



COMPUTER SCIENCE

LAB PRACTICALS

2020-21

LANGUAGE : PYTHON

NAME
 **ARSH MISHRA**

CLASS
XII – D

ROLL NO
14744885

LAB PRACTICAL 1

(REVISION TOUR)

Q-1

Calculate area and circumference of circle while radius is entered by user.

|PROGRAMMING|

```
#calculate area and circumference of circle
r=float(input('Enter radius of circle : '))
c=2*3.14*r
print('Circumference : ',c)
a=3.14*r**2
print('Area : ',a)
```

|OUTPUT|

```
Enter radius of circle : 4.5
Circumference : 28.26
Area : 63.585
```

Q-2

Input value in seconds and convert in hours, minutes and seconds.

|PROGRAMMING|

```
#input in seconds and convert in hours, minutes and seconds
v=int(input('Enter the seconds : '))
hh=v//(60*60)
hr=v%(60*60)
mm=hr//60
mr=hr%60
ss=mr
print('HH : ',hh,'MM : ',mm,'SS : ',ss)
```

|OUTPUT|

```
Enter the seconds : 15731
HH : 4 MM : 22 SS : 11
```

Q-3

Accept ASCII or character from user and return corresponding character or ASCII.

|PROGRAMMING|

```
#display ASCII code from character or character from ASCII code
while True:
    print('|Choose from following options|')
    print('1.ASCII to character\n','2.Character to ASCII',sep='')
    c=int(input('Enter your choice : '))
    if c==1:
        n=int(input('Enter the ASCII code : '))
        print('Character is : ',chr(n))
```

```
elif c==2:
    n=input('Enter the character : ')
    print('ASCII code is :',ord(n))
else:
    print('Wrong Input.Try Again')
c=input('Do you want to continue(Y/N) : ')
if c!='Y' and c!='y':
    break
```

[OUTPUT]

```
|Choose from following options|
1.ASCII to character
2.Character to ASCII
Enter your choice : 2
Enter the character : d
ASCII code is : 100
Do you want to continue(Y/N) : y
|Choose from following options|
1.ASCII to character
2.Character to ASCII
Enter your choice : 6
Wrong Input.Try Again
Do you want to continue(Y/N) : y
|Choose from following options|
1.ASCII to character
2.Character to ASCII
Enter your choice : 1
Enter the ASCII code : 76
Character is : L
Do you want to continue(Y/N) : n
```

LAB PRACTICAL 2

(FUNCTIONS)

Q-1

Display prime numbers from 1 to the number entered by user.

|PROGRAMMING|

```
#display prime numbers between 1 to n
def disp_prime(n):
    for i in range(1,n+1):
        nof=0
        for j in range(1,i+1):
            if i%j==0:
                nof=nof+1
        if nof==2:
            print(i,end='\t')
n=int(input('Enter the upto number : '))
disp_prime(n)
```

|OUTPUT|

```
Enter the upto number : 19
2      3      5      7      11     13     17     19
```

Q-2

Check is string is palindrome.

|PROGRAMMING|

```
#if string is palindrome
def rev(s):
    r=s[::-1]
    return r
def ispali(s,r):
    if s.lower()==r.lower():
        return True
    else:
        return False
s=input('Enter the string : ')
r=rev(s)
ispali=ispali(s,r)
if ispali==True:
    print('String is palindrome')
else:
    print('String is not palindrome')
```

|OUTPUT|

Enter the string : malayalam
String is palindrome

Enter the string : mumbai
String is not palindrome

Q-3

Find sum of series : $(1)+(1+2)+(1+2+3)+\dots$ upto n terms, where n is entered by the user.

|PROGRAMMING|

```
#find sum of series:(1)+(1+2)+(1+2+3)+....+upto n terms
def sum(n):
    sum=0
    for i in range(1,n+1):
        for j in range(1,i+1):
            sum=sum+j
    return sum
print('|Sum of series (1)+(1+2)+(1+2+3)+....+upto n terms|')
n=int(input('Enter the upto terms : '))
sum=sum(n)
print('Sum of the series upto',n,'terms is :',sum)
```

|OUTPUT|

|Sum of series (1)+(1+2)+(1+2+3)+....+upto n terms|
Enter the upto terms : 5
Sum of the series upto 5 terms is : 35

LAB PRACTICAL 3

(LIBRARY)

Use Python library.

|Operation Module|

```
def add(x,y):  
    return x+y  
def sub(x,y):  
    return x-y  
def mul(x,y):  
    return x*y  
def div(x,y):  
    return x/y
```

|PROGRAMMING|

```
import operations as o  
print('|Menu|\n'  
      '1.Addition\n'  
      '2.Subtraction\n'  
      '3.Multiplication\n'  
      '4.Division\n'  
      '5.Quit\n')  
n1=int(input('Enter first number : '))  
n2=int(input('Enter second number : '))  
print()  
while True:  
    n=int(input('Enter you choice(1,2,3,4,5) : '))  
    if n==1:  
        print('Result is :',o.add(n1,n2))  
    elif n==2:  
        print('Result is :',o.sub(n1,n2))  
    elif n==3:  
        print('Result is :',o.mul(n1,n2))  
    elif n==4:  
        print('Result is :',round(o.div(n1,n2),2))  
    elif n==5:  
        break  
    else:  
        print('Wrong Input')
```

|OUTPUT|

```
|Menu|  
1.Addition  
2.Subtraction  
3.Multiplication  
4.Division  
5.Quit
```

Enter first number : 43

Enter second number : 21

Enter you choice(1,2,3,4,5) : 3

Result is : 903

Enter you choice(1,2,3,4,5) : 1

Result is : 64

Enter you choice(1,2,3,4,5) : 2

Result is : 22

Enter you choice(1,2,3,4,5) : 4

Result is : 2.05

Enter you choice(1,2,3,4,5) : 6

Wrong Input

Enter you choice(1,2,3,4,5) : 5

LAB PRACTICAL 4

(LIST)

Q-1

Reverse a list.

|PROGRAMMING|

```
#reverse a list
def rev(l):
    mid=int(len(l)/2)
    i,j=0,len(l)-1
    while i<mid:
        l[i],l[j]=l[j],l[i]
        i,j=i+1,j-1
n=int(input('Enter the size of the list : '))
l=[int(input('Enter the element '+str(i+1)+' : ')) for i in range(n)]
print('Entered list is :',l)
rev(l)
print('New list is :',l)
```

|OUTPUT|

```
Enter the size of the list : 4
Enter the element 1 : 5
Enter the element 2 : 8
Enter the element 3 : 3
Enter the element 4 : 9
Entered list is : [5, 8, 3, 9]
New list is : [9, 3, 8, 5]
```

Q-2

Bubble sort.

|PROGRAMMING|

```
#main
n=int(input("Enter the size of the array : "))
a=[0]*n
print("|Enter the array elements|")
for i in range(n):
    a[i]=int(input("Enter element "+str(i+1)+" : "))
print("|Array entered by user|")
print(a)
#bubble sort
for i in range(n):
    for j in range(n-i-1):
        if a[j]>a[j+1]:
            a[j],a[j+1]=a[j+1],a[j]
print("|Array after sorting(ascending)|")
print(a)
```


|OUTPUT|

```
Enter the size of the array : 5
|Enter the array elements|
Enter element 1 : 3
Enter element 2 : 6
Enter element 3 : 9
Enter element 4 : 5
Enter element 5 : 4
|Array entered by user|
[3, 6, 9, 5, 4]
|Array after sorting(ascending)|
[3, 4, 5, 6, 9]
```

Q-3

Perform binary search by sorting list first with insertion sort.

|PROGRAMMING|

```
#binary search by sorting first with insertion sort
def insertion(l):
    for i in range(1,len(l)):
        value=l[i]
        j=i-1
        while j>=0 and value<l[j]:
            l[j+1]=l[j]
            j-=1
        l[j+1]=value
def bsearch(l,x):
    beg=0
    end=len(l)-1
    while beg<=end:
        mid=(beg+end)//2
        if(l[mid]==x):
            return mid
        elif(l[mid]>x):
            end=mid-1
        else:
            beg=mid+1
    return -1
#main
n=int(input("Enter the size of the array : "))
l=[]
for i in range(n):
    l.append(int(input("Enter the element "+str(i+1)+" : ")))
print("|Array entered by the user|")
print(l)
print("|Sorted array in ascending order|")
insertion(l)
print(l)
while(True):
    x=int(input("Enter the element to be searched : "))
    idx=bsearch(l,x)
```

```
if(idx>=0):
    print("Element found at position",idx+1)
else:
    print("Element not found")
c=input("Do you want to continue(Y/y) : ")[0]
if(c!='Y' and c!='y'):
    break
```

|OUTPUT|

```
Enter the size of the array : 5
Enter the element 1 : 7
Enter the element 2 : 3
Enter the element 3 : 6
Enter the element 4 : 1
Enter the element 5 : 9
|Array entered by the user|
[7, 3, 6, 1, 9]
|Sorted array in ascending order|
[1, 3, 6, 7, 9]
Enter the element to be searched : 6
Element found at position 3
Do you want to continue(Y/y) : y
Enter the element to be searched : 9
Element found at position 5
Do you want to continue(Y/y) : y
Enter the element to be searched : 2
Element not found
Do you want to continue(Y/y) : n
```

LAB PRACTICAL 5

(FILE HANDLING)

Q-1

Count number of consonant in text file.

|TEXT FILE DATA|

ik tura Un bela
gro Ter jecu kai

|PROGRAMMING|

```
#count number of consonant
f=open('text.txt','r')
vow=['a','e','i','o','u']
cnt=0
while True:
    l=f.readline()
    if l=="":
        break
    l=l.strip()
    for i in l:
        if i.lower() not in vow and i!=' ':
            cnt=cnt+1
f.close()
print('Number of consonant :',cnt)
```

|OUTPUT|

Number of consonant : 13

Q-2

Reverse every word of the text file.

|TEXT FILE DATA|

ik tura Un bela
gro Ter jecu kai

|PROGRAMMING|

```
#reversing every word of text file
f=open('text.txt','r')
data=[]
while True:
    l=f.readline()
    if len(l)==0:
        break
    l=l.strip()
    l=l.split()
    for i in range(len(l)):
        l[i]=l[i][::-1]
    l=' '.join(l)
```

```
l=l+'\n'
data.append(l)
f.close()
f=open('text.txt','w')
f.writelines(data)
f.close()
print('Reversed Successfully')
```

|OUTPUT|

Reversed Successfully

|NEW TEXT FILE DATA|

ki arut nU Aleb
org reT ucej iak

Q-3

Read record in binary files.

|PROGRAMMING|

```
#read record in binary
import pickle as p
f=open('rec binary.txt','rb')
while True:
    try:
        rec=p.load(f)
        print(rec)
    except:
        break
f.close()
```

|OUTPUT|

[1, 'Raghav', 78]
[2, 'Sushant', 72]
[3, 'Tarun', 88]

Q-4

Delete entry from CSV file.

|CSV FILE DATA|

1,Rahul,89
2,Tarun,97
3,Zakir,94
4,Rajesh,72
5,Umesh,85

|PROGRAMMING|

```
#delete entry from csv file
import csv
f=open('csv file.csv','r')
n=int(input('Enter the roll number of student to be deleted : '))
recs=[]
flag=0
reader=csv.reader(f)
for rec in reader:
    if rec[0]==str(n):
        flag=1
    elif rec[0]!=str(n):
        recs.append(rec)
f.close()
if flag==0:
    print('NOT FOUND')
else:
    f=open('csv file.csv','w',newline='')
    writer=csv.writer(f)
    writer.writerows(recs)
    f.close()
    print('SUCCESSFULLY DELETED')
```

|OUTPUT|

```
Enter the roll number of student to be deleted : 2
SUCCESSFULLY DELETED
```

|NEW CSV FILE DATA|

```
1,Rahul,89
3,Zakir,94
4,Rajesh,72
5,Umesh,85
```

LAB PRACTICAL 6

(STACKS AND QUEUES)

Q-1

Program of stack using list.

|PROGRAMMING|

```
#stack
def isempty(s):
    if len(s)==0:
        return True
    else:
        return False
def top(s):
    top=len(s)-1
    return top
def push(s,v):
    s.append(v)
def pop(s):
    if isempty(s):
        return 'Underflow'
    else:
        x=s.pop()
        return x
def peek(s):
    if isempty(s):
        return 'Empty'
    else:
        t=top(s)
        x=s[t]
        return x
def display(s):
    if isempty(s):
        return 'Empty'
    else:
        t=top(s)
        print('Stack is as follows : ')
        print(s[t],"-> top")
        for i in range(t-1,-1,-1):
            print(s[i])
#main
s=list()
print('|Choose from the following option|')
print('1.Push\n','2.Pop\n','3.Peek\n','4.Display\n','5.Quit\n',sep='')
while True:
    c=int(input('Enter your choice (1/2/3/4/5) : '))
    if c==1:
        val=int(input('Enter the item you want to push : '))
        push(s,val)
        print('Done')
    elif c==2:
        val=pop(s)
```

```

        if val=='Underflow':
            print('|Underflow : String is empty|')
        else:
            print('Done\n','Popped value is :',val,sep='')
    elif c==3:
        val=peek(s)
        if val=='Empty':
            print('|String is empty|')
        else:
            print('Done\n','Peeked value is :',val,sep='')
    elif c==4:
        flag=display(s)
        if flag=='Empty':
            print('|Nothing to display : String is empty|')
        else:
            print('Done')
    elif c==5:
        break
    else:
        print('|Wrong Input|')

```

|OUTPUT|

|Choose from the following option|

- 1.Push
- 2.Pop
- 3.Peek
- 4.Display
- 5.Quit

Enter your choice (1/2/3/4/5) : 3

|String is empty|

Enter your choice (1/2/3/4/5) : 4

|Nothing to display : String is empty|

Enter your choice (1/2/3/4/5) : 2

|Underflow : String is empty|

Enter your choice (1/2/3/4/5) : 1

Enter the item you want to push : 54

Done

Enter your choice (1/2/3/4/5) : 1

Enter the item you want to push : 33

Done

Enter your choice (1/2/3/4/5) : 1

Enter the item you want to push : 45

Done

Enter your choice (1/2/3/4/5) : 4

Stack is as follows :

45 -> top

33

54

Done

Enter your choice (1/2/3/4/5) : 3

```
Done
Peeked value is :45
Enter your choice (1/2/3/4/5) : 2
Done
Popped value is :45
Enter your choice (1/2/3/4/5) : 1
Enter the item you want to push : 65
Done
Enter your choice (1/2/3/4/5) : 4
Stack is as follows :
65 -> top
33
54
Done
Enter your choice (1/2/3/4/5) : 6
|Wrong Input|
Enter your choice (1/2/3/4/5) : 5
```

Q-2

Program of queue using list.

|PROGRAMMING|

```
#queue
def isempty(q):
    if len(q)==0:
        return True
    else:
        return False
def isunit(q):
    if len(q)==1:
        return True
    else:
        return False
def front(q):
    front=0
    return front
def rear(q):
    rear=len(q)-1
    return rear
def enqueue(q,v):
    q.append(v)
def deque(q):
    if isempty(q):
        return 'Underflow'
    else:
        x=q.pop(0)
        return x
def peek(q):
    if isempty(q):
        return 'Empty'
    else:
```



```

        f=front(q)
        x=q[f]
        return x
def display(q):
    if isempty(q):
        return 'Empty'
    elif isunit(q):
        print(q[0],'-> front and rear')
    else:
        f=front(q)
        r=rear(q)
        print(q[f],'-> front')
        for i in range(1,r):
            print(q[i])
        print(q[r],'-> rear')
#main
q=list()
print('|Choose from the following option|')
print('1.Enqueue\n','2.Dequeue\n','3.Peek\n','4.Display\n','5.Quit\n',sep='')
while True:
    c=int(input('Enter your choice (1/2/3/4/5) : '))
    if c==1:
        val=int(input('Enter the item you want to push : '))
        enqueue(q,val)
        print('Done')
    elif c==2:
        val=deque(q)
        if val=='Underflow':
            print('|Underflow : String is empty|')
        else:
            print('Done\n','Popped value is :',val,sep='')
    elif c==3:
        val=peek(q)
        if val=='Empty':
            print('|String is empty|')
        else:
            print('Done\n','Peeked value is :',val,sep='')
    elif c==4:
        flag=display(q)
        if flag=='Empty':
            print('|Nothing to display : String is empty|')
        else:
            print('Done')
    elif c==5:
        break
    else:
        print('|Wrong Input|')

```

[OUTPUT]

```

|Choose from the following option|
1.Enqueue
2.Dequeue

```

- 3.Peek
- 4.Display
- 5.Quit

Enter your choice (1/2/3/4/5) : 4
|Nothing to display : String is empty|
Enter your choice (1/2/3/4/5) : 3
|String is empty|
Enter your choice (1/2/3/4/5) : 2
|Underflow : String is empty|
Enter your choice (1/2/3/4/5) : 1
Enter the item you want to push : 23
Done
Enter your choice (1/2/3/4/5) : 1
Enter the item you want to push : 45
Done
Enter your choice (1/2/3/4/5) : 3
Done
Peeked value is :23
Enter your choice (1/2/3/4/5) : 4
23 -> front
45 -> rear
Done
Enter your choice (1/2/3/4/5) : 1
Enter the item you want to push : 33
Done
Enter your choice (1/2/3/4/5) : 4
23 -> front
45
33 -> rear
Done
Enter your choice (1/2/3/4/5) : 1
Enter the item you want to push : 28
Done
Enter your choice (1/2/3/4/5) : 4
23 -> front
45
33
28 -> rear
Done
Enter your choice (1/2/3/4/5) : 2
Done
Popped value is :23
Enter your choice (1/2/3/4/5) : 4
45 -> front
33
28 -> rear
Done
Enter your choice (1/2/3/4/5) : 6
|Wrong Input|
Enter your choice (1/2/3/4/5) : 5

Q-3

Reverse string using stack.

|PROGRAMMING|

```
#reverse string using stack
def push(s,v):
    s.append(v)
def pop(s):
    if len(s)==0:
        return 'Underflow'
    else:
        x=s.pop()
        return x
def reverse(old):
    stk=[]
    new=""
    for i in old:
        push(stk,i)
    for i in range(len(stk)):
        new=new+pop(stk)
    return new
old=input('Enter the string : ')
new=reverse(old)
print('New string is :',new)
```

|OUTPUT|

```
Enter the string : UNITED KINGDOM
New string is : MODGNIK DETINU
```

LAB PRACTICAL 7

(INTERFACE MYSQL)

Menu driven program to add, delete or show records in database.

[PROGRAMMING]

```
#interface with mysql
import mysql.connector as mysql
connect=mysql.connect(host='localhost',user='root',
                      passwd='12345',database='practical')
cursor=connect.cursor()
print('|MENU|')
print('1.Add record\n','2.Delete record\n','3.Show record',sep='')
while True:
    c=int(input('Choose from the menu : '))
    if c==1:
        roll=int(input('Enter roll number : '))
        name=input('Enter name : ')
        marks=int(input('Enter marks : '))
        command="insert into student \
                values({},'{}',{})".format(roll,name,marks)
        cursor.execute(command)
        connect.commit()
        print('Added Successfully')
    elif c==2:
        roll=int(input('Enter the roll number of record to delete : '))
        command="delete from student where roll={}".format(roll)
        cursor.execute(command)
        connect.commit()
        print('Deleted Successfully')
    elif c==3:
        command="select * from student"
        cursor.execute(command)
        l=cursor.fetchall()
        if l==[]:
            print('No Record')
        for i in l:
            print(i)
    else:
        print('Wrong Input.Try Again')
    c=input('Do you want to try again(Y/N) : ')
    if c!='Y' and c!='y':
        break
connect.close()
```

[OUTPUT]

```
|MENU|
1.Add record
2.Delete record
3.Show record
Choose from the menu : 1
```

Enter roll number : 1
Enter name : Raghav
Enter marks : 98
Added Successfully
Do you want to try again(Y/N) : y
Choose from the menu : 1
Enter roll number : 2
Enter name : Tarun
Enter marks : 78
Added Successfully
Do you want to try again(Y/N) : y
Choose from the menu : 1
Enter roll number : 3
Enter name : Jaspreet
Enter marks : 92
Added Successfully
Do you want to try again(Y/N) : y
Choose from the menu : 3
(1, 'Raghav', 98)
(2, 'Tarun', 78)
(3, 'Jaspreet', 92)
Do you want to try again(Y/N) : y
Choose from the menu : 2
Enter the roll number of record to delete : 2
Deleted Successfully
Do you want to try again(Y/N) : y
Choose from the menu : 3
(1, 'Raghav', 98)
(3, 'Jaspreet', 92)
Do you want to try again(Y/N) : n

LAB PRACTICAL 8

(SQL)

Q-1

```
mysql> create database loan;
Query OK, 1 row affected (0.05 sec)
```

```
mysql> use loan;
```

Database changed

```
mysql> create table loan_account
-> (accno integer primary key,
-> cust_name varchar(30) not null,
-> loan_amt integer,
-> installment integer,
-> int_rate decimal(5,2),
-> start_date date);
```

```
Query OK, 0 rows affected (0.09 sec)
```

```
mysql> desc loan_account;
```

Field	Type	Null	Key	Default	Extra
accno	int(11)	NO	PRI	NULL	
cust_name	varchar(30)	NO		NULL	
loan_amt	int(11)	YES		NULL	
installment	int(11)	YES		NULL	
int_rate	decimal(5,2)	YES		NULL	
start_date	date	YES		NULL	

```
6 rows in set (0.02 sec)
```

```
mysql> insert into loan_account
-> values(1,'R.K. GUPTA',300000,36,12.00,'2015-07-19');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into loan_account
-> values(2,'S.P. SHARMA',500000,48,10.00,'2012-03-22');
Query OK, 1 row affected (0.36 sec)
```

```
mysql> insert into loan_account
-> values(3,'K.P. JAIN',300000,36,null,'2013-03-08');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into loan_account
-> values(4,'M.P. YADAV',800000,60,10.00,'2012-12-06');
Query OK, 1 row affected (0.36 sec)
```

```
mysql> insert into loan_account
-> values(5,'S.P. SINHA',200000,36,12.50,'2014-01-03');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into loan_account
-> values(6,'P. SHARMA',700000,60,12.50,'2015-06-05');
Query OK, 1 row affected (0.03 sec)
```

```
mysql> insert into loan_account
-> values(7,'K.S. DHALL',500000,48,null,'2016-03-05');
Query OK, 1 row affected (0.00 sec)
```

```
mysql> select*from loan_account;
```

accno	cust_name	loan_amt	installment	int_rate	start_date
1	R.K. GUPTA	300000	36	12.00	2015-07-19
2	S.P. SHARMA	500000	48	10.00	2012-03-22
3	K.P. JAIN	300000	36	NULL	2013-03-08
4	M.P. YADAV	800000	60	10.00	2012-12-06
5	S.P. SINHA	200000	36	12.50	2014-01-03
6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	NULL	2016-03-05

7 rows in set (0.00 sec)

```
mysql> select accno,cust_name,loan_amt from loan_account;
```

accno	cust_name	loan_amt
1	R.K. GUPTA	300000
2	S.P. SHARMA	500000
3	K.P. JAIN	300000
4	M.P. YADAV	800000
5	S.P. SINHA	200000
6	P. SHARMA	700000
7	K.S. DHALL	500000

7 rows in set (0.01 sec)

```
mysql> select accno,loan_amt from loan_account
-> where start_date<'2013-04-01';
```

accno	loan_amt
2	500000
3	300000
4	800000

3 rows in set (0.00 sec)

```
mysql> select*from loan_account
-> where start_date>'2015-04-01';
```

accno	cust_name	loan_amt	installment	int_rate	start_date
1	R.K. GUPTA	300000	36	12.00	2015-07-19
6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	NULL	2016-03-05

3 rows in set (0.00 sec)

```
mysql> select*from loan_account
-> where int_rate is null;
```

```

+-----+-----+-----+-----+-----+-----+
| accno | cust_name | loan_amt | installment | int_rate | start_date |
+-----+-----+-----+-----+-----+-----+
|      3 | K.P. JAIN | 300000 |          36 |      NULL | 2013-03-08 |
|      7 | K.S. DHALL | 500000 |          48 |      NULL | 2016-03-05 |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

```

```

mysql> select*from loan_account
-> where int_rate is not null;

```

```

+-----+-----+-----+-----+-----+-----+
| accno | cust_name | loan_amt | installment | int_rate | start_date |
+-----+-----+-----+-----+-----+-----+
|      1 | R.K. GUPTA | 300000 |          36 |    12.00 | 2015-07-19 |
|      2 | S.P. SHARMA | 500000 |          48 |    10.00 | 2012-03-22 |
|      4 | M.P. YADAV | 800000 |          60 |    10.00 | 2012-12-06 |
|      5 | S.P. SINHA | 200000 |          36 |    12.50 | 2014-01-03 |
|      6 | P. SHARMA | 700000 |          60 |    12.50 | 2015-06-05 |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

```

```

mysql> select cust_name,loan_amt from loan_account
-> where loan_amt<500000 or int_rate>12;

```

```

+-----+-----+
| cust_name | loan_amt |
+-----+-----+
| R.K. GUPTA | 300000 |
| K.P. JAIN | 300000 |
| S.P. SINHA | 200000 |
| P. SHARMA | 700000 |
+-----+-----+
4 rows in set (0.00 sec)

```

```

mysql> select cust_name from loan_account
-> where loan_amt=500000 and int_rate>5;

```

```

+-----+
| cust_name |
+-----+
| S.P. SHARMA |
+-----+
1 row in set (0.00 sec)

```

```

mysql> select cust_name from loan_account
-> where loan_amt=500000 and (int_rate>5 or int_rate is null);

```

```

+-----+
| cust_name |
+-----+
| S.P. SHARMA |
| K.S. DHALL |
+-----+
2 rows in set (0.01 sec)

```

```

mysql> select*from loan_account
-> where start_date>='2013-01-01';

```

```

+-----+-----+-----+-----+-----+-----+
| accno | cust_name | loan_amt | installment | int_rate | start_date |
+-----+-----+-----+-----+-----+-----+

```



```

+-----+-----+-----+-----+-----+-----+
| 1 | R.K. GUPTA | 300000 | 36 | 12.00 | 2015-07-19 |
| 3 | K.P. JAIN | 300000 | 36 | NULL | 2013-03-08 |
| 5 | S.P. SINHA | 200000 | 36 | 12.50 | 2014-01-03 |
| 6 | P. SHARMA | 700000 | 60 | 12.50 | 2015-06-05 |
| 7 | K.S. DHALL | 500000 | 48 | NULL | 2016-03-05 |
+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)

```

```

mysql> select*from loan_account
-> where int_rate between 10 and 14;

```

```

+-----+-----+-----+-----+-----+-----+
| accno | cust_name | loan_amt | installment | int_rate | start_date |
+-----+-----+-----+-----+-----+-----+
| 1 | R.K. GUPTA | 300000 | 36 | 12.00 | 2015-07-19 |
| 2 | S.P. SHARMA | 500000 | 48 | 10.00 | 2012-03-22 |
| 4 | M.P. YADAV | 800000 | 60 | 10.00 | 2012-12-06 |
| 5 | S.P. SINHA | 200000 | 36 | 12.50 | 2014-01-03 |
| 6 | P. SHARMA | 700000 | 60 | 12.50 | 2015-06-05 |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.37 sec)

```

```

mysql> select cust_name from loan_account
-> where installment in(35,48,60);

```

```

+-----+
| cust_name |
+-----+
| S.P. SHARMA |
| M.P. YADAV |
| P. SHARMA |
| K.S. DHALL |
+-----+
4 rows in set (0.00 sec)

```

```

mysql> update loan_account
-> set cust_name='S.P. GARUN'
-> where cust_name='S.P. SINHA';
Query OK, 1 row affected (0.38 sec)
Rows matched: 1 Changed: 1 Warnings: 0

```

```

mysql> update loan_account
-> set cust_name='P.Q. TIWARI'
-> where accno=2;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1 Changed: 1 Warnings: 0

```

```

mysql> select*from loan_account;

```

```

+-----+-----+-----+-----+-----+-----+
| accno | cust_name | loan_amt | installment | int_rate | start_date |
+-----+-----+-----+-----+-----+-----+
| 1 | R.K. GUPTA | 300000 | 36 | 12.00 | 2015-07-19 |
| 2 | P.Q. TIWARI | 500000 | 48 | 10.00 | 2012-03-22 |
| 3 | K.P. JAIN | 300000 | 36 | NULL | 2013-03-08 |
| 4 | M.P. YADAV | 800000 | 60 | 10.00 | 2012-12-06 |
| 5 | S.P. GARUN | 200000 | 36 | 12.50 | 2014-01-03 |
| 6 | P. SHARMA | 700000 | 60 | 12.50 | 2015-06-05 |

```

```
|      7 | K.S. DHALL | 500000 |      48 | NULL | 2016-03-05 |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

```
mysql> select accno,cust_name,loan_amt from loan_account
-> where cust_name like '%r%';
```

```
+-----+-----+-----+
| accno | cust_name | loan_amt |
+-----+-----+-----+
|      1 | R.K. GUPTA | 300000 |
|      2 | P.Q. TIWARI | 500000 |
|      5 | S.P. GARUN | 200000 |
|      6 | P. SHARMA | 700000 |
+-----+-----+-----+
4 rows in set (0.00 sec)
```

```
mysql> select accno,cust_name,loan_amt from loan_account
-> where cust_name like '%a_';
```

```
+-----+-----+-----+
| accno | cust_name | loan_amt |
+-----+-----+-----+
|      4 | M.P. YADAV | 800000 |
+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select accno,cust_name,loan_amt from loan_account
-> where cust_name like '%n%';
```

```
+-----+-----+-----+
| accno | cust_name | loan_amt |
+-----+-----+-----+
|      3 | K.P. JAIN | 300000 |
|      5 | S.P. GARUN | 200000 |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

```
mysql> select*from loan_account
-> order by start_date desc;
```

```
+-----+-----+-----+-----+-----+-----+
| accno | cust_name | loan_amt | installment | int_rate | start_date |
+-----+-----+-----+-----+-----+-----+
|      7 | K.S. DHALL | 500000 |      48 | NULL | 2016-03-05 |
|      1 | R.K. GUPTA | 300000 |      36 | 12.00 | 2015-07-19 |
|      6 | P. SHARMA | 700000 |      60 | 12.50 | 2015-06-05 |
|      5 | S.P. GARUN | 200000 |      36 | 12.50 | 2014-01-03 |
|      3 | K.P. JAIN | 300000 |      36 | NULL | 2013-03-08 |
|      4 | M.P. YADAV | 800000 |      60 | 10.00 | 2012-12-06 |
|      2 | P.Q. TIWARI | 500000 |      48 | 10.00 | 2012-03-22 |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

```
mysql> update loan_account
-> set int_rate=11.50
-> where int_rate is null;
```

```
Query OK, 2 rows affected (0.05 sec)
Rows matched: 2 Changed: 2 Warnings: 0
```

```
mysql> select*from loan_account;
```

accno	cust_name	loan_amt	installment	int_rate	start_date
1	R.K. GUPTA	300000	36	12.00	2015-07-19
2	P.Q. TIWARI	500000	48	10.00	2012-03-22
3	K.P. JAIN	300000	36	11.50	2013-03-08
4	M.P. YADAV	800000	60	10.00	2012-12-06
5	S.P. GARUN	200000	36	12.50	2014-01-03
6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	11.50	2016-03-05

```
7 rows in set (0.00 sec)
```

```
mysql> update loan_account
```

```
-> set loan_amt=loan_amt+7
```

```
-> where installment=36;
```

```
Query OK, 3 rows affected (0.39 sec)
```

```
Rows matched: 3 Changed: 3 Warnings: 0
```

```
mysql> select*from loan_account;
```

accno	cust_name	loan_amt	installment	int_rate	start_date
1	R.K. GUPTA	300007	36	12.00	2015-07-19
2	P.Q. TIWARI	500000	48	10.00	2012-03-22
3	K.P. JAIN	300007	36	11.50	2013-03-08
4	M.P. YADAV	800000	60	10.00	2012-12-06
5	S.P. GARUN	200007	36	12.50	2014-01-03
6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	11.50	2016-03-05

```
7 rows in set (0.00 sec)
```

```
mysql> alter table loan_account
```

```
-> modify cust_name varchar(35); #'MODIFY' only modifies definition, no brackets
```

```
Query OK, 7 rows affected (0.34 sec)
```

```
Records: 7 Duplicates: 0 Warnings: 0
```

```
mysql> desc loan_account;
```

Field	Type	Null	Key	Default	Extra
accno	int(11)	NO	PRI	NULL	
cust_name	varchar(35)	YES		NULL	
loan_amt	int(11)	YES		NULL	
installment	int(11)	YES		NULL	
int_rate	decimal(5,2)	YES		NULL	
start_date	date	YES		NULL	

```
6 rows in set (0.02 sec)
```

```
mysql> alter table loan_account
```

```
-> change installment insta integer; #'CHANGE' changes name
```

```
Query OK, 7 rows affected (0.03 sec)
```

Records: 7 Duplicates: 0 Warnings: 0

mysql> desc loan_account;

Field	Type	Null	Key	Default	Extra
accno	int(11)	NO	PRI	NULL	
cust_name	varchar(35)	YES		NULL	
loan_amt	int(11)	YES		NULL	
insta	int(11)	YES		NULL	
int_rate	decimal(5,2)	YES		NULL	
start_date	date	YES		NULL	

6 rows in set (0.02 sec)

mysql> select*from loan_account;

accno	cust_name	loan_amt	insta	int_rate	start_date
1	R.K. GUPTA	300007	36	12.00	2015-07-19
2	P.Q. TIWARI	500000	48	10.00	2012-03-22
3	K.P. JAIN	300007	36	11.50	2013-03-08
4	M.P. YADAV	800000	60	10.00	2012-12-06
5	S.P. GARUN	200007	36	12.50	2014-01-03
6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	11.50	2016-03-05

7 rows in set (0.00 sec)

mysql> alter table loan_account

-> change cust_name cust_name varchar(32);#changes only def

Query OK, 7 rows affected (0.06 sec)

Records: 7 Duplicates: 0 Warnings: 0

mysql> desc loan_account;

Field	Type	Null	Key	Default	Extra
accno	int(11)	NO	PRI	NULL	
cust_name	varchar(32)	YES		NULL	
loan_amt	int(11)	YES		NULL	
insta	int(11)	YES		NULL	
int_rate	decimal(5,2)	YES		NULL	
start_date	date	YES		NULL	

6 rows in set (0.00 sec)

mysql> select*from loan_account;

accno	cust_name	loan_amt	insta	int_rate	start_date
1	R.K. GUPTA	300007	36	12.00	2015-07-19
2	P.Q. TIWARI	500000	48	10.00	2012-03-22
3	K.P. JAIN	300007	36	11.50	2013-03-08
4	M.P. YADAV	800000	60	10.00	2012-12-06
5	S.P. GARUN	200007	36	12.50	2014-01-03

6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	11.50	2016-03-05

7 rows in set (0.00 sec)

```
mysql> alter table loan_account
-> change cust_name custname varchar(40); #2.'CHANGE' changes definition as well
```

Query OK, 7 rows affected (0.41 sec)
Records: 7 Duplicates: 0 Warnings: 0

```
mysql> desc loan_account;
```

Field	Type	Null	Key	Default	Extra
accno	int(11)	NO	PRI	NULL	
custname	varchar(40)	YES		NULL	
loan_amt	int(11)	YES		NULL	
insta	int(11)	YES		NULL	
int_rate	decimal(5,2)	YES		NULL	
start_date	date	YES		NULL	

6 rows in set (0.00 sec)

```
mysql> select*from loan_account;
```

accno	custname	loan_amt	insta	int_rate	start_date
1	R.K. GUPTA	300007	36	12.00	2015-07-19
2	P.Q. TIWARI	500000	48	10.00	2012-03-22
3	K.P. JAIN	300007	36	11.50	2013-03-08
4	M.P. YADAV	800000	60	10.00	2012-12-06
5	S.P. GARUN	200007	36	12.50	2014-01-03
6	P. SHARMA	700000	60	12.50	2015-06-05
7	K.S. DHALL	500000	48	11.50	2016-03-05

7 rows in set (0.00 sec)

```
mysql> alter table loan_account
```

```
-> add(cate char(1));
```

Query OK, 7 rows affected (0.41 sec)
Records: 7 Duplicates: 0 Warnings: 0

```
mysql> desc loan_account;
```

Field	Type	Null	Key	Default	Extra
accno	int(11)	NO	PRI	NULL	
custname	varchar(40)	YES		NULL	
loan_amt	int(11)	YES		NULL	
insta	int(11)	YES		NULL	
int_rate	decimal(5,2)	YES		NULL	
start_date	date	YES		NULL	
cate	char(1)	YES		NULL	

7 rows in set (0.01 sec)

```
mysql> select*from loan_account;
```

accno	custname	loan_amt	insta	int_rate	start_date	cate
1	R.K. GUPTA	300007	36	12.00	2015-07-19	NULL
2	P.Q. TIWARI	500000	48	10.00	2012-03-22	NULL
3	K.P. JAIN	300007	36	11.50	2013-03-08	NULL
4	M.P. YADAV	800000	60	10.00	2012-12-06	NULL
5	S.P. GARUN	200007	36	12.50	2014-01-03	NULL
6	P. SHARMA	700000	60	12.50	2015-06-05	NULL
7	K.S. DHALL	500000	48	11.50	2016-03-05	NULL

7 rows in set (0.00 sec)

Q-2

```
mysql> create database library;  
Query OK, 1 row affected (0.00 sec)
```

```
mysql> use library;  
Database changed
```

```
mysql> create table books  
-> (bookid varchar(5) primary key,  
-> bookname varchar(30) not null,  
-> authorname varchar(30),  
-> publisher varchar(30),  
-> price integer,  
-> type varchar(15),  
-> qty integer);
```

Query OK, 0 rows affected (0.36 sec)

```
mysql> create table issued  
-> (bookid varchar(5) unique not null,  
-> qty_issued integer,  
-> foreign key (bookid) references books(bookid));
```

Query OK, 0 rows affected (0.41 sec)

```
mysql> show tables;
```

Tables_in_library
books
issued

2 rows in set (0.05 sec)

```
mysql> desc books;
```

Field	Type	Null	Key	Default	Extra
bookid	varchar(5)	NO	PRI	NULL	
bookname	varchar(30)	NO		NULL	
authorname	varchar(30)	YES		NULL	
publisher	varchar(30)	YES		NULL	
price	int(11)	YES		NULL	

type	varchar(15)	YES		NULL	
qty	int(11)	YES		NULL	

7 rows in set (0.00 sec)

```
mysql> desc issued;
```

Field	Type	Null	Key	Default	Extra
bookid	varchar(5)	NO	PRI	NULL	
qty_issued	int(11)	YES		NULL	

2 rows in set (0.00 sec)

```
mysql> insert into books
```

```
-> values('C0001','Fast Cook','Lata Kapoor','EPB',355,'Cookery',5);
```

```
Query OK, 1 row affected (0.38 sec)
```

```
mysql> insert into books
```

```
-> values('F0001','The Tears','William Hopkins','F.ST PUBL',650,'Fiction',20);
```

```
Query OK, 1 row affected (0.01 sec)
```

```
mysql> insert into books
```

```
-> values('T0001','My First C++','Brian&Brooke','EPB',350,'Text',10);
```

```
Query OK, 1 row affected (0.36 sec)
```

```
mysql> insert into books
```

```
-> values('T0002','C++ Brainwork','AW Rossaine','TDH',350,'Text',15);
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into books(bookid,bookname,authorname,publisher,price,type)
```

```
-> values('F0002','Thunderbolts','Anna Roberts','F.ST PUBL',750,'Fiction');
```

```
Query OK, 1 row affected (0.30 sec)
```

```
mysql> select*from books;
```

bookid	bookname	authorname	publisher	price	type	qty
C0001	Fast Cook	Lata Kapoor	EPB	355	Cookery	5
F0001	The Tears	William Hopkins	F.ST PUBL	650	Fiction	20
F0002	Thunderbolts	Anna Roberts	F.ST PUBL	750	Fiction	NULL
T0001	My First C++	Brian&Brooke	EPB	350	Text	10
T0002	C++ Brainwork	AW Rossaine	TDH	350	Text	15

5 rows in set (0.00 sec)

```
mysql> insert into issued
```

```
-> values('T0001',4);
```

```
Query OK, 1 row affected (0.34 sec)
```

```
mysql> insert into issued
```

```
-> values('C0001',5);
```

```
Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into issued
```

```
-> values('F0001',8);
```

Query OK, 1 row affected (0.00 sec)

```
mysql> insert into issued
-> values('T0002',5);
```

Query OK, 1 row affected (0.03 sec)

```
mysql> insert into issued
-> values('F0001',3);
```

ERROR 1062 (23000): Duplicate entry 'F0001' for key 'bookid'

```
mysql> insert into issued
-> values('F0003',3);
```

ERROR 1452 (23000): Cannot add or update a child row: a foreign key constraint fails (`library`.`issued`, CONSTRAINT `issued_ibfk_1` FOREIGN KEY (`bookid`) REFERENCES `books` (`bookid`))

```
mysql> select*from issued;
```

bookid	qty_issued
C0001	5
F0001	8
T0001	4
T0002	5

4 rows in set (0.00 sec)

```
mysql> select bookname,authorname,price from books
-> where publisher='f.st publ';
```

bookname	authorname	price
The Tears	Willium Hopkins	650
Thunderbolts	Anna Roberts	750

2 rows in set (0.00 sec)

```
mysql> select bookname,price from books
-> order by price desc;
```

bookname	price
Thunderbolts	750
The Tears	650
Fast Cook	355
My First C++	350
C++ Brainwork	350

5 rows in set (0.00 sec)

```
mysql> select bookname,price from books
-> order by price asc;
```

bookname	price
My First C++	350

C++ Brainwork	350
Fast Cook	355
The Tears	650
Thunderbolts	750

+-----+

5 rows in set (0.00 sec)

mysql> update books

-> set price=price+75

-> where publisher='epb';

Query OK, 2 rows affected (0.08 sec)

Rows matched: 2 Changed: 2 Warnings: 0

mysql> select bookname,publisher,price from books;

+-----+

bookname	publisher	price
Fast Cook	EPB	430
The Tears	F.ST PUBL	650
Thunderbolts	F.ST PUBL	750
My First C++	EPB	425
C++ Brainwork	TDH	350

+-----+

5 rows in set (0.00 sec)

mysql> select A.bookid,bookname,B.qty_issued from books A,issued B

-> where A.bookid=B.bookid;

+-----+

bookid	bookname	qty_issued
C0001	Fast Cook	5
F0001	The Tears	8
T0001	My First C++	4
T0002	C++ Brainwork	5

+-----+

4 rows in set (0.37 sec)

mysql> select a.bookid,a.authername,b.qty_issued from books a,issued b

-> where a.bookid=b.bookid;

+-----+

bookid	authername	qty_issued
C0001	Lata Kapoor	5
F0001	William Hopkins	8
T0001	Brian&Brooke	4
T0002	AW Rossaine	5

+-----+

4 rows in set (0.00 sec)

mysql> delete from books

-> where bookid in(select bookid from issued where qty_issued=5);

ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`library`.`issued`, CONSTRAINT `issued_ibfk_1` FOREIGN KEY (`bookid`) REFERENCES `books` (`bookid`))

mysql> delete from issued

```
-> where bookid in(select bookid from books where publisher='epb');
Query OK, 2 rows affected (0.00 sec)
```

```
mysql> select*from issued;
```

bookid	qty_issued
F0001	8
T0002	5

```
2 rows in set (0.00 sec)
```

```
mysql> select*from books;
```

bookid	bookname	authorname	publisher	price	type	qty
C0001	Fast Cook	Lata Kapoor	EPB	430	Cookery	5
F0001	The Tears	William Hopkins	F.ST PUBL	650	Fiction	20
F0002	Thunderbolts	Anna Roberts	F.ST PUBL	750	Fiction	NULL
T0001	My First C++	Brian&Brooke	EPB	425	Text	10
T0002	C++ Brainwork	AW Rossaine	TDH	350	Text	15

```
5 rows in set (0.00 sec)
```

```
mysql> delete from books
```

```
-> where bookid='f0001';
```

```
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`library`.`issued`, CONSTRAINT `issued_ibfk_1` FOREIGN KEY (`bookid`) REFERENCES `books` (`bookid`))
```

```
mysql> delete from books
```

```
-> where bookid='f0002';
```

```
Query OK, 1 row affected (0.06 sec)
```

```
mysql> delete from books
```

```
-> where bookid='t0002';
```

```
ERROR 1451 (23000): Cannot delete or update a parent row: a foreign key constraint fails (`library`.`issued`, CONSTRAINT `issued_ibfk_1` FOREIGN KEY (`bookid`) REFERENCES `books` (`bookid`))
```

```
mysql> select*from books;
```

bookid	bookname	authorname	publisher	price	type	qty
C0001	Fast Cook	Lata Kapoor	EPB	430	Cookery	5
F0001	The Tears	William Hopkins	F.ST PUBL	650	Fiction	20
T0001	My First C++	Brian&Brooke	EPB	425	Text	10
T0002	C++ Brainwork	AW Rossaine	TDH	350	Text	15

```
4 rows in set (0.00 sec)
```