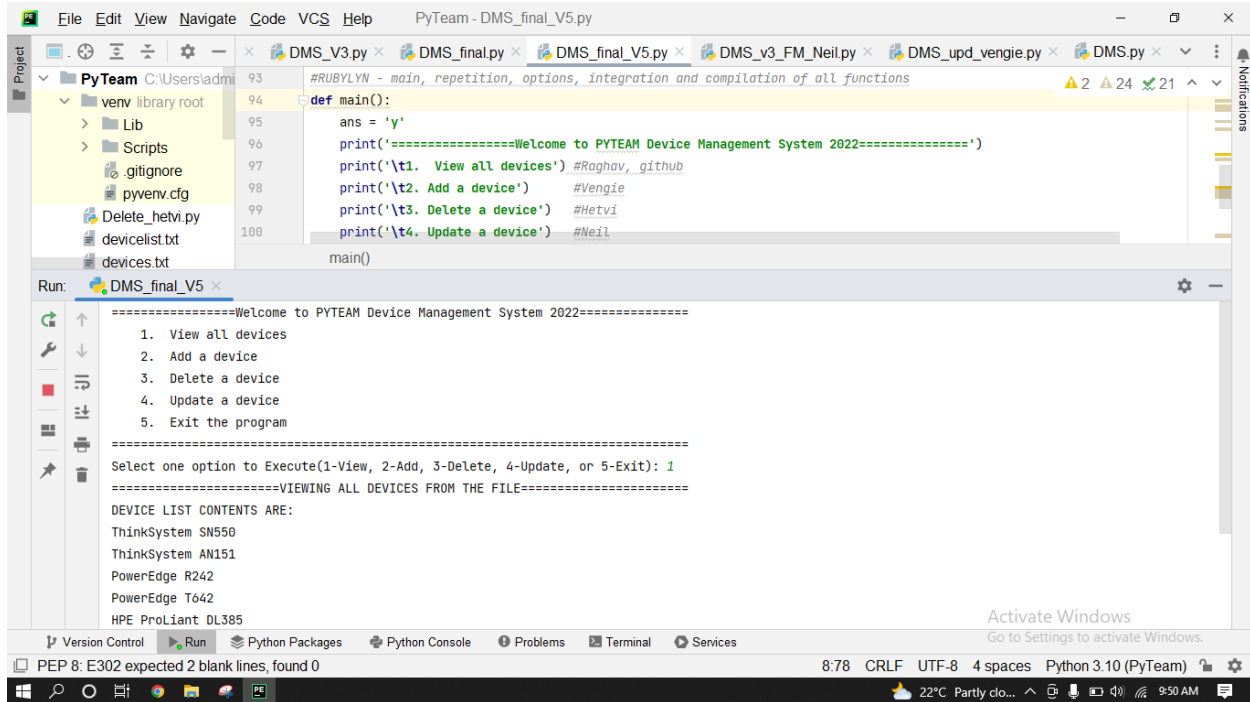


DEVICE MANAGEMENT SYSTEM PROJECT

SAMPLE OUTPUT 1: MAIN MENU OPTIONS

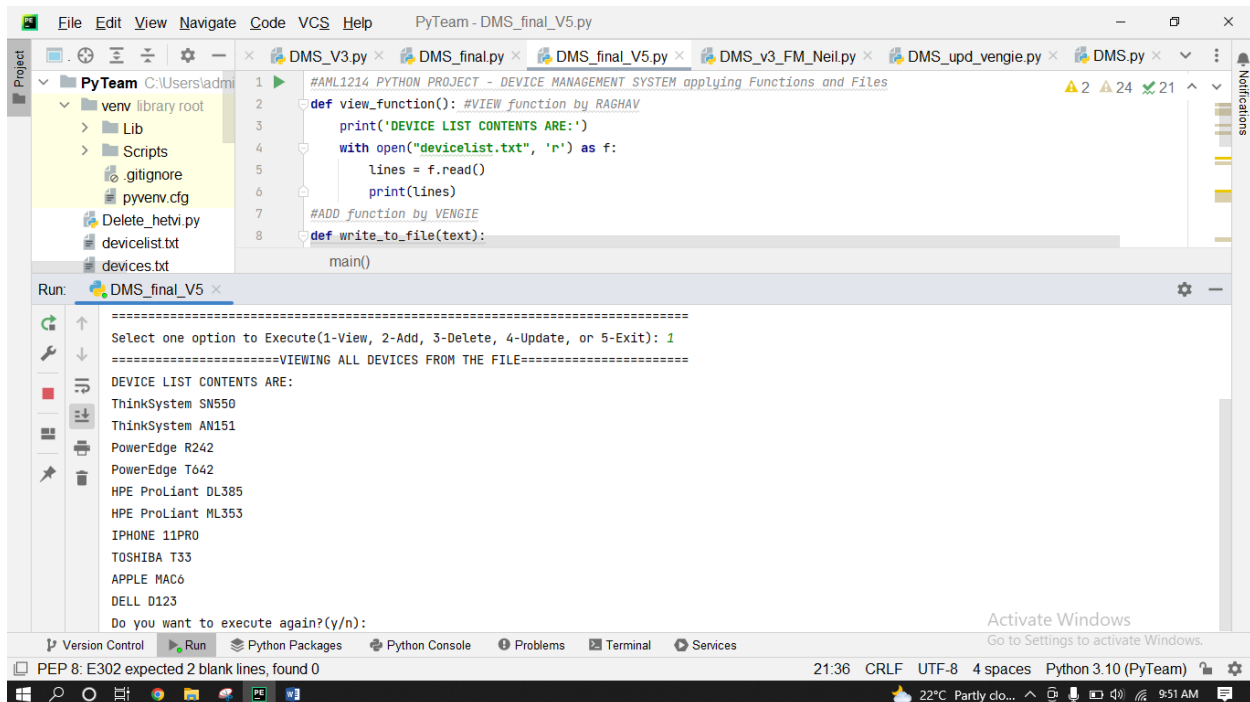


The screenshot shows the PyCharm IDE with the file `DMS_final_V5.py` open. The code defines a `main()` function that prints a welcome message and a list of options: 1. View all devices, 2. Add a device, 3. Delete a device, 4. Update a device, and 5. Exit the program. The Run console shows the output of the program, including the welcome message and the list of options. The user has selected option 1, and the program has printed the contents of the `deviceList.txt` file, which contains a list of devices: ThinkSystem SN550, ThinkSystem AN151, PowerEdge R242, PowerEdge T642, and HPE ProLiant DL385.

```
#RUBYLYN - main, repetition, options, integration and compilation of all functions
93
94 def main():
95     ans = 'y'
96     print('=====Welcome to PYTEAM Device Management System 2022=====')
97     print('\t1. View all devices') #Raghav, github
98     print('\t2. Add a device') #Vengie
99     print('\t3. Delete a device') #Hetvi
100    print('\t4. Update a device') #Neil
    main()
```

```
Run: DMS_final_V5
=====Welcome to PYTEAM Device Management System 2022=====
1. View all devices
2. Add a device
3. Delete a device
4. Update a device
5. Exit the program
=====
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): 1
=====VIEWING ALL DEVICES FROM THE FILE=====
DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
```

SAMPLE OUTPUT 2: VIEWING DEVICE LISTS

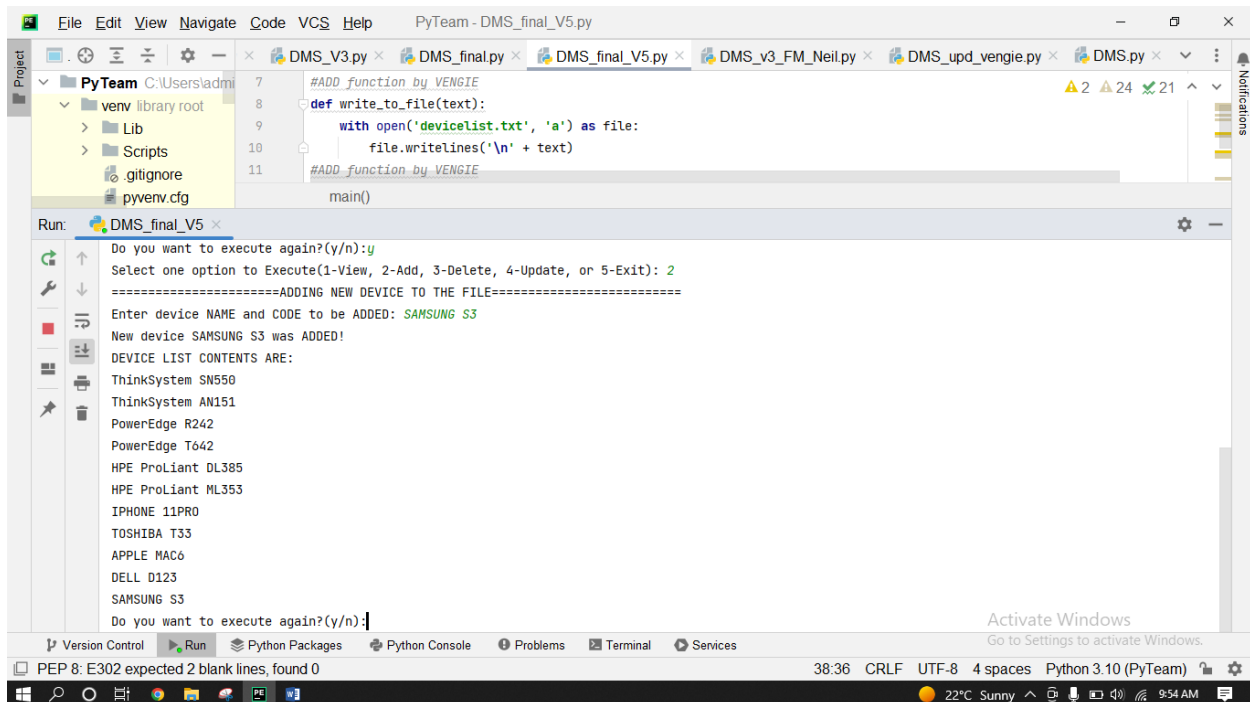


The screenshot shows the PyCharm IDE with the file `DMS_final_V5.py` open. The code defines a `view_function()` that reads the contents of the `deviceList.txt` file and prints them. The Run console shows the output of the program, including the welcome message and the list of options. The user has selected option 1, and the program has printed the contents of the `deviceList.txt` file, which contains a list of devices: ThinkSystem SN550, ThinkSystem AN151, PowerEdge R242, PowerEdge T642, HPE ProLiant DL385, HPE ProLiant ML353, IPHONE 11PRO, TOSHIBA T33, APPLE MAC6, and DELL D123. The user has also entered 'y' to execute the program again.

```
1 #AML1214 PYTHON PROJECT - DEVICE MANAGEMENT SYSTEM applying Functions and Files
2 def view_function(): #VIEW function by RAGHAV
3     print('DEVICE LIST CONTENTS ARE:')
4     with open("deviceList.txt", 'r') as f:
5         lines = f.read()
6         print(lines)
7     #ADD function by VENGIE
8 def write_to_file(text):
    main()
```

```
Run: DMS_final_V5
=====
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): 1
=====VIEWING ALL DEVICES FROM THE FILE=====
DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PRO
TOSHIBA T33
APPLE MAC6
DELL D123
Do you want to execute again?(y/n): y
```

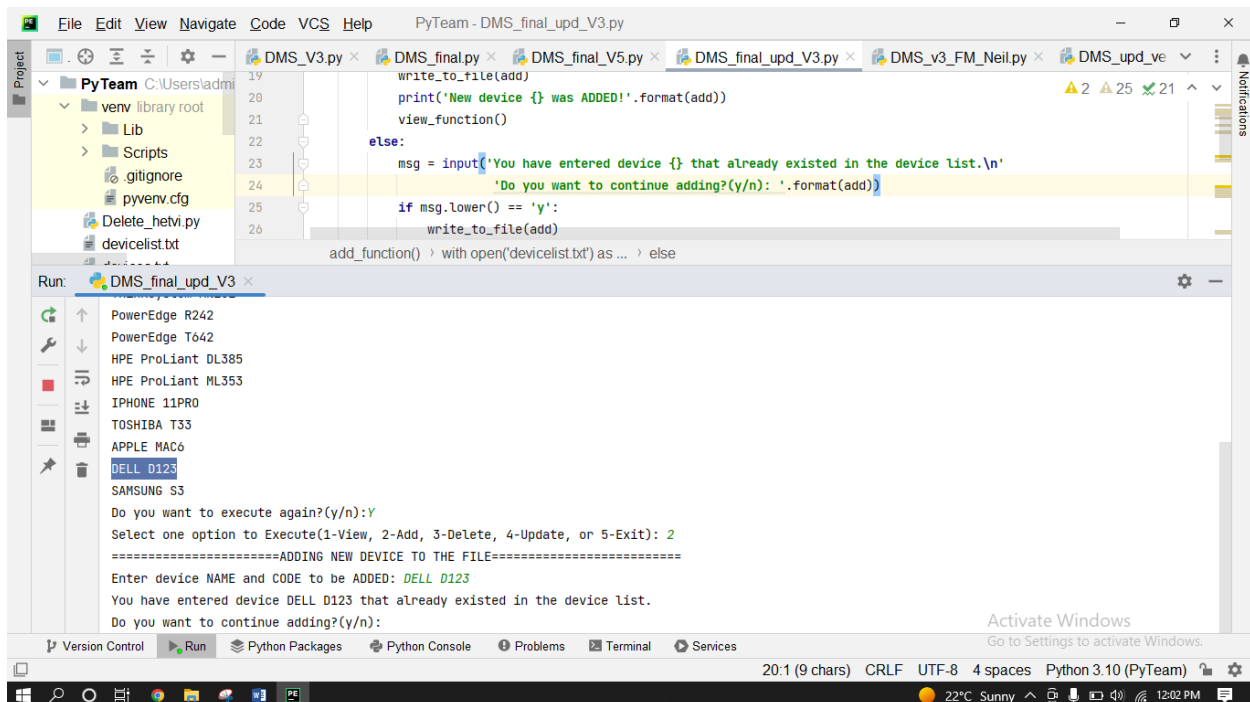
SAMPLE OUTPUT 3: ADDING A NEW DEVICE



```

File Edit View Navigate Code VCS Help PyTeam - DMS_final_V5.py
DMS_V3.py DMS_final.py DMS_final_V5.py DMS_v3_FM_Neil.py DMS_upd_vengie.py DMS.py
Project
  PyTeam C:\Users\admi
  venv library root
  Lib
  Scripts
  gitignore
  pyenv.cfg
Run: DMS_final_V5
Do you want to execute again?(y/n):y
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): 2
=====ADDING NEW DEVICE TO THE FILE=====
Enter device NAME and CODE to be ADDED: SAMSUNG S3
New device SAMSUNG S3 was ADDED!
DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PRO
TOSHIBA T33
APPLE MAC6
DELL D123
SAMSUNG S3
Do you want to execute again?(y/n):
  
```

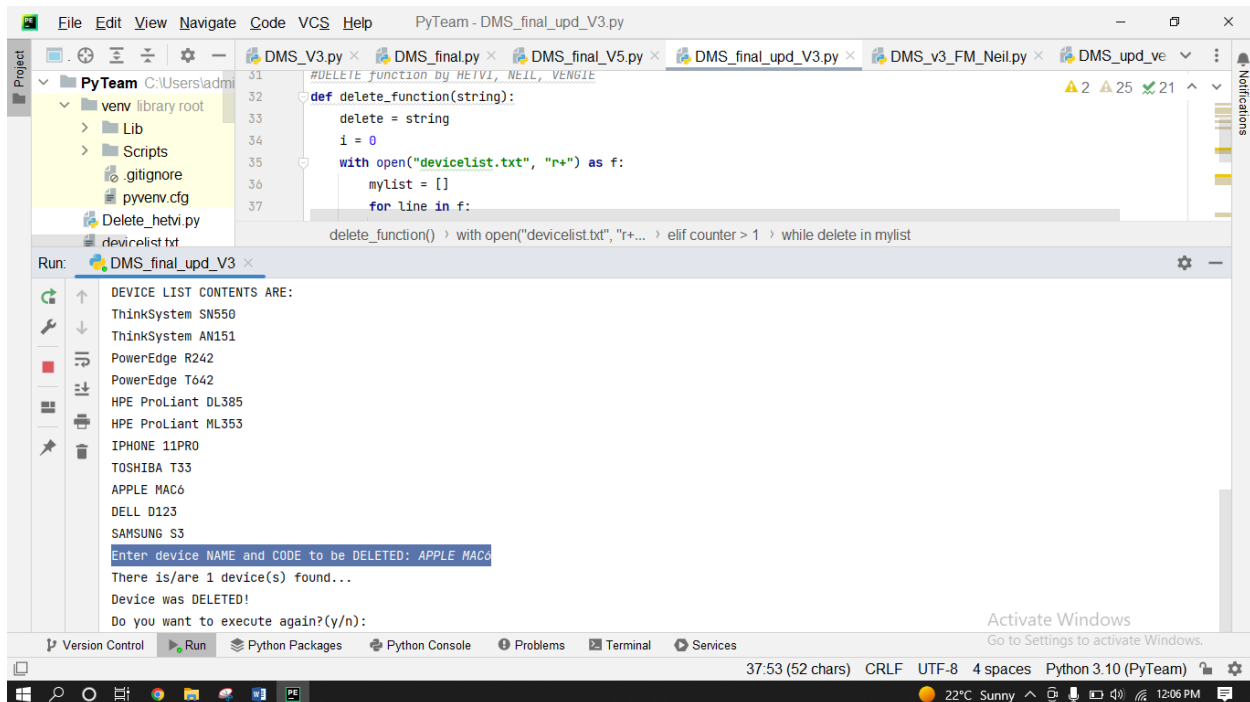
SAMPLE OUTPUT 4: ADDING AN EXISTING DEVICE



```

File Edit View Navigate Code VCS Help PyTeam - DMS_final_upd_V3.py
DMS_V3.py DMS_final.py DMS_final_V5.py DMS_final_upd_V3.py DMS_v3_FM_Neil.py DMS_upd_ve
Project
  PyTeam C:\Users\admi
  venv library root
  Lib
  Scripts
  gitignore
  pyenv.cfg
  Delete_hetvi.py
  devicelist.txt
Run: DMS_final_upd_V3
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PRO
TOSHIBA T33
APPLE MAC6
DELL D123
SAMSUNG S3
Do you want to execute again?(y/n):Y
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): 2
=====ADDING NEW DEVICE TO THE FILE=====
Enter device NAME and CODE to be ADDED: DELL D123
You have entered device DELL D123 that already existed in the device list.
Do you want to continue adding?(y/n):
  
```

SAMPLE OUTPUT 5: DELETING AN EXISTING DEVICE

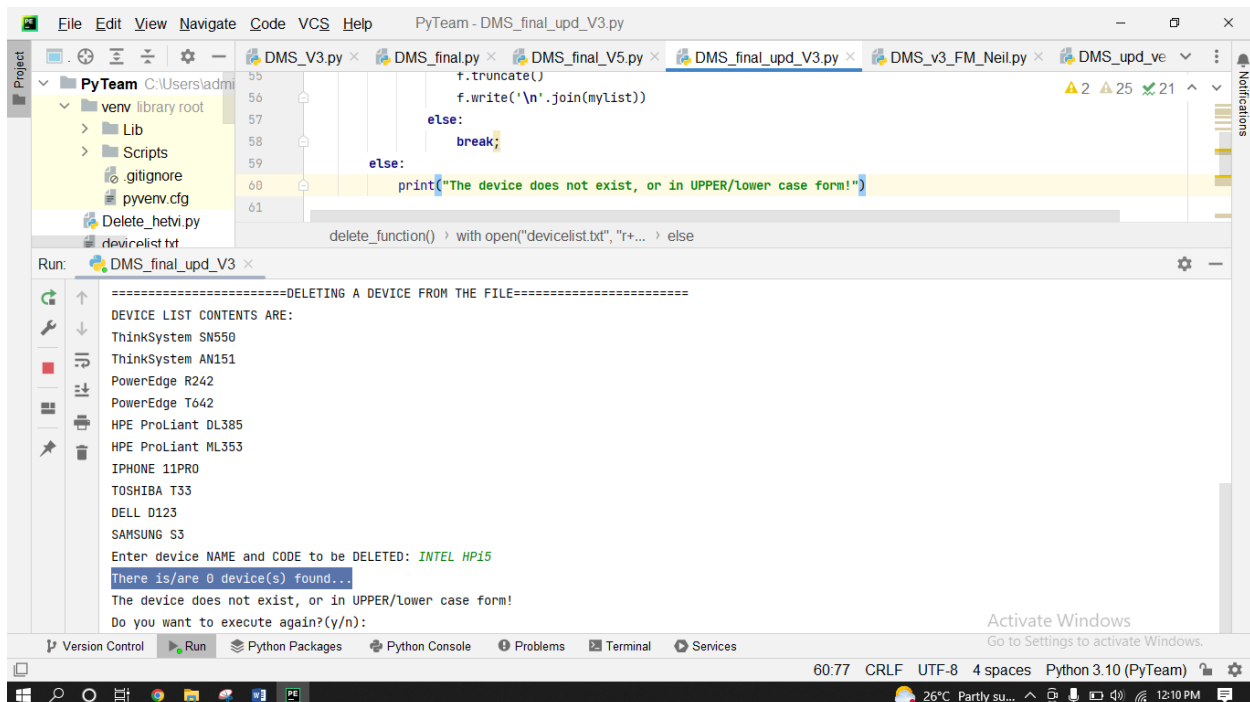


The screenshot shows the PyCharm IDE with the file `DMS_final_upd_V3.py` open. The code defines a `delete_function` that reads from `deviceList.txt`, finds a device to delete, and updates the list. The Run window shows the output of the script:

```

DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PRO
TOSHIBA T33
APPLE MAC6
DELL D123
SAMSUNG S3
Enter device NAME and CODE to be DELETED: APPLE MAC6
There is/are 1 device(s) found...
Device was DELETED!
Do you want to execute again?(y/n):
  
```

SAMPLE OUTPUT 6: DELETING A NON-EXISTING DEVICE

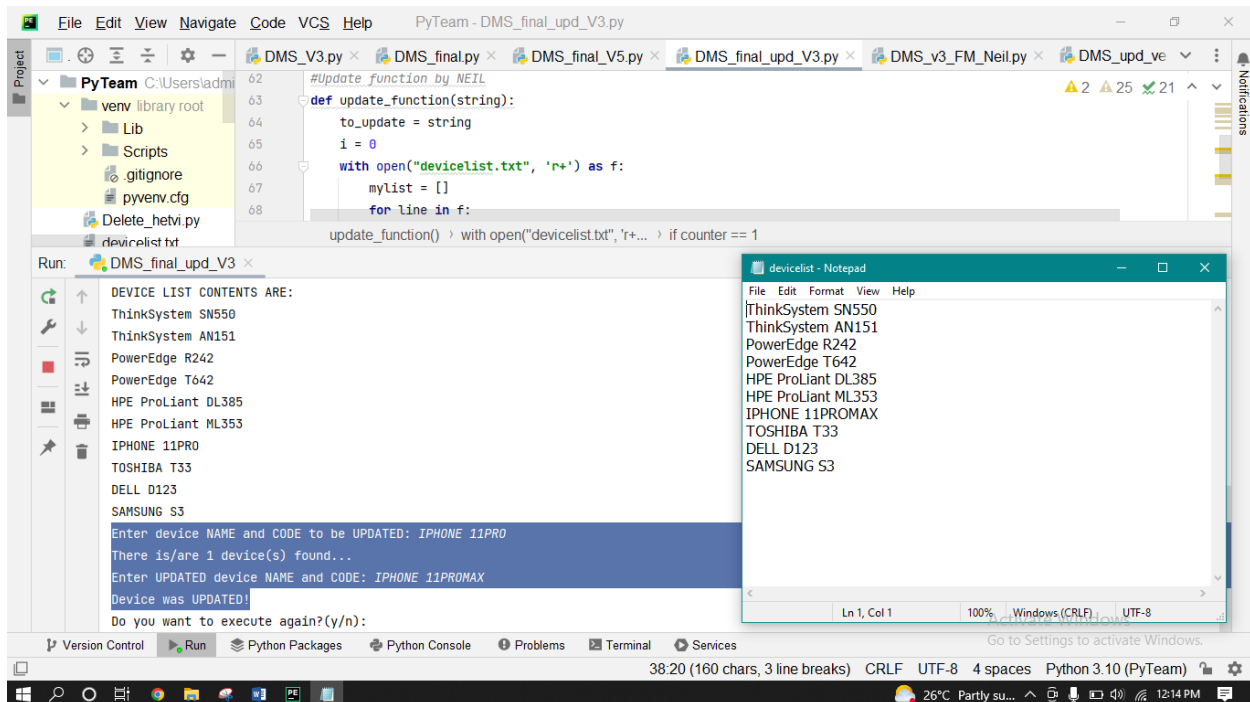


The screenshot shows the PyCharm IDE with the file `DMS_final_upd_V3.py` open. The code defines a `delete_function` that reads from `deviceList.txt`, finds a device to delete, and updates the list. The Run window shows the output of the script:

```

=====DELETING A DEVICE FROM THE FILE=====
DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PRO
TOSHIBA T33
DELL D123
SAMSUNG S3
Enter device NAME and CODE to be DELETED: INTEL HP15
There is/are 0 device(s) found...
The device does not exist, or in UPPER/lower case form!
Do you want to execute again?(y/n):
  
```

SAMPLE OUTPUT 7: UPDATING AN EXISTING DEVICE

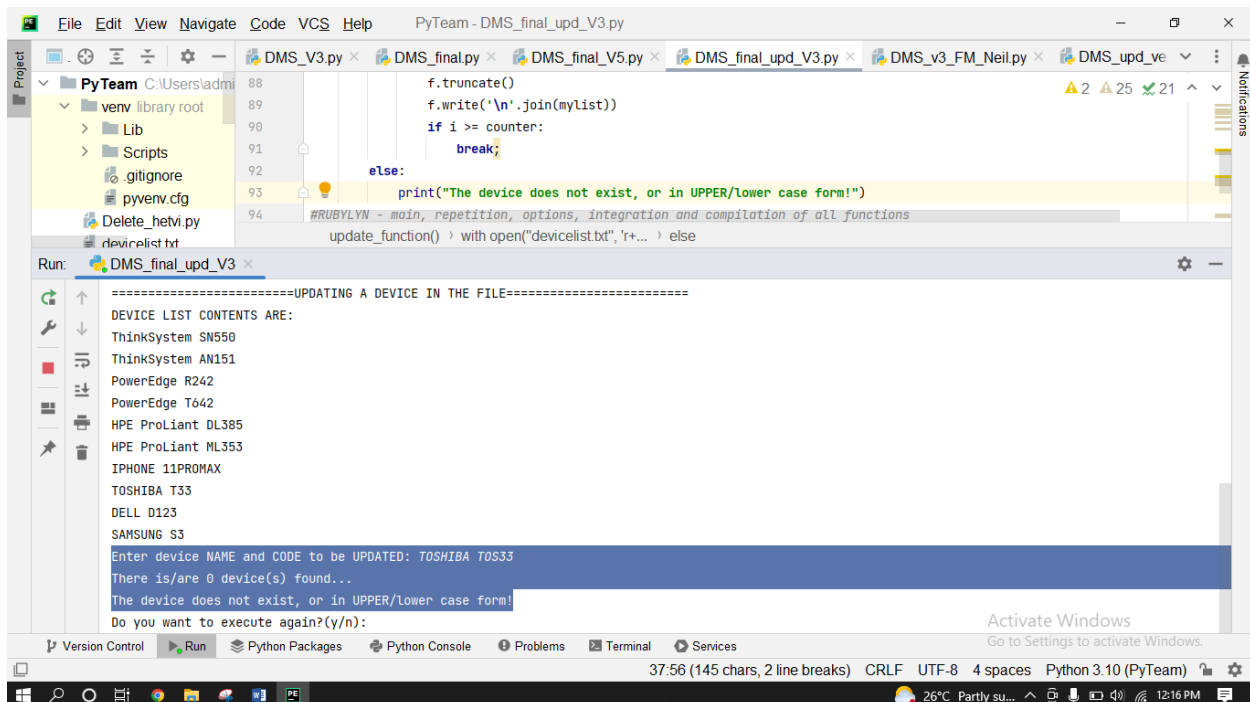


The screenshot shows the PyCharm IDE with the file `DMS_final_upd_V3.py` open. The code defines an `update_function(string)` that reads a list of devices from `deviceList.txt`, finds the device to be updated, and updates its name and code. The Run window shows the output of the script, which successfully updates the device `IPHONE 11PRO` to `IPHONE 11PROMAX`. A Notepad window shows the contents of `deviceList.txt` after the update.

```
#Update function by NEIL
def update_function(string):
    to_update = string
    i = 0
    with open("deviceList.txt", 'r+') as f:
        myList = []
        for line in f:
            update_function() > with open("deviceList.txt", 'r+') as f: if counter == 1

DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PRO
TOSHIBA T33
DELL D123
SAMSUNG S3
Enter device NAME and CODE to be UPDATED: IPHONE 11PROMAX
There is/are 1 device(s) found...
Enter UPDATED device NAME and CODE: IPHONE 11PROMAX
Device was UPDATED!
Do you want to execute again?(y/n):
```

SAMPLE OUTPUT 8: UPDATING A NON-EXISTING DEVICE

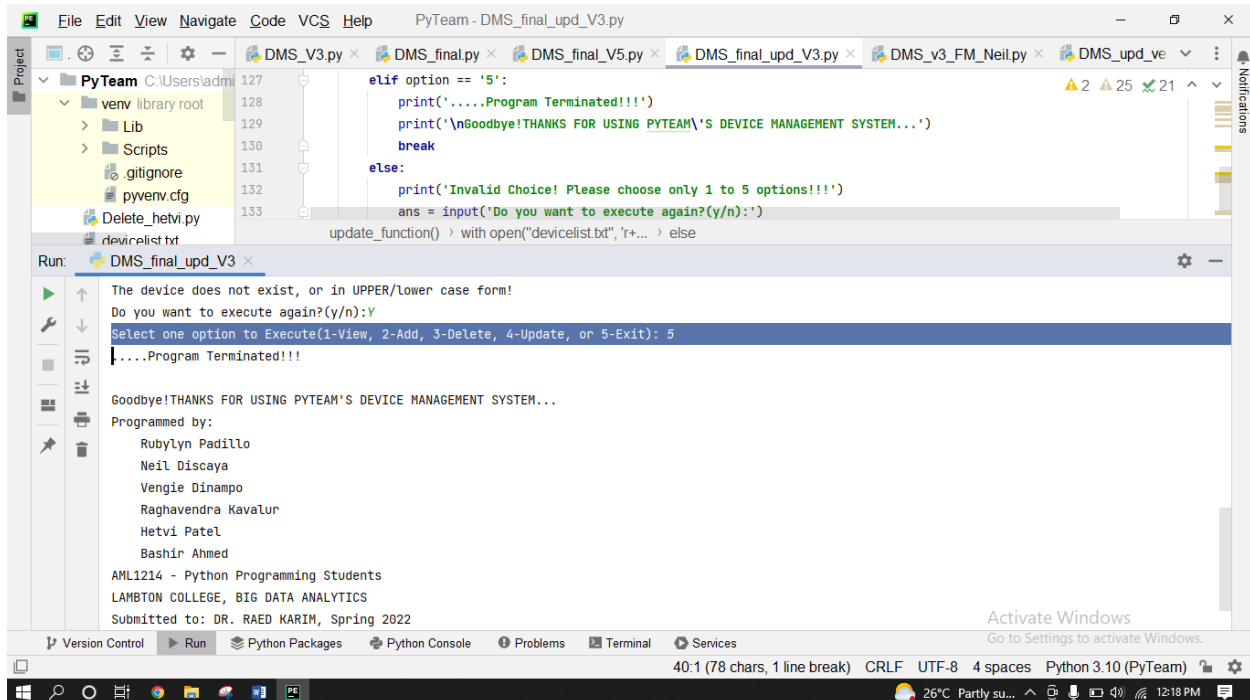


The screenshot shows the PyCharm IDE with the file `DMS_final_upd_V3.py` open. The code defines an `update_function(string)` that reads a list of devices from `deviceList.txt`, finds the device to be updated, and updates its name and code. The Run window shows the output of the script, which successfully updates the device `TOSHIBA TOS33` to `TOSHIBA T33`. A Notepad window shows the contents of `deviceList.txt` after the update.

```
f.truncate()
f.write('\n'.join(myList))
if i >= counter:
    break
else:
    print("The device does not exist, or in UPPER/lower case form!")
#RUBYLYN - main, repetition, options, integration and compilation of all functions
update_function() > with open("deviceList.txt", 'r+') as f: if counter == 1

=====UPDATING A DEVICE IN THE FILE=====
DEVICE LIST CONTENTS ARE:
ThinkSystem SN550
ThinkSystem AN151
PowerEdge R242
PowerEdge T642
HPE ProLiant DL385
HPE ProLiant ML353
IPHONE 11PROMAX
TOSHIBA T33
DELL D123
SAMSUNG S3
Enter device NAME and CODE to be UPDATED: TOSHIBA TOS33
There is/are 0 device(s) found...
The device does not exist, or in UPPER/lower case form!
Do you want to execute again?(y/n):
```

SAMPLE OUTPUT 9: EXIT OPTION, PROGRAM TERMINATION

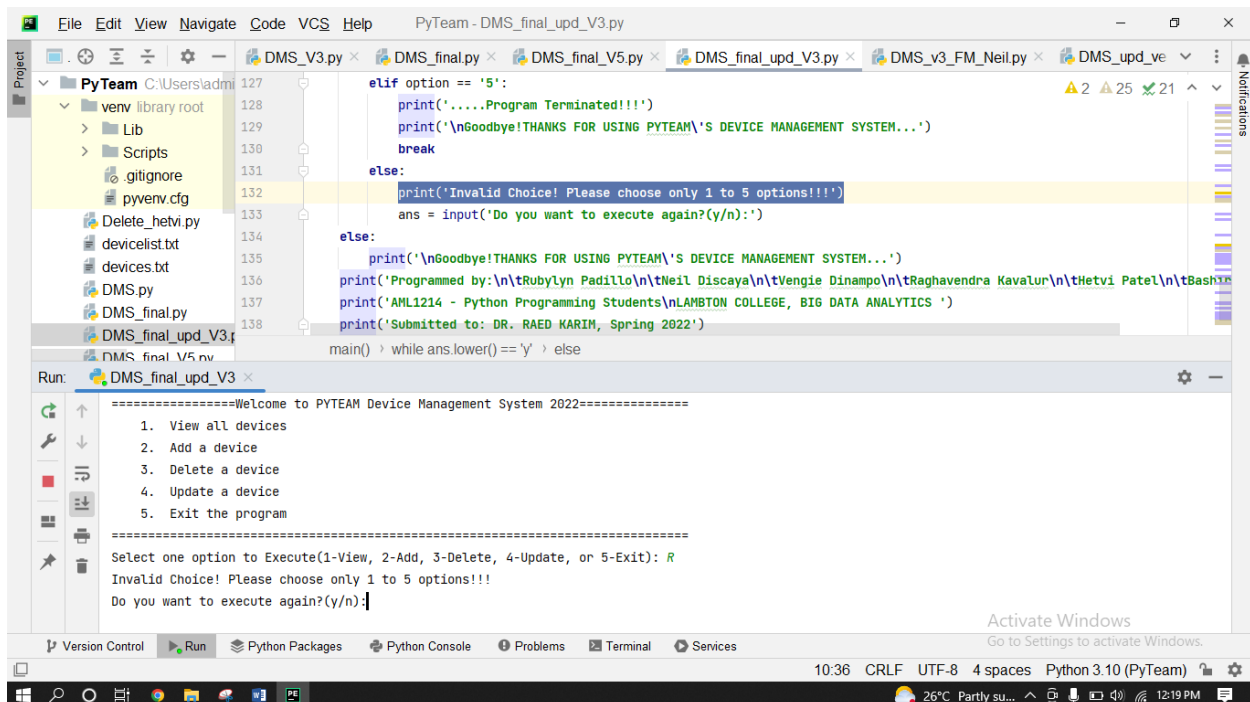


```

File Edit View Navigate Code VCS Help PyTeam - DMS_final_upd_V3.py
DMS_V3.py x DMS_final.py x DMS_final_V5.py x DMS_final_upd_V3.py x DMS_v3_FM_Neil.py x DMS_upd_ve
Project
PyTeam C:\Users\admi
venv library root
Lib
Scripts
.gitignore
pyenv.cfg
Delete_hetvi.py
devicelist.txt
Run: DMS_final_upd_V3 x
The device does not exist, or in UPPER/Lower case form!
Do you want to execute again?(y/n):Y
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): 5
|....Program Terminated!!!
Goodbye!THANKS FOR USING PYTEAM'S DEVICE MANAGEMENT SYSTEM...
Programmed by:
Rubylyn Padillo
Neil Discaya
Vengie Dinampo
Raghavendra Kavalur
Hetvi Patel
Bashir Ahmed
AML1214 - Python Programming Students
LAMBTON COLLEGE, BIG DATA ANALYTICS
Submitted to: DR. RAED KARIM, Spring 2022
Activate Windows
Go to Settings to activate Windows.
40:1 (78 chars, 1 line break) CRLF UTF-8 4 spaces Python 3.10 (PyTeam)
26°C Partly su... 12:18 PM

```

SAMPLE OUTPUT 10: VALIDATION IF CHOSEN OPTION IS NOT 1, 2, 3, 4 OR 5

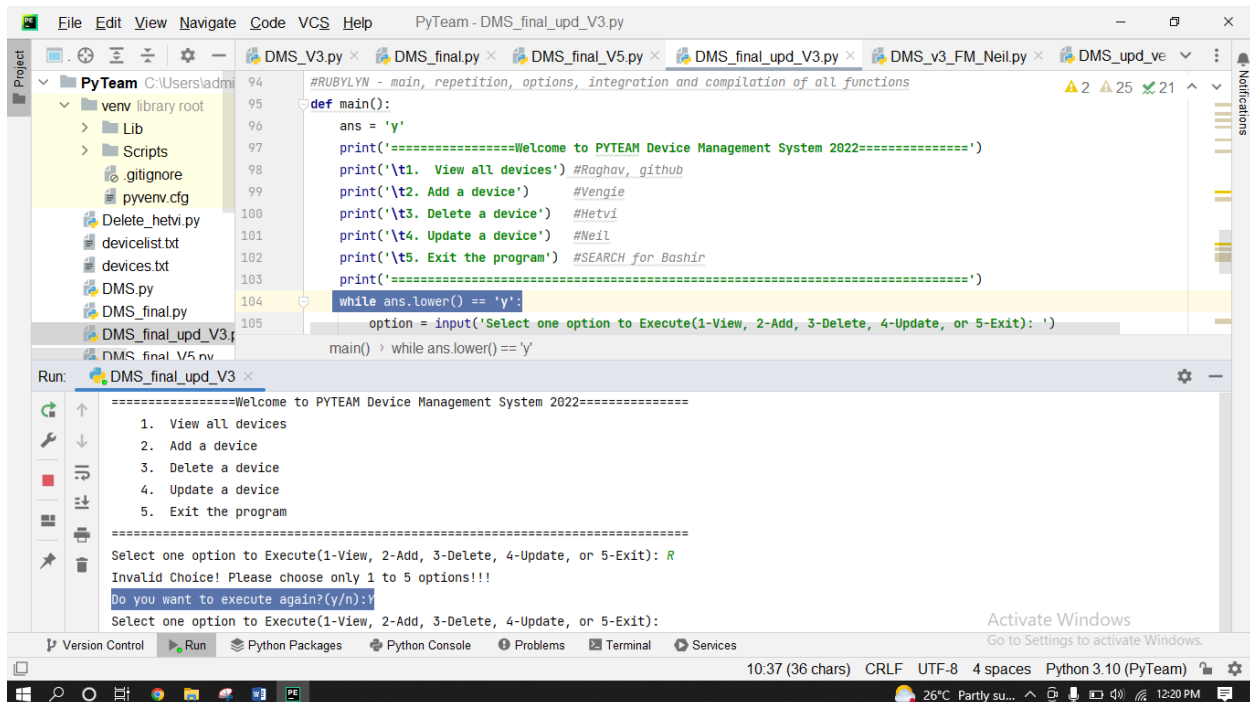


```

File Edit View Navigate Code VCS Help PyTeam - DMS_final_upd_V3.py
DMS_V3.py x DMS_final.py x DMS_final_V5.py x DMS_final_upd_V3.py x DMS_v3_FM_Neil.py x DMS_upd_ve
Project
PyTeam C:\Users\admi
venv library root
Lib
Scripts
.gitignore
pyenv.cfg
Delete_hetvi.py
devicelist.txt
devices.txt
DMS.py
DMS_final.py
DMS_final_upd_V3.py
DMS_final_V5.py
Run: DMS_final_upd_V3 x
=====Welcome to PYTEAM Device Management System 2022=====
1. View all devices
2. Add a device
3. Delete a device
4. Update a device
5. Exit the program
=====
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): 6
Invalid Choice! Please choose only 1 to 5 options!!!
Do you want to execute again?(y/n):|
Activate Windows
Go to Settings to activate Windows.
10:36 CRLF UTF-8 4 spaces Python 3.10 (PyTeam)
26°C Partly su... 12:19 PM

```

SAMPLE OUTPUT 11: REPETITION OF PROGRAM WHILE ANSWER IS Y OR y



```

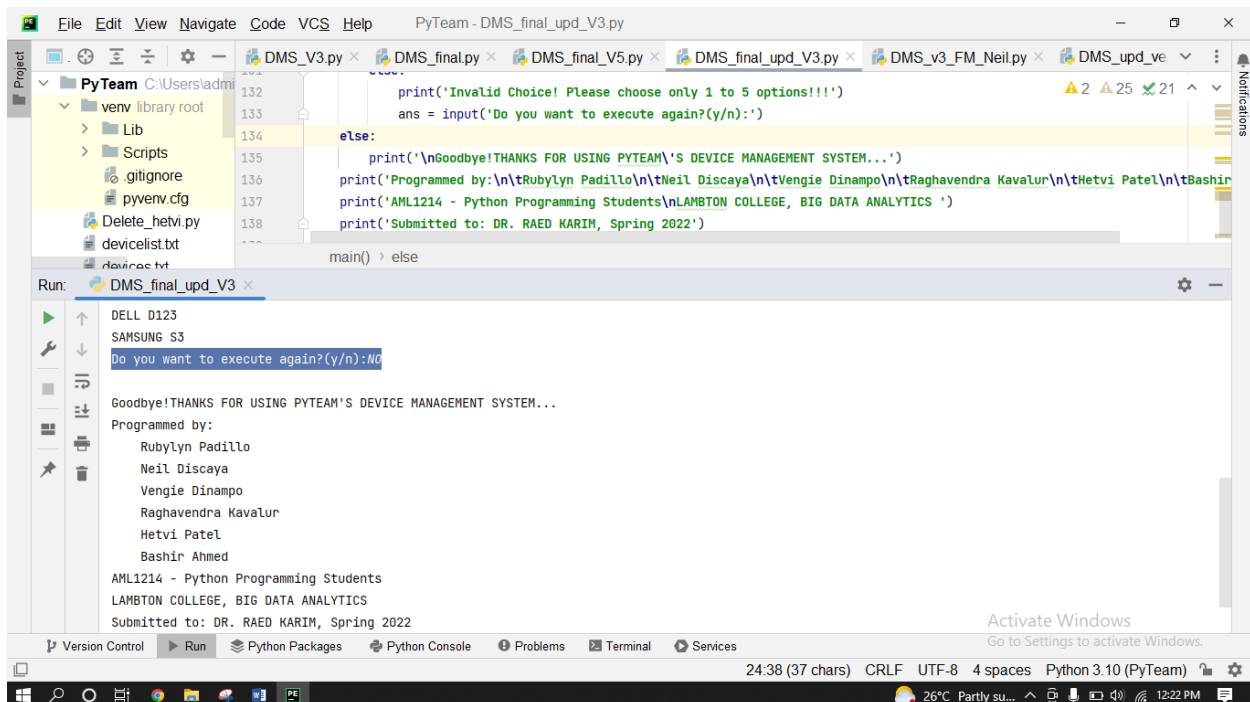
#RUBYLYN - main, repetition, options, integration and compilation of all functions
def main():
    ans = 'y'
    print('=====Welcome to PYTEAM Device Management System 2022=====')
    print('\t1. View all devices') #Raghav, github
    print('\t2. Add a device') #Vengie
    print('\t3. Delete a device') #Hetvi
    print('\t4. Update a device') #Neil
    print('\t5. Exit the program') #SEARCH for Bashir
    print('=====')
    while ans.lower() == 'y':
        option = input('Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): ')
        main()
  
```

Run: DMS_final_upd_V3

```

=====Welcome to PYTEAM Device Management System 2022=====
1. View all devices
2. Add a device
3. Delete a device
4. Update a device
5. Exit the program
=====
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit): R
Invalid Choice! Please choose only 1 to 5 options!!!
Do you want to execute again?(y/n):Y
Select one option to Execute(1-View, 2-Add, 3-Delete, 4-Update, or 5-Exit):
  
```

SAMPLE OUTPUT 12: PROGRAM WILL TERMINATE IF ANSWER IS NOT Y OR y



```

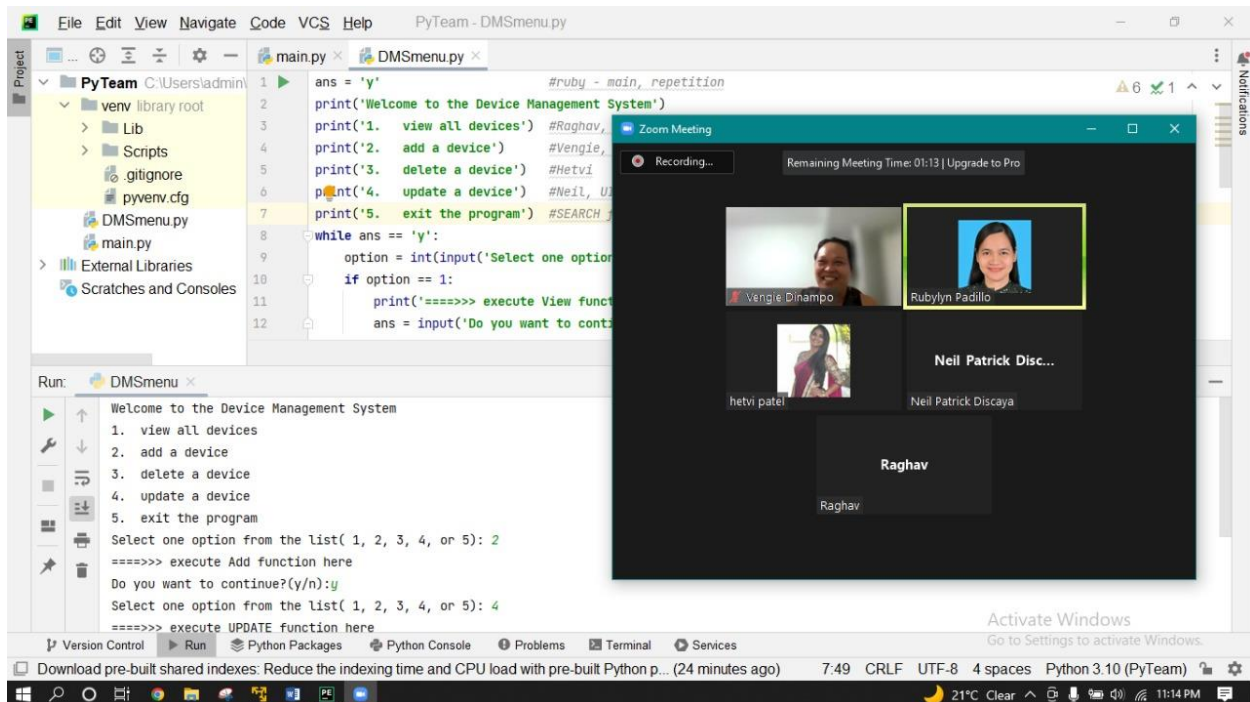
print('Invalid Choice! Please choose only 1 to 5 options!!!')
ans = input('Do you want to execute again?(y/n):')
else:
    print('\nGoodbye! THANKS FOR USING PYTEAM'S DEVICE MANAGEMENT SYSTEM...')
    print('Programmed by:\nRubylyn Padillo\nNeil Discaya\nVengie Dinampo\nRaghavendra Kavalur\nHetvi Patel\nBashir')
    print('AML1214 - Python Programming Students\nLAMBTON COLLEGE, BIG DATA ANALYTICS ')
    print('Submitted to: DR. RAED KARIM, Spring 2022')
  
```

Run: DMS_final_upd_V3

```

DELL D123
SAMSUNG S3
Do you want to execute again?(y/n):N
Goodbye! THANKS FOR USING PYTEAM'S DEVICE MANAGEMENT SYSTEM...
Programmed by:
Rubylyn Padillo
Neil Discaya
Vengie Dinampo
Raghavendra Kavalur
Hetvi Patel
Bashir Ahmed
AML1214 - Python Programming Students
LAMBTON COLLEGE, BIG DATA ANALYTICS
Submitted to: DR. RAED KARIM, Spring 2022
  
```


PICTURE 1: PYTEAM GROUP MEETING (BRAINSTORMING)



PICTURES 2-3: PYTEAM GROUP COMMUNICATION (WHATSAPP/GITHUB)

