesk prd1
Value store operation Successful

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

1
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fwcrm prd2 10
Value store operation Successful

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

1
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fwfwfwfwfwfwfwfwfwfwfwfwfwfwfwfwfwfwfw 3
Error: The entered key contains more than 32 chars
```



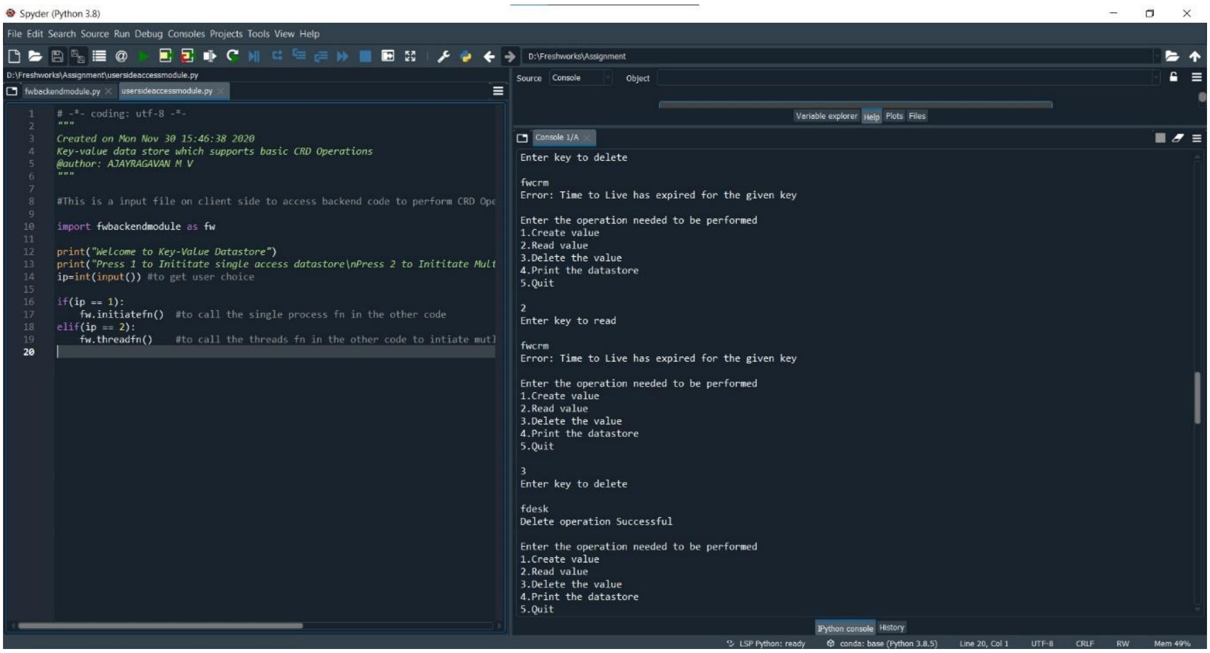
```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl
```

```
Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

1
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fw@success 4
Error: Keyname Invalid. Alphanumeral values not entertained in keys. Try again

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fdesk
["prd1", 0]

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read
```



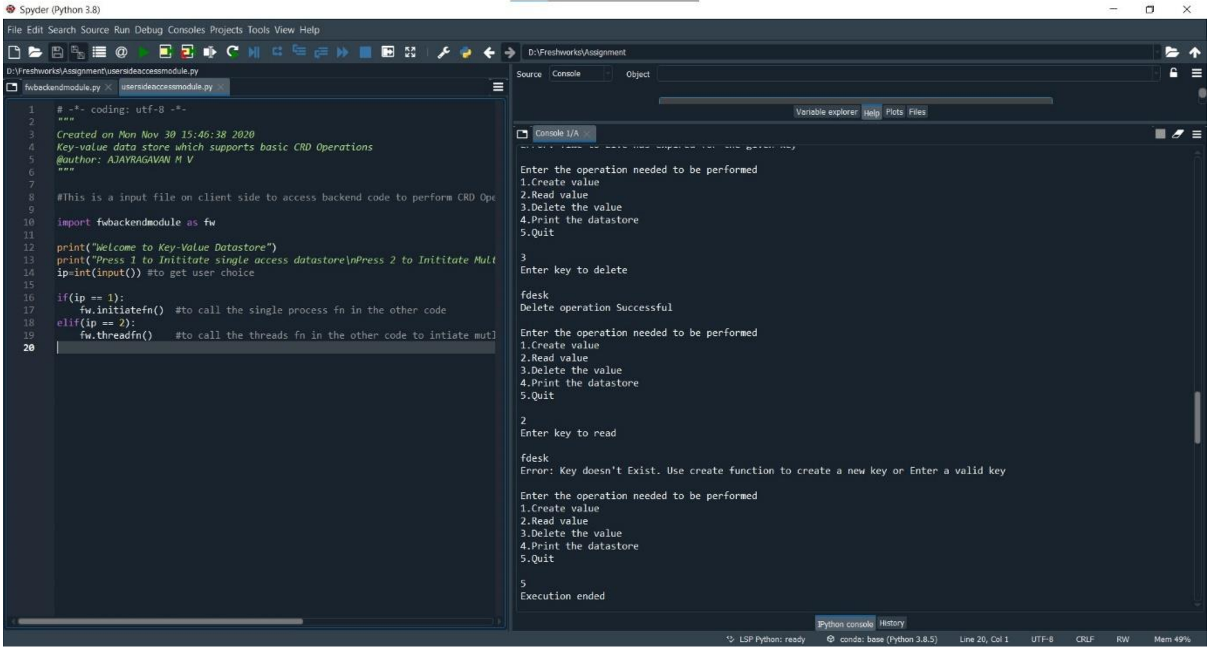
```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl
```

```
Enter key to read

fwcrm
Error: Time to Live has expired for the given key

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

1
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fdesk 5 200
Error: The entered key exists already

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

4
{'fdesk': ['prd1', 0], 'fwcrm': ['prd2', 1606752666.5269604]}



Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit
```


Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Freshworks\Assignment\usersideaccessmodule.py
fwbackendmodule.py
usersideaccessmodule.py
D:\Freshworks\Assignment
Source Console Object
Variable explorer Help Plots Files
Console 1/A
IPython console History
LSP Python: ready conda: base (Python 3.8.5) Line 20, Col 1 UTF-8 CRLF RW Mem 49%

```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl

```

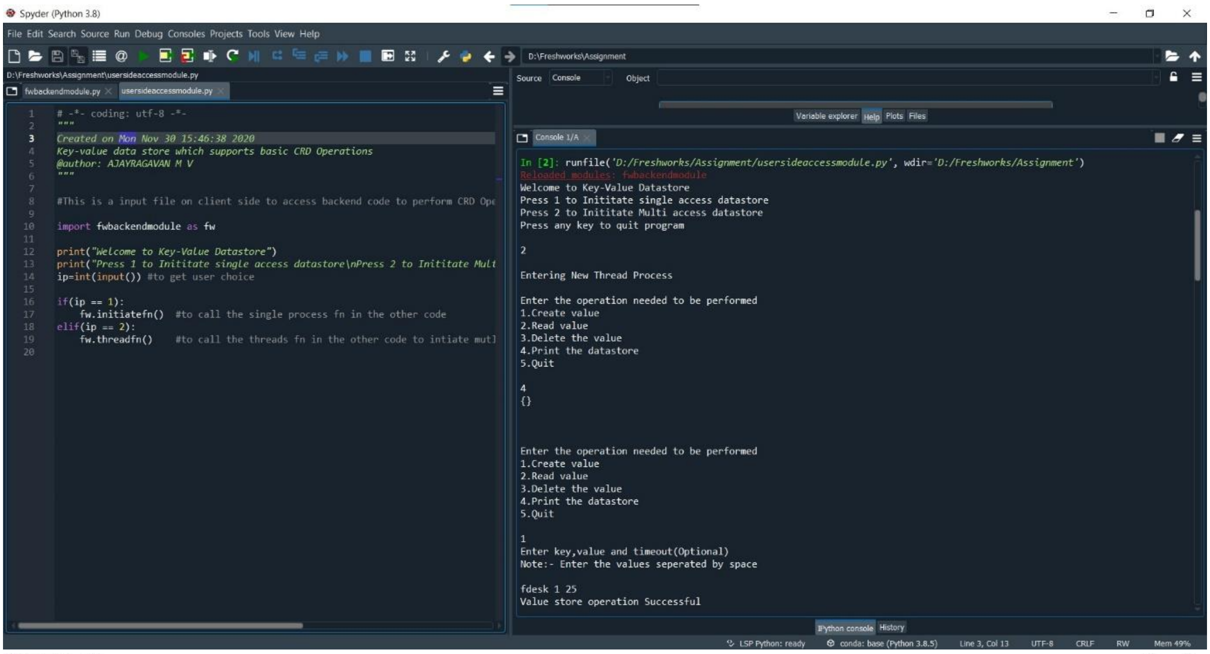
```
Enter key to delete

fwcrm
Error: Time to Live has expired for the given key

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fwcrm
Error: Time to Live has expired for the given key

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

3
Enter key to delete

fdesk
Delete operation Successful

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit
```


Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Freshworks\Assignment\usersideaccessmodule.py
fwbackendmodule.py
usersideaccessmodule.py
D:\Freshworks\Assignment
Source Console Object
Variable explorer Help Plots Files
Console 1/A
IPython console History
LSP Python: ready conda: base (Python 3.8.5) Line 20, Col 1 UTF-8 CRLF RW Mem 49%

```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl

```

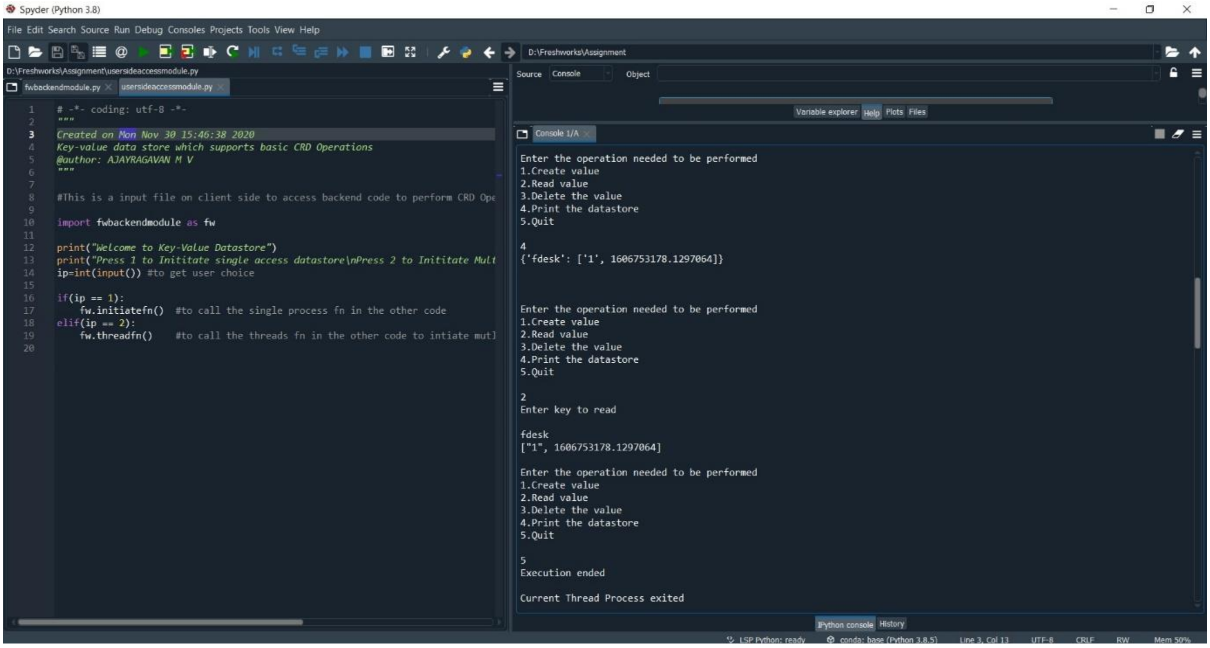
```
Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

3
Enter key to delete

fdesk
Delete operation Successful

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fdesk
Error: Key doesn't Exist. Use create function to create a new key or Enter a valid key

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

5
Execution ended
```

Multi thread op – Single thread execution at a time



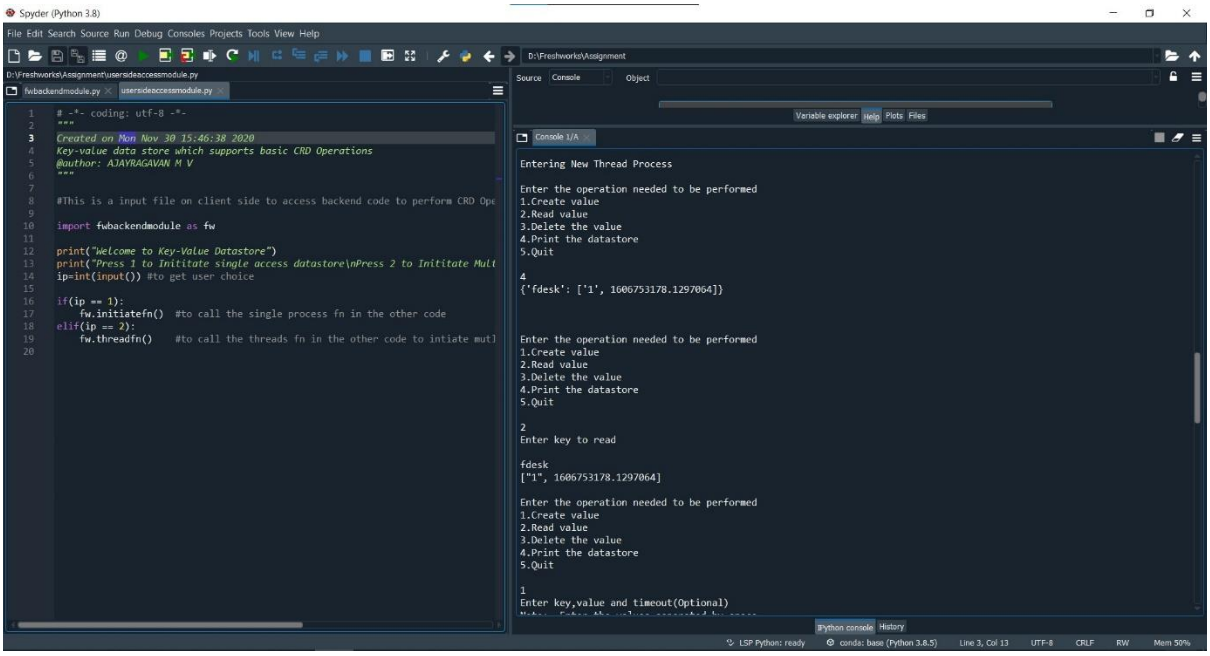
```python
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl
```

```
In [2]: runfile('D:/Freshworks/Assignment/usersideaccessmodule.py', wdir='D:/Freshworks/Assignment')
Reloaded modules: fwbackendmodule
Welcome to Key-Value Datastore
Press 1 to Inititate single access datastore
Press 2 to Inititate Multi access datastore
Press any key to quit program

2

Entering New Thread Process

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

4
{}



Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

1
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fdesk 1 25
Value store operation Successful
```



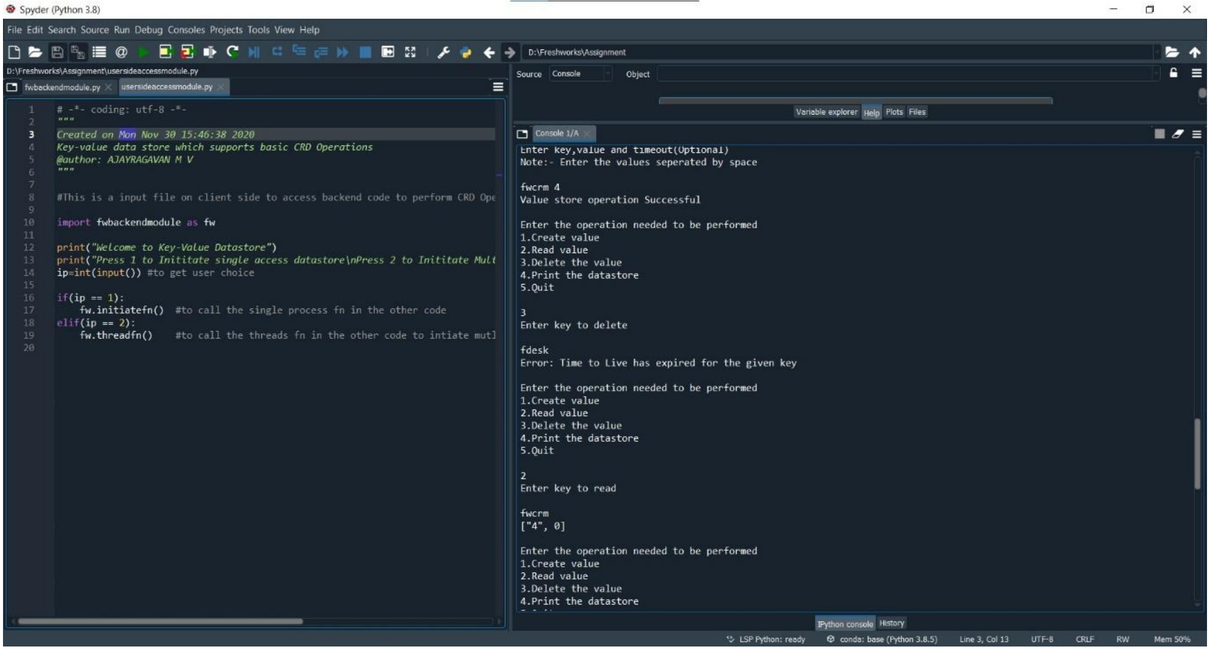
```python
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl
```

```
Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

4
{'fdesk': ['1', 1606753178.1297064]}



Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fdesk
["1", 1606753178.1297064]

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

5
Execution ended

Current Thread Process exited
```


Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Freshworks\Assignment
D:\Freshworks\Assignment\usersideaccessmodule.py
fwbackendmodule.py
usersideaccessmodule.py
Source Console Object
Variable explorer Help Plots Files
Console 1/A
IPython console History
LSP Python: ready
conda: base (Python 3.8.5)
Line 3, Col 13 UTF-8 CRLF RW Mem 50%

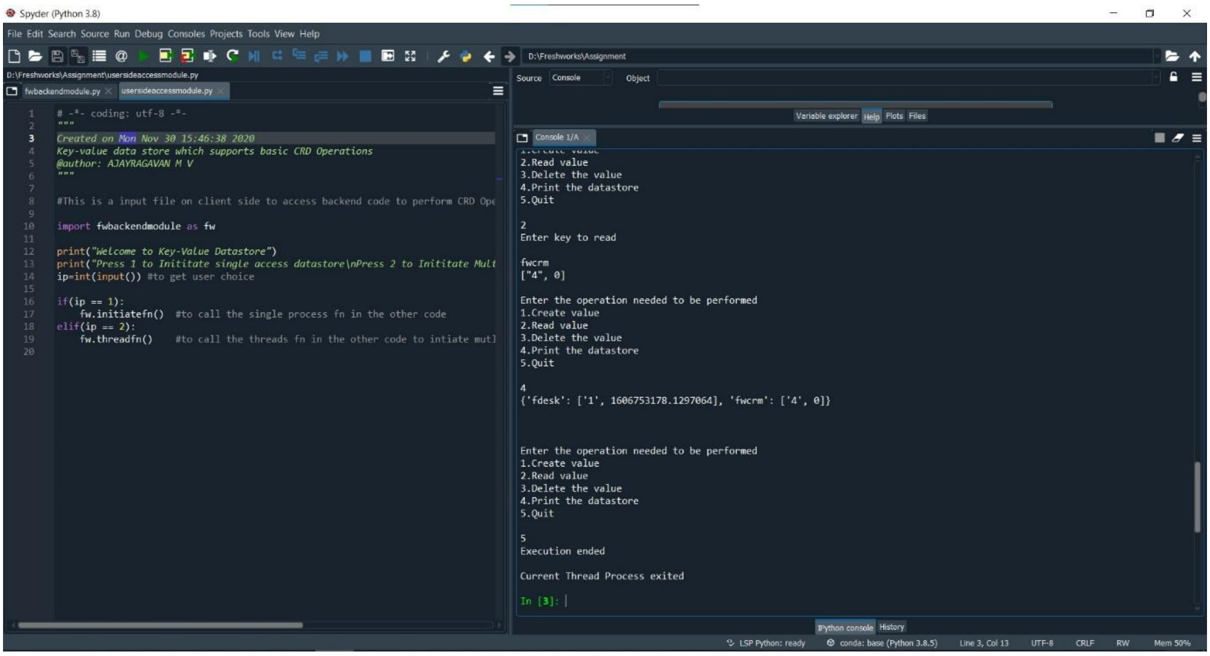
```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl

```

```
Entering New Thread Process

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

4
{'fdesk': ['1', 1606753178.1297064]}



Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fdesk
["1", 1606753178.1297064]

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

1
Enter key,value and timeout(Optional)
```


Spyder (Python 3.8)
File Edit Search Source Run Debug Consoles Projects Tools View Help
D:\Freshworks\Assignment
D:\Freshworks\Assignment\usersideaccessmodule.py
fwbackendmodule.py
usersideaccessmodule.py
Source Console Object
Variable explorer Help Plots Files
Console 1/A
IPython console History
LSP Python: ready
conda: base (Python 3.8.5)
Line 3, Col 13 UTF-8 CRLF RW Mem 50%

```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""

#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl

```

```
Enter key,value and timeout(Optional)
Note:- Enter the values seperated by space

fwcrm 4
Value store operation Successful

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

3
Enter key to delete

fdesk
Error: Time to Live has expired for the given key

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fwcrm
["4", 0]

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
```



```
# -*- coding: utf-8 -*-
"""
Created on Mon Nov 30 15:46:38 2020
Key-value data store which supports basic CRD Operations
@author: AJAYRAGAVAN M V
"""


#This is a input file on client side to access backend code to perform CRD Ope

import fwbackendmodule as fw

print("Welcome to Key-Value Datastore")
print("Press 1 to Inititate single access datastore\nPress 2 to Inititate Mult
ip=int(input()) #to get user choice

if(ip == 1):
    fw.initiatefn()  #to call the single process fn in the other code
elif(ip == 2):
    fw.threadfn()    #to call the threads fn in the other code to intiate mutl

```

```
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

2
Enter key to read

fwcrm
["4", 0]

Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

4
{'fdesk': ['1', 1606753178.1297064], 'fwcrm': ['4', 0]}



Enter the operation needed to be performed
1.Create value
2.Read value
3.Delete the value
4.Print the datastore
5.Quit

5
Execution ended

Current Thread Process exited

In [3]: 
```