**Lab 5**

**Python**

**Name:** Etcherla Sai Manoj **Mis. No:** 112015044 **Branch:** CSE

**Question:**

**Code:**

import pickle

class Result:

    def \_\_init\_\_(self, t1, t2, final) -> None:

        self.t1 = t1

        self.t2 = t2

        self.final = final

class Student:

    def \_\_init\_\_(self, id, mis, email, result) -> None:

        self.id = id

        self.mis = mis

        self.email = email

        self.result = result

def update(records, stuId):

    attrs = ['id', 'mis', 'email', 't1', 't2', 'final']

    print('Enter the attribute you wish to change')

    print(f'Choices are {attrs}')

    attr = input('Your choice: ').lower()

    for record in records:

        if record.id == stuId:

            found\_record = record

            break

    else:

        print('The ID is not in the database')

        return

    val = input('Enter the value: ')

    if attr in ('t1', 't2', 'final'):

        setattr(found\_record.result, attr, int(val))

    else:

        setattr(found\_record, attr, val)

    with open('file', 'wb') as f:

        pickle.dump(records, f)

def display(records):

    print(f'\n{"ID":^20}{"MIS":^20}{"Email":^20}{"T1":^20}{"T2":^20}{"final":^20}')

    for record in records:

        print(f'{record.id:^20}{record.mis:^20}{record.email:^20}{record.result.t1:^20}{record.result.t2:^20}{record.result.final:^20}')

if \_\_name\_\_ == '\_\_main\_\_':

    exams = ['t1', 't2', 'final']

    records = []

    for \_ in range(5):

        stuId = input('\nEnter the student ID: ')

        stuMis = input('Enter the student MIS: ')

        email = input('Enter the student email: ')

        d = {}

        for exam in exams:

            marks = int(input('Enter {exam} marks: '))

            d[exam] = (marks)

        result = Result(\*\*d)

        records.append(Student(stuId, stuMis, email, result))

    with open('file', 'wb') as f:

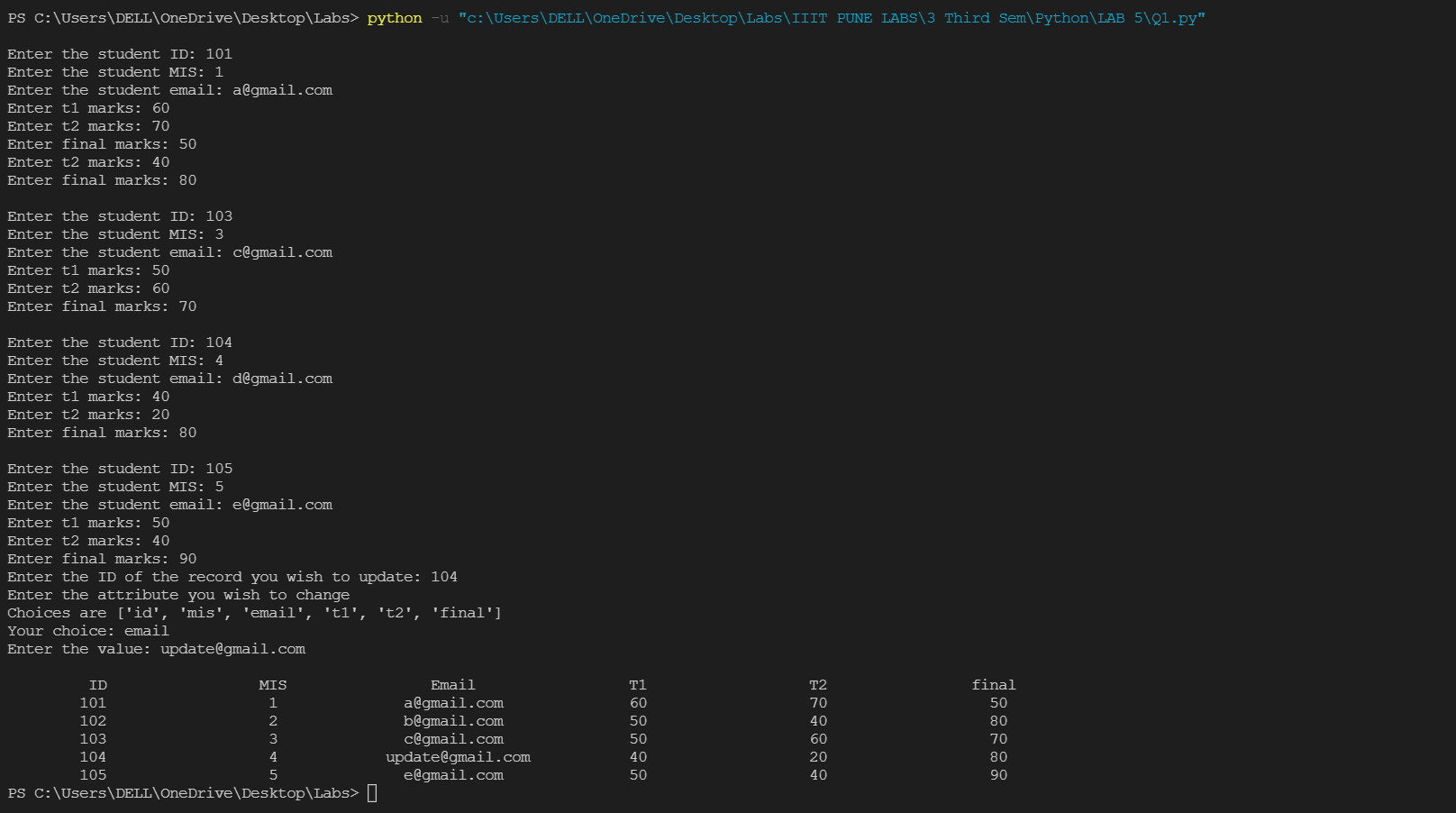
        pickle.dump(records, f)

    stuId = input('Enter the ID of the record you wish to update: ')

    update(records, stuId)

    display(records)

**Output:**

****