```
Import numpy as np
Import datetime
Import calendar
Import pandas as pd

def login_screen(teacherDB):
    print("Auto Adjustment For Faculty")
    print("Teacher 1: 1021 \n Teacher 2: 1022 \n Teacher 3: 1023 \n Teacher 4: 1024 \n")
    teacherID = int(input("please enter your Teacher ID >> "))
    if(teacherID in teacherDB.keys()):
        teacherID = teacherDB[teacherID]
        welcome_screen(teacherID)
    else:
        print("your record doesnot exist")

def show_free_slots(teacherID, day):
    for time_Slot in range(5):
        if (profTT[teacherID][day][time_Slot] == 0):
            print(trans_dict_time[time_Slot] , " is free")

def show_taken_slots(teacherID, day):
    for time_Slot in range(5):
        if (profTT[teacherID][day][time_Slot] == 1):
            print(trans_dict_time[time_Slot] , " is occupied")

def show_makeup_slots(teacherID, day):
    for time_Slot in range(5):
        if (profTT[teacherID][day][time_Slot] == 2):
            print(trans_dict_time[time_Slot] , " is make up adjustment")


def welcome_screen(teacherID):
    print("\n\nFaculty Requirment Portal! \n")
    print("please select one option from the list.")
    print("Select 1>    to veiw schedule ")
    print("Select 2>    to appoint a make-up class ")
    print("Select 3>    to apply emergency leave ")
    print("Select 4>    to exit this portal ")

    choice = int(input("option > "))
    if(choice == 1):
        day = input("enter the day >> ")
        print("\nYOUR SCHEDULE \n")
```

```
            day = trans_dict_day[day]
            show_taken_slots(teacherID, day)
            show_free_slots(teacherID, day)
            show_makeup_slots(teacherID, day)
            choice = input("would you like to return to home page? >> ").lower()
            if(choice == "yes"):
                print("redirecting you to home page \n\n")
                welcome_screen(teacherID)

        elif(choice == 2):
            day = input("please enter the day you wish to apply for a make up >> ")
            day = trans_dict_day[day]
            print("please choose time slot for the leave")
            print("10:00AM - 11:00AM >> 0")
            print("11:00AM - 12:0PAM >> 1")
            print("1:00PM - 2:00PM >> 2")
            print("2:00PM - 3:00PM >> 3")
            print("3:00PM - 4:00PM >> 4")
            time_slot = int(input("option >> "))
            MakeUpApp(0,teacherID,day,time_slot)

        elif(choice == 3):
            day = input("please enter the day you wish to apply for a leave >> ")
            day = trans_dict_day[day]
            print("please choose time slot for the leave")
            print("10:00AM - 11:00AM >> 0")
            print("11:00AM - 12:0PAM >> 1")
            print("1:00PM - 2:00PM >> 2")
            print("2:00PM - 3:00PM >> 3")
            print("3:00PM - 4:00PM >> 4")
            time_slot = int(input("option >> "))
            leaveApp(teacherID,day,time_slot)


        elif(choice == 4):
            print("\nexiting!")


    def leaveApp(teacherID, day, time_slot):
        print("\nupdating time schedule")

        profTT[teacherID][day][time_slot] = 0
        MakeUpApp(1,teacherID,day,time_slot)
```

```
     choice = input("would you like to return to home page? >> ").lower()
     if(choice == "yes"):
         print("redirecting you to home page \n\n")
         welcome_screen(teacherID)


def MakeUpApp(flag,teacherID,day,time_slot):
    if(flag == 0):
        profTT[teacherID][day][time_slot] = 2
        print("\nsucessfully updated\n")
        choice = input("would you like to return to home page? >> ").lower()
        if(choice == "yes"):
            print("redirecting you to home page \n\n")
            welcome_screen(teacherID)
    else:
        ResProfChoices = []
        for ResProf in range(len(resprofTT)):
            if(resprofTT[ResProf][day][time_slot] == 0):
                ResProfChoices.append(ResProf)
        if(ResProfChoices.count == 1):
            resprofTT[ResProfChoices[0]][day][time_slot] = 2
            print("\n\nassigning reserve professor",ResProfChoices[0]+1,"for the time slot\n\n")
        else:
            #print(ResProfChoices)
            bestChoice = GetBestChoice(ResProfChoices,day,time_slot)
            print("\n\nassigning reserve professor",bestChoice+1,"for the time slot\n\n")

def GetBestChoice(choices,day,time_slot):
    priorityTable = {}
    for i in choices:
        priorityTable[i] = 0
        if(time_slot > 4 and time_slot < 0):
            if (resprofTT[i][day][time_slot-1] == 1 or resprofTT[i][day][time_slot+1] == 1):
                priorityTable[i] += 1
            elif (resprofTT[i][day][time_slot-1] == 1 and resprofTT[i][day][time_slot+1] == 1):
                priorityTable[i] += 2
    #print(priorityTable)
    for key,value in priorityTable.items():
            if(value == max(priorityTable.values())):
                #print(key)
                return key
defTT =  [[0,0,0,0,0],[0,0,0,0,0],[0,0,0,0,0],[0,0,0,0,0],[0,0,0,0,0]]
```

```
profTT = [[[0,1,0,1,0],[1,1,0,0,1],[1,1,0,1,1],[0,1,1,1,0],[1,1,0,1,1]],
          [[1,0,0,1,1],[0,1,0,1,0],[0,1,1,0,0],[1,0,1,0,0],[1,0,1,0,1]],
          [[0,1,1,0,1],[1,0,1,0,0],[1,0,0,1,0],[1,0,1,0,1],[1,1,1,0,0]],
          [[1,1,1,0,0],[0,0,1,1,1],[0,0,1,0,1],[0,1,0,1,1],[0,0,1,1,1]]]

resprofTT = [[[1,0,0,1,1],[0,1,0,1,1],[1,1,0,0,1],[1,0,0,1,1],[0,1,0,1,1]],
             [[0,1,1,1,0],[1,1,0,0,0],[0,1,0,1,0],[0,1,1,0,0],[1,1,0,0,0]],
             [[1,1,0,0,0],[1,0,1,0,0],[1,0,0,1,1],[1,1,0,1,1],[1,0,1,0,0]],
             [[0,0,1,1,1],[0,0,1,1,1],[1,0,1,0,0],[0,0,1,0,1],[0,0,1,1,1]]]

trans_dict_day = {"monday" :0, "tuesday":1, "wednesday":2, "thursday":3, "friday":4}
trans_dict_day_opp = {0:"monday" ,1:"tuesday", 2:"wednesday", 3:"thursday",4: "friday"}
trans_dict_time= {0:"10:00AM - 11:00AM", 1:"11:00AM - 12:00PM ", 2:"1:00PM - 2:00PM",
3:"2:00PM - 3:00 PM ", 4:"3:00PM - 4:00PM"}
teacherDB = {1:0, 1021 :0, 1022 :1, 1023:2, 1024:3}

login_screen(teacherDB)
```