Program 10

```
import pickle

def read_records():
    with open("emp.dat", "rb") as f:
        try:
            while True:
                emp = pickle.load(f)
                empcode = emp[0]
                name = emp[1]
                place = emp[2]
                salary = emp[3]

                print(empcode, name,place, salary)
        except EOFError:
            pass

def write_records():
    with open("emp.dat","ab") as f:
        n = int(input("Enter number of employees to be entered: "))
        for i in range(n):
            empcode = int(input("Enter employee code: "))
            name = input("Enter employee name: ")
            place = input("Enter employee place: ")
            salary = int(input("Enter employee salary: "))

            emp = [empcode,name,place,salary]
            pickle.dump(emp,f)

while True:
    ch = input("Enter choice:\n1. Read record\n2. Write record: ")
    if ch == "1":
        read_records()
    elif ch == "2":
        write_records()
    else:
        print("Invalid choice")
```

```
Enter choice:
1. Read record
2. Write record: 2
Enter number of employees to be entered: 2
Enter employee code: 100
Enter employee name: Smitha
Enter employee place: Kannur
Enter employee salary: 40000
Enter employee code: 105
Enter employee name: Rajesh
Enter employee place: Kochi
Enter employee salary: 35000
Enter choice:
1. Read record
2. Write record: 1
100 Smitha Kannur 40000
105 Rajesh Kochi 35000
Enter choice:
1. Read record
2. Write record: 2
Enter number of employees to be entered: 1
Enter employee code: 110
Enter employee name: Rahul
Enter employee place: Payannur
Enter employee salary: 67500
Enter choice:
1. Read record
2. Write record: 1
100 Smitha Kannur 40000
105 Rajesh Kochi 35000
110 Rahul Payannur 67500
Enter choice:
1. Read record
2. Write record:
```

Program 11

```python
import pickle

def read_records():
    with open("emp.dat", "rb") as f:
```

```
        try:
            while True:
                emp = pickle.load(f)
                empcode = emp["ecode"]
                name = emp["ename"]
                place = emp["eplace"]
                salary = emp["esal"]

                print(empcode, name,place, salary)
        except EOFError:
            pass

def write_records():
    with open("emp.dat","ab") as f:
        n = int(input("Enter number of employees to be entered: "))
        for i in range(n):
            emp = {}

            empcode = int(input("Enter employee code: "))
            name = input("Enter employee name: ")
            place = input("Enter employee place: ")
            salary = int(input("Enter employee salary: "))

            emp["ecode"] = empcode
            emp["ename"] = name
            emp["eplace"] = place
            emp["esal"] = salary
            pickle.dump(emp,f)

while True:
    ch = input("Enter choice:\n1. Read record\n2. Write record: ")
    if ch == "1":
        read_records()
    elif ch == "2":
        write_records()
    else:
        print("Invalid choice")
```

```
Enter choice:
1. Read record
2. Write record: 2
Enter number of employees to be entered: 2
Enter employee code: 200
Enter employee name: Alan
Enter employee place: Gujarat
Enter employee salary: 45000
Enter employee code: 201
Enter employee name: David
Enter employee place: New Delhi
Enter employee salary: 56000
Enter choice:
1. Read record
2. Write record: 1
200 Alan Gujarat 34500
201 David New Delhi 56000
202 Ajay Kerala 40000
200 Alan Gujarat 45000
200 Alan Gujarat 45000
201 David New Delhi 56000
Enter choice:
1. Read record
2. Write record: 2
Enter number of employees to be entered: 1
Enter employee code: 202
Enter employee name: Ajay
Enter employee place: Kerala
Enter employee salary: 80000
Enter choice:
1. Read record
2. Write record: 1
200 Alan Gujarat 34500
201 David New Delhi 56000
202 Ajay Kerala 40000
200 Alan Gujarat 45000
200 Alan Gujarat 45000
201 David New Delhi 56000
202 Ajay Kerala 80000
Enter choice:
1. Read record
2. Write record:
```

Program 12

```
import pickle

def search_records():
    with open("emp.dat", "rb") as f:
        found = False
        try:
            ecode = int(input("Enter employee code to search: "))
            while True:
                emp = pickle.load(f)
                if emp[0] == ecode:
                    empcode = emp[0]
                    name = emp[1]
                    place = emp[2]
                    salary = emp[3]
                    found = True
                    print(empcode, name,place, salary)
        except EOFError:
            pass
        if not found:
            print("Not Found")

def write_records():
    with open("emp.dat","ab") as f:
        n = int(input("Enter number of employees to be entered: "))
        for i in range(n):
            empcode = int(input("Enter employee code: "))
            name = input("Enter employee name: ")
            place = input("Enter employee place: ")
            salary = int(input("Enter employee salary: "))

            emp = [empcode,name,place,salary]
            pickle.dump(emp,f)

while True:
    ch = input("Enter choice:\n1. Search record\n2. Write record: ")
    if ch == "1":
        search_records()
    elif ch == "2":
        write_records()
    else:
        print("Invalid choice")
```

```
Enter choice:
1. Search record
2. Write record: 2
Enter number of employees to be entered: 2
Enter employee code: 100
Enter employee name: Ajay
Enter employee place: Kerala
Enter employee salary: 40000
Enter employee code: 110
Enter employee name: Krishna
Enter employee place: Kerala
Enter employee salary: 34000
Enter choice:
1. Search record
2. Write record: 1
Enter employee code to search: 200
Not Found
Enter choice:
1. Search record
2. Write record: 1
Enter employee code to search: 100
100 Ajay Kerala 40000
Enter choice:
1. Search record
2. Write record:
```

Program 13

```python
import pickle

def search_records():
    with open("emp.dat", "rb") as f:
        found = False
        try:
```

```
            ecode = int(input("Enter employee code to search: "))
            while True:
                emp = pickle.load(f)
                if emp["ecode"] == ecode:
                    empcode = emp["ecode"]
                    name = emp["ename"]
                    place = emp["eplace"]
                    salary = emp["esal"]
                    found = True
                    print(empcode, name,place, salary)
        except EOFError:
            pass
        if not found:
            print("Not Found")

def write_records():
    with open("emp.dat","ab") as f:
        n = int(input("Enter number of employees to be entered: "))
        for i in range(n):
            empcode = int(input("Enter employee code: "))
            name = input("Enter employee name: ")
            place = input("Enter employee place: ")
            salary = int(input("Enter employee salary: "))

            emp = {}
            emp["ecode"] = empcode
            emp["ename"] = name
            emp["eplace"] = place
            emp["esal"] = salary
            pickle.dump(emp,f)

while True:
    ch = input("Enter choice:\n1. Search record\n2. Write record: ")
    if ch == "1":
        search_records()
    elif ch == "2":
        write_records()
    else:
        print("Invalid choice")
```

```
Enter choice:
1. Search record
2. Write record: 2
Enter number of employees to be entered: 1
Enter employee code: 300
Enter employee name: Abhinav
Enter employee place: Kannur
Enter employee salary: 45000
Enter choice:
1. Search record
2. Write record: 2
Enter number of employees to be entered: 1
Enter employee code: 100
Enter employee name: Ajay
Enter employee place: Kochi
Enter employee salary: 50000
Enter choice:
1. Search record
2. Write record: 1
Enter employee code to search: 200
Not Found
Enter choice:
1. Search record
2. Write record: 1
Enter employee code to search: 100
100 Ajay Kochi 50000
Enter choice:
1. Search record
2. Write record: 1
Enter employee code to search: 300
300 Abhinav Kannur 45000
Enter choice:
1. Search record
2. Write record:
```

Program 14

```
import csv

def read_record():
    with open("emp.csv", "r") as f:

        csv_r = csv.reader(f)
        for i in csv_r:
            ecode = i[0]
            ename = i[1]
            esal = i[2]
            print(ecode,ename,esal)

def write_records():
    with open("emp.csv", "a") as f:
        csv_w = csv.writer(f)

        n = int(input("Enter number of employees to add: "))

        for i in range(n):
            empcode = int(input("Enter employee code: "))
            name = input("Enter employee name: ")
            salary = int(input("Enter employee salary: "))

            emp = [empcode,name,salary]
            csv_w.writerow(emp)


while True:
    ch = input("Enter choice:\n1. Read record\n2. Write record: ")
    if ch == "1":
        read_record()
    elif ch == "2":
        write_records()
    else:
        print("Invalid choice")
```

```
Enter choice:
1. Read record
2. Write record: 2
Enter number of employees to add: 2
Enter employee code: 100
Enter employee name: Arnav
Enter employee salary: 30000
Enter employee code: 101
Enter employee name: Abhilash
Enter employee salary: 40000
Enter choice:
1. Read record
2. Write record: 1
100 Arnav 30000
101 Abhilash 40000
Enter choice:
1. Read record
2. Write record:
```

```
100,Arnav,30000
101,Abhilash,40000
```

Program 15

```python
import csv

def search_record():
    with open("emp.csv", "r") as f:
        csv_r = csv.reader(f)
        ecode = int(input("Enter ecode: "))
        found = False
        for i in csv_r:
            if int(i[0]) == ecode:
```

```
                found = True
                ecode = i[0]
                ename = i[1]
                esal = i[2]
                print(ecode,ename,esal)
        if not found:
            print("Not found")

def write_records():
    with open("emp.csv", "a") as f:
        csv_w = csv.writer(f)

        n = int(input("Enter number of employees to add: "))

        for i in range(n):
            empcode = int(input("Enter employee code: "))
            name = input("Enter employee name: ")
            salary = int(input("Enter employee salary: "))

            emp = [empcode,name,salary]
            csv_w.writerow(emp)


while True:
    ch = input("Enter choice:\n1. Search record\n2. Write record: ")
    if ch == "1":
        search_record()
    elif ch == "2":
        write_records()
    else:
        print("Invalid choice")
```

```
Enter choice:
1. Search record
2. Write record: 2
Enter number of employees to add: 1
Enter employee code: 1
Enter employee name: Abhay
Enter employee salary: 40000
Enter choice:
1. Search record
2. Write record: 2
Enter number of employees to add: 1
Enter employee code: 2
Enter employee name: Amritha
Enter employee salary: 35000
Enter choice:
1. Search record
2. Write record: 1
Enter ecode: 4
Not found
Enter choice:
1. Search record
2. Write record: 1
Enter ecode: 2
2 Amritha 35000
Enter choice:
1. Search record
2. Write record:
```

```
1,Abhay,40000
2,Amritha,35000
```

Program 16

```
import pickle
import os

def writerecord():
    with open("emp.dat","ab") as f:

        enono = int(input("Enter no: "))
        ename = input("Enter name: ")
        esal = int(input("Enter salary: "))

        emp =[enono,ename,esal]
        pickle.dump(emp,f)

def display():
    with open("emp.dat", "rb") as f:
        try:
            while True:
                emp = pickle.load(f)
                empcode = emp[0]
                name = emp[1]
                salary = emp[2]

                print(empcode, name, salary)
        except EOFError:
            pass

def remove():
    f = open("emp.dat","rb")
    new_f = open("emp_backup.dat","wb")

    try:
        eno= int(input("Enter employee number to delete: "))
        while True:
            emp = pickle.load(f)
            if emp[0] == eno:
                continue
            pickle.dump(emp,new_f)
    except EOFError:
        pass
```

```
    os.remove("emp.dat")
    os.rename("emp_backup.dat","emp.dat")


while True:
    ch = input("Enter choice:\n1. Add record\n2. Remove record\n3. Display record: ")
    if ch == "1":
        writerecord()
    elif ch == "2":
        remove()
    elif ch == "3":
        display()
```

```
Enter choice:
1. Add record
2. Remove record
3. Display record: 1
Enter no: 1
Enter name: Ramesh
Enter salary: 40000
Enter choice:
1. Add record
2. Remove record
3. Display record: 1
Enter no: 2
Enter name: Rahul
Enter salary: 50000
Enter choice:
1. Add record
2. Remove record
3. Display record: 1
Enter no: 3
Enter name: Abhay
Enter salary: 45000
Enter choice:
1. Add record
2. Remove record
3. Display record: 3
1 Ramesh 40000
2 Rahul 50000
3 Abhay 45000
Enter choice:
1. Add record
2. Remove record
3. Display record: 2
Enter employee number to delete: 4
```

```
Enter choice:
1. Add record
2. Remove record
3. Display record: 3
1 Ramesh 40000
2 Rahul 50000
3 Abhay 45000
Enter choice:
1. Add record
2. Remove record
3. Display record: 2
Enter employee number to delete: 1
Enter choice:
1. Add record
2. Remove record
3. Display record: 3
2 Rahul 50000
3 Abhay 45000
Enter choice:
1. Add record
2. Remove record
3. Display record:
```

Program 17

```
import pickle

def add():
    with open("emp.dat","ab") as f:
        eidno = int(input("Enter employee number: "))
        ename = input("Enter employee name: ")
        esal = int(input("Enter employee salary: "))
```

```
        emp = [eidno,ename,esal]
        pickle.dump(emp,f)


def display():
    with open("emp.dat", "rb") as f:
        try:
            while True:
                emp = pickle.load(f)
                empcode = emp[0]
                name = emp[1]
                salary = emp[2]

                print(empcode, name, salary)
        except EOFError:
            pass

def updateesal():
    with open("emp.dat","rb+") as f:
        try:
            found = False
            eid= int(input("Enter eid to update: "))
            while True:
                orig = f.tell()
                emp = pickle.load(f)
                if emp[0] == eid:
                    f.seek(orig)
                    emp[2] = int(input("Enter new salary: "))
                    found= True
                    pickle.dump(emp,f)
        except EOFError:
            pass
        if not found:
            print("Not found")



while True:
    ch = input("Enter choice:\n1. Add record\n2. Update record\n3. Display record: ")
    if ch == "1":
        add()
    elif ch == "2":
        updateesal()
    elif ch == "3":
        display()
```

```
Enter choice:
1. Add record
2. Update record
3. Display record: 1
Enter employee number: 1
Enter employee name: Arun
Enter employee salary: 40000
Enter choice:
1. Add record
2. Update record
3. Display record: 1
Enter employee number: 2
Enter employee name: Arjun
Enter employee salary: 34500
Enter choice:
1. Add record
2. Update record
3. Display record: 3
1 Arun 40000
2 Arjun 34500
Enter choice:
1. Add record
2. Update record
3. Display record: 2
Enter eid to update: 1
Enter new salary: 60500
Enter choice:
1. Add record
2. Update record
3. Display record: 3
1 Arun 60500
2 Arjun 34500
Enter choice:
1. Add record
2. Update record
3. Display record: 2
Enter eid to update: 5
Not found
Enter choice:
1. Add record
2. Update record
3. Display record:
```

Program 18

```
import pickle
import os

def writerecord():
    rollno = int(input("Enter roll: "))
    name = input("Enter name: ")
    mark = int(input("Enter marks: "))

    with open("stu.dat","ab") as f:
        stu = [rollno,name,mark]
        pickle.dump(stu,f)

def display():
    with open("stu.dat", "rb") as f:
        try:
            while True:
                stu = pickle.load(f)
                rollno = stu[0]
                name = stu[1]
                mark = stu[2]

                print(rollno, name, mark)
        except EOFError:
            pass

def remove():
    f = open("stu.dat","rb")
    new_f = open("stu_backup.dat","wb")

    try:
        roll= int(input("Enter roll to delete: "))
        while True:
            orig = f.tell()
            stu = pickle.load(f)
            if stu[0] == roll:
                continue
            pickle.dump(stu,new_f)
    except EOFError:
        pass


    os.remove("stu.dat")
    os.rename("stu_backup.dat","stu.dat")
```

```
while True:
    ch = input("Enter choice:\n1. Add record\n2. Remove record\n3. Display record: ")
    if ch == "1":
        writerecord()
    elif ch == "2":
        remove()
    elif ch == "3":
        display()
```

```
Enter choice:
1. Add record
2. Remove record
3. Display record: 1
Enter roll: 1
Enter name: Raman
Enter marks: 30
Enter choice:
1. Add record
2. Remove record
3. Display record: 1
Enter roll: 2
Enter name: Arjun
Enter marks: 29
Enter choice:
1. Add record
2. Remove record
3. Display record: 3
1 Raman 30
2 Arjun 29
Enter choice:
1. Add record
2. Remove record
3. Display record: 2
Enter roll to delete: 1
Enter choice:
1. Add record
2. Remove record
3. Display record: 3
2 Arjun 29
Enter choice:
1. Add record
2. Remove record
3. Display record: 2
Enter roll to delete: 4
Enter choice:
1. Add record
2. Remove record
3. Display record:
```

Program 19

```
import pickle

def add():
    with open("stu.dat","ab") as f:
        rollno = int(input("Enter roll: "))
        name = input("Enter name: ")
        mark = int(input("Enter marks: "))
        stu = [rollno,name,mark]
        pickle.dump(stu,f)

def display():
    with open("stu.dat", "rb") as f:
        try:
            while True:
                stu = pickle.load(f)
                rollno = stu[0]
                name = stu[1]
                mark = stu[2]

                print(rollno, name, mark)
        except EOFError:
            pass

def updatemark():
    with open("stu.dat","rb+") as f:
        try:
            found = False
            roll= int(input("Enter roll to update: "))
            while True:
                orig = f.tell()
                stu = pickle.load(f)
                if stu[0] == roll:
                    f.seek(orig)
                    stu[2] = int(input("Enter new mark: "))
                    found= True
                    pickle.dump(stu,f)
        except EOFError:
            pass
        if not found:
            print("Not found")



while True:
```

```
ch = input("Enter choice:\n1. Add record\n2. Update record\n3. Display record: ")
if ch == "1":
    add()
elif ch == "2":
    updatemark()
elif ch == "3":
    display()
```

```
Enter choice:
1. Add record
2. Update record
3. Display record: 1
Enter roll: 1
Enter name: Ramesh
Enter marks: 25
Enter choice:
1. Add record
2. Update record
3. Display record: 1
Enter roll: 2
Enter name: Amal
Enter marks: 30
Enter choice:
1. Add record
2. Update record
3. Display record: 3
1 Ramesh 25
2 Amal 30
Enter choice:
1. Add record
2. Update record
3. Display record: 2
Enter roll to update: 1
Enter new mark: 27
Enter choice:
1. Add record
2. Update record
3. Display record: 3
1 Ramesh 27
2 Amal 30
Enter choice:
1. Add record
2. Update record
3. Display record: 2
Enter roll to update: 4
Not found
Enter choice:
1. Add record
2. Update record
3. Display record:
```

Program 20

```
s = []

def push():
    bid = int(input("Enter id: "))
    bname = input("Enter name: ")
    bauthor = input("Enter author: ")
    book =[bid,bname,bauthor]
    s.append(book)


def pop():
    if len(s) == 0:
        print("Underflow")
        return

    bp = s.pop()
    bid = bp[0]
    bname = bp[1]
    bauthor =  bp[2]

    print('The book to be popped is: ')
    print("Id: ",bid)
    print("Name: ",bname)
    print("Author: ",bauthor)

while True:
    ch = input("Menu:\n1. Push\n2. Pop: ")
    if ch == "1":
        push()
    elif ch == "2":
        pop()
    else:
        print("invalid input")
```

```
Menu:
1. Push
2. Pop: 1
Enter id: 1001
Enter name: The Christmas Pig
Enter author: JK Rowling
Menu:
1. Push
2. Pop: 1
Enter id: 1002
Enter name: The Maverick Effect
Enter author: Harish Mehta
Menu:
1. Push
2. Pop: 2
The book to be popped is:
Id:  1002
Name:  The Maverick Effect
Author:  Harish Mehta
Menu:
1. Push
2. Pop: 1
Enter id: 1101
Enter name: A Place Called Home
Enter author: Preeti Shenoy
Menu:
1. Push
2. Pop: 2
The book to be popped is:
Id:  1101
Name:  A Place Called Home
Author:  Preeti Shenoy
Menu:
1. Push
2. Pop: 2
The book to be popped is:
Id:  1001
Name:  The Christmas Pig
Author:  JK Rowling
Menu:
1. Push
2. Pop: 2
Underflow
Menu:
1. Push
2. Pop:
```

Program 21

```
s = []

def push():
    emid = int(input("Enter id: "))
    emname = input("Enter name: ")
    emsal = int(input("Enter salary: "))
    emp = [emid,emname,emsal]
    s.append(emp)

def pop():
    if len(s) == 0:
        print("Underflow")
        return

    emp = s.pop()
    emid = emp[0]
    emname = emp[1]
    emsal =  emp[2]

    print("The employee to be popped is: ")
    print("Id: ",emid)
    print("Name: ",emname)
    print("Salary: ",emsal)

while True:
    ch = input("Menu:\n1. Push\n2. Pop: ")
    if ch == "1":
        push()
    elif ch == "2":
        pop()
    else:
        print("invalid input")
```

```
Menu:
1. Push
2. Pop: 1
Enter id: 100
Enter name: Rahul
Enter salary: 40000
Menu:
1. Push
2. Pop: 1
Enter id: 101
Enter name: Ramesh
Enter salary: 50000
Menu:
1. Push
2. Pop: 2
The employee to be popped is:
Id:  101
Name:  Ramesh
Salary:  50000
Menu:
1. Push
2. Pop: 2
The employee to be popped is:
Id:  100
Name:  Rahul
Salary:  40000
Menu:
1. Push
2. Pop: 2
Underflow
Menu:
1. Push
2. Pop:
```