**AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH**

***PROJECT****: COMPUTER LAB MANAGEMENT SYSTEM*

**Course**: Programming with Python

**Section**: [B]

**Date of submission:** 01/03/2023

**Faulty:** Dr. Akinul Islam Jony

**Semester:** Spring 2022-2023

**Submitted by:**

**Name**: Tanvir Chowdhury

**ID**: 20-42699-1

**Project Overview:**

This is a computer lab management system coded in python. It is a console-based application that records and keep track of the computer information in a lab. We can register a PC into the system by inserting information such as PC number, PC operating system, PC status. Once data are inserted in the system, we can also update the existing PC information. We may also remove any PC information form the system. The user can also choose to save all the Computer details in a text file to save it on the Hard drive for later use.

This system has multiple editing functionality and user-friendly menu.

**Project Solution Design:**

START

Loop:

* Show Main Menu Options
* Take user input to select an option
* Press 1 = Call add\_pc() function
* Press 2 = Call search\_pc function
* Press 3 = Call update\_pc() function
* Press 4= Call remove\_pc() function
* Press 5= Call show\_all\_pc() function
* Press 6= Call save\_file() function
* Press 7= Call quit() function

END

**IMPLEMENTATION:**

**class\_module.py :**

class PC:

all\_pc = [] # stores all instances of the class

def \_\_init\_\_(self, pc\_num, pc\_os, pc\_status):

self.number = pc\_num

self.os = pc\_os

self.status = pc\_status

PC.all\_pc.append(self)

print(f"\n New PC Registered\n") #Adds to the list

def add\_pc(self): #Adding New Computer

print("Enter new PC details:-\n")

pc\_num = input("PC number: ")

pc\_os = input("PC operating system: ")

pc\_status = input("PC status: ")

pc\_exists = PC.check\_pc(pc\_num) # check for duplicates

if pc\_exists == 0:

new\_pc = PC(pc\_num, pc\_os, pc\_status) #Adding PC object to the list object

else:

print("\n PC with same number already exists\n")

print("To update pc number - press '1'")

print("TO remove pc - press '2'")

print("TO cancel - press '3'")

i = input("\n Waiting for input: ")

if i == '1':

new\_pc\_num = input("\n New pc number: ")

PC.update\_pc(pc\_exists, new\_pc\_num, pc\_exists.os,

pc\_exists.status)

new\_pc = PC(pc\_num, pc\_os, pc\_status)

elif i == '2':

PC.remove\_pc(pc\_exists)

new\_pc = PC(pc\_num, pc\_os, pc\_stat)

elif i == '3':

PC.Main()

else:

PC.Main()

def remove\_pc(computer): #Delete PC

print(f"\n {computer.number} is deleted")

PC.all\_pc.remove(computer)

def update\_pc(computer, new\_pc\_num, new\_pc\_os, new\_pc\_status): #Update Information

if PC.check\_pc(new\_pc\_num) == 0:

old\_pc\_num = computer.number

computer.number = new\_pc\_num

computer.os = new\_pc\_os

computer.status = new\_pc\_status

print(f"PC {old\_pc\_num} has been updated\n")

else:

print(f"\n The number {computer.number} is already registered\n")

def check\_pc(pc\_num): #Checking whether Pc number is unique

flag = 1

for computer in PC.all\_pc:

if computer.number == pc\_num:

flag = 0

break

if flag == 0:

return computer

else:

return 0

def show\_all\_pc(): #Display all computer information

if len(PC.all\_pc) != 0:

print("All PC details:-\n")

for computer in PC.all\_pc:

print(f"PC number: {computer.number}")

print(f"PC Operating System: {computer.os}")

print(f"PC Status: {computer.status}\n")

else:

print("No PC available")

def search\_pc(pc\_num): # Search for PC number

flag = 0

for computer in PC.all\_pc:

if computer.number == pc\_num:

flag = 1

print("Search results:-\n")

print(f"PC number: {computer.number}")

print(f"PC operating System: {computer.os}")

print(f"PC status: {computer.status}")

pc\_exists = computer

break

if flag == 1:

return pc\_exists

else:

print(f"No PC found with PC number {pc\_num}")

return 0

def save\_file(): #Saves the Data in the Directory

try:

file = "ALL\_PC.txt"

with open(file, 'w') as file\_obj:

for computer in PC.all\_pc:

file\_obj.write(f"{computer.number},{computer.os},{computer.status}\n")

print("\nAll PC information has been saved in the file")

except Exception as e :

print("\n File saving error!")

**main.py :**

from class\_module import PC # Import Module

while 1>0:

print("COMPUTER LAB MANAGEMENT SYSTEM \n")

print("------------------------------ \n")

print("Here is the main menu\n")

print("1: Add a new PC")

print("2: Search PC")

print("3: Update PC")

print("4: Remove PC")

print("5: Show all PC info")

print("6: Save all PC info")

print("7: EXIT!\n")

choice = input("Waiting for Input: \n")

if choice == '1':

PC.add\_pc(PC)

elif choice == '2':

n = input("Enter PC number to search: ")

PC.search\_pc(n)

elif choice == '3':

PC.show\_all\_pc()

n = input("Enter PC number to be updated: ")

pc\_exists = PC.search\_pc(n)

if pc\_exists != 0:

print(f"\nFor {n} selected PC")

new\_pc\_num = input("\nEnter new PC number: ")

new\_pc\_os = input("\Enter new PC'S operating system: ")

new\_pc\_stat = input("\Enter new PC'S status: ")

PC.update\_pc(pc\_exists, new\_pc\_num, new\_pc\_os, new\_pc\_stat)

elif choice == '4':

pc\_num = input("\nEnter PC number to remove: ")

pc\_exists = PC.search\_pc(pc\_num)

if pc\_exists != 0:

PC.remove\_pc(pc\_exists)

elif choice == '5':

PC.show\_all\_pc()

elif choice == '6':

PC.save\_file()

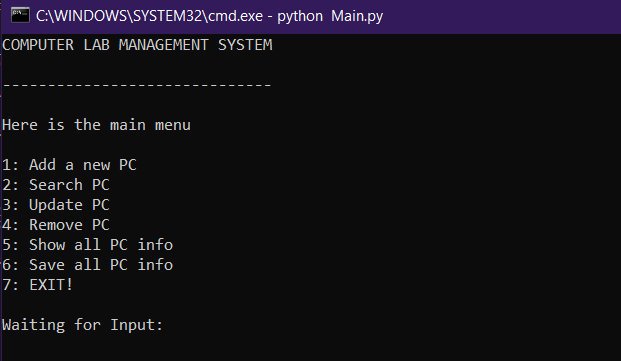
elif choice == '7':

quit()

**APPLICATION OVERVIEW:**

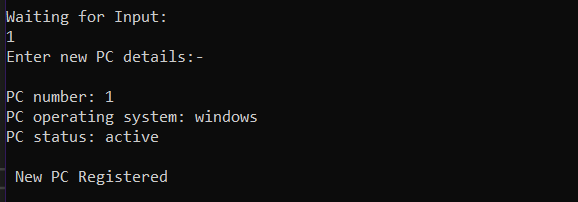
***Main menu:***

This is the startup menu which shows the executable features of the application

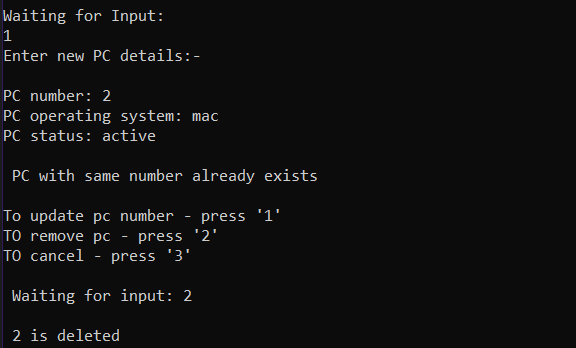


1. **Adding a new PC:**

The user can add a new PC into the system by inserting information such as, PC number, PC Operating system and PC status

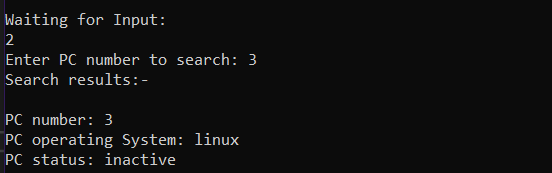


If the user wants to add a PC with already existing PC number. The program will prompt with options to update that PC number or remove that PC from the system



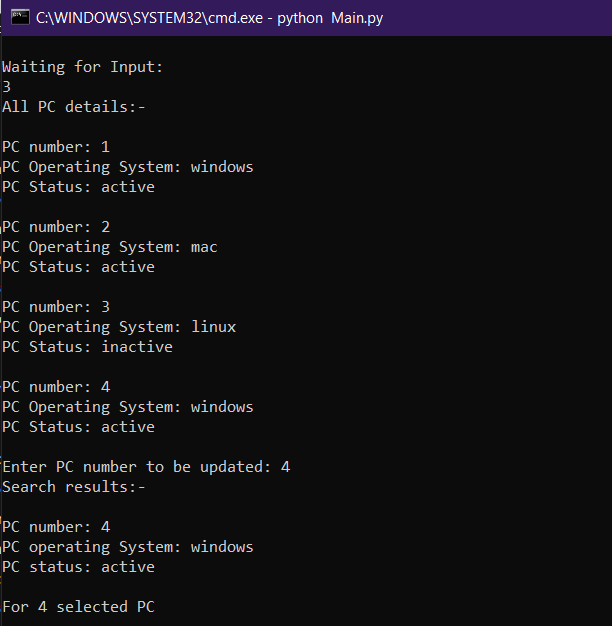
1. **Search for PC:**

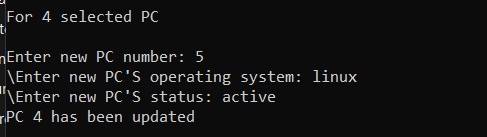
As the details of all computers are kept in a list as objects, user can search for specific PC by the PC number. The system implements linear search to look for the requested PC



1. **Updating PC details**

User can update the existing PC details in the system, the prompt will show all available PC and can choose the which one to update by the PC number.User can update and save the details of that PC

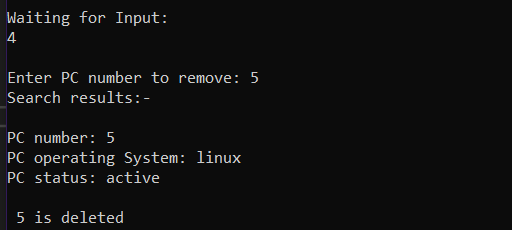




The selected PC gets updated.

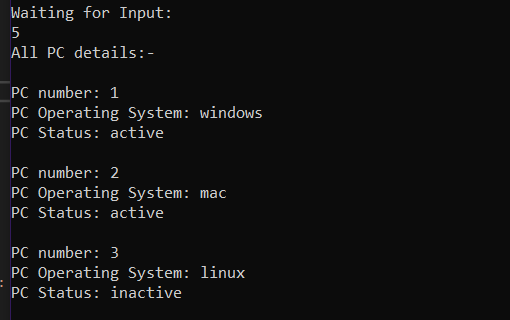
1. **Remove PC**

The user call also remove existing PC from the system by the Remove functionality



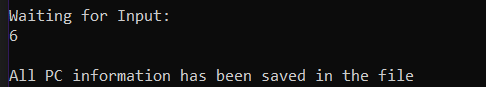
1. **Display all PC details:**

This function shows the detailed list of all the PC registered in the system

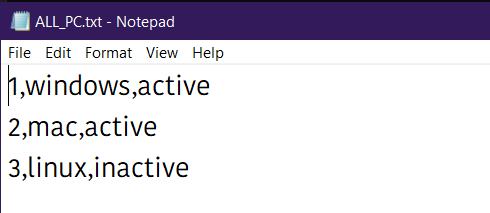


1. **Save all PC details in a text file**

This function saves the data on the hard drive in a text file : ALL\_PC.txt



The file is stored in the Project Directory



1. **Exit:**

This option stop the execution of the program

