Practical 1: Creating a python program named “Mycsv.py” to store the details of Empno, Name and Salary in “Emp.csv” and search for an Empno entered by the user. If found, display the details of the Employee else display Empno is not present.

Source code

```
import csv
f = open("pract2.csv", "w+", newline="")
csv_w = csv.writer(f, delimiter = ",")
fields = ["Emp_No.","Name", "Salary"]
row = []
rec = []
n = int(input("Enter no. of required records:"))
for i in range(n):
 rec = eval(input("Enter Employee no, name, and salary:"))
 row.append(rec)
print(row)
csv_w.writerow(fields)
for y in row:
 csv_w.writerow(y)
print("File created.")
f.close()
f = open("pract2.csv", "r+", newline="")
c=[]
d = "n"
csv_r = csv.reader(f, delimiter = ",")
for j in csv_r:
 c.append(j)
search = input("Enter Employee no.:")
or i in c:
 if i[0] == search:
 emp = [i[1], i[2]]
 d = "y"
 else:
 pass
if d == "y":
 print("Employee:",emp)
else:
 print("Employee does not exist.")
f.close()
```

Practical 2: Write a python program named “my.py” to create a binary file with name and roll number. Search for a given roll number and display the name, if not found display appropriate message.

Source code

```
import pickle
record = []
stud_rec = []
n = int(input("Enter no. of students:"))
for i in range(n):
 roll = int(input("Enter roll no.:"))
 name = input("Enter name:")
 data = [roll, name]
 record.append(data)
print(record)
f = open("school", "wb")
pickle.dump(record, f)
print("Records have been added.")
f.close()
f = open("school", "rb")
stud_rec = pickle.load(f)
f.close()
search = int(input("Enter roll no. of required student:"))
d = "n"
for i in stud_rec:
 if i[0] == search:
 d = "y"
name = i[1]
 break
 else:
```

```
 pass
if d == "y":
 print("Student:", name)
else:
 print("Student does not exist.")
```

Practical 3: Write a menu driven python program using list to implement stack data structure and perform the following operations:

a. Push(), b. Pop(), c. Peek(), d. Display()

Source code

```
stack = []
def push(item):
    stack.append(item)
    print(f"{item} pushed onto the stack.")
def pop():
    if not is_empty():
        return stack.pop()
    else:
        print("Stack is empty. Cannot pop.")
        return None
def peek():
    if not is_empty():
        return stack[-1]
    else:
        print("Stack is empty. Cannot peek.")
        return None
def display():
    if not is_empty():
        print("Stack elements:", stack)
    else:
        print("Stack is empty.")
def is_empty():
    return len(stack) == 0
def main():
    while True:
        print("\nStack Operations:")
        print("1. Push")
        print("2. Pop")
        print("3. Peek")
        print("4. Display")
        print("5. Quit")
        choice = input("Enter your choice (1-5): ")
        if choice == '1':
            item = input("Enter element to push onto the stack: ")
            push(item)
        elif choice == '2':
            popped_item = pop()
            if popped_item is not None:
                print(f"Popped item: {popped_item}")
        elif choice == '3':
            top_item = peek()
            if top_item is not None:
                print(f"Top item: {top_item}")
        elif choice == '4':
            display()
        elif choice == '5':
            print("Exiting the program. Goodbye!")
            break
        else:
            print("Invalid choice. Please enter a valid option (1-5).")
if __name__ == "__main__":
    main()
```

Practical 4: Write a python program to create a binary file “student.dat” with roll number, name and marks. Input a roll number from user and update their marks.

Source code

```
import pickle
record = []
```

```
stud_rec = []
n = int(input("Enter no. of students:"))
for i in range(n):
 roll = int(input("Enter roll no.:"))
 name = input("Enter name:")
 marks = int(input("Enter marks:"))
 data = [roll, name, marks]
 record.append(data)
print(record)
f = open("school", "wb")
pickle.dump(record, f)
print("Records have been added.")
f.close()
f = open("school", "rb")
stud_rec = pickle.load(f)
search = int(input("Enter roll no. of required student:"))
mar = int(input("Enter new marks:"))
d = "n"
for i in stud_rec:
 if i[0] == search:
 d = "y"
 i[2] = mar
 break
 else:
 pass
if d == "y":
 print("Updated record:", stud_rec)
else:
 print("Student does not exist.")
```

Practical 5: Write a python program to create a CSV file by entering user-id and password, read and search the password for given user-id.

Source code:

```
import csv
```

```
f = open("pract.csv", "w+", newline="")
csv_w = csv.writer(f, delimiter = ",")
fields = ["User-ID","Password"]
row = []
rec = []
n = int(input("Enter no. of required records:"))
for i in range(n):
 rec = eval(input("Enter User-ID and Password:"))
 row.append(rec)
print(row)
csv_w.writerow(fields)
for y in row:
 csv_w.writerow(y)
print("File created.")
f.close()
f = open("pract.csv", "r", newline="")
csv_r = csv.reader(f, delimiter = ",")
for j in csv_r:
 print(j)
f.close()
f = open("pract.csv", "r", newline="")
c=[]
d = "n"
csv_r = csv.reader(f, delimiter = ",")
for j in csv_r:
 c.append(j)
search = input("Enter User-ID:")
for i in c:
 if i[0] == search:
password = i[1]
 d = "y"
 else:
 pass
if d == "y":
 print("Your password is:",password)
else:
 print("User-ID does not exist.")
f.close()
```