WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.

Theory:

Partitioned Join:

The relational tables that are to be joined gets partitioned on the joining attributes of both tables using same partitioning function to perform Join operation in parallel.

How does Partitioned Join work?

Assume that relational tables r and s need to be joined using attributes r.A and s.B. The system partitions both r and s using same partitioning technique into n partitions. In this process A and B attributes (joining attributes) to be used as partitioning attributes as well for r and s respectively. r is partitioned into r_0, r_1, r_2, …, r_{n-1} and s is partitioned into s_0, s_1, s_2, …, s_{n-1}. Then, the system sends partitions r_i and s_i into processor P_i, where the join is performed locally.as Equi-Joins and Natural Joins can be performed using Partitioned join technique .

Equi-Join or Natural Join is done between two tables using an equality condition such as r.A = s.B. The tuples which are satisfying this condition, i.e, same value for both A and B, are joined together. Others are discarded. Hence, if we partition the relations r and s on applying certain partitioning technique on both A and B, then the tuples having same value for A and B will end up in the same partition. Let us analyze this using simple example;

RegNo	SName	Gen	Phone
1	Sundar	M	9898786756
3	Karthik	M	8798987867
4	John	M	7898886756
2	Ram	M	9897786776

Table 1 - STUDENT

WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.

RegNo	Courses
4	Database
2	Database
3	Data Structures
1	Multimedia

Table 2 – COURSES_REGD

Let us assume the following;

The RegNo attributes of tables STUDENT and COURSES_REGD are used for joining.

Observe the order of tuples in both tables. They are not in particular order. They are stored in random order on RegNo.

Partition the tables on RegNo attribute using Hash Partition. We have 2 disks and we need to partition the relational tables into two partitions (possibly equal). Hence, n is 2.

The hash function is, h(RegNo) = (RegNo mod n) = (RegNo mod 2). And, if we apply the hash function we shall get the tables STUDENT and COURSES_REGD partitioned into Disk_0 and Disk_1 as stated below.

WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.

Partition 0

RegNo	SName	Gen	Phone
4	John	M	7898886756
2	Ram	M	9897786776

STUDENT_0

RegNo	Courses
4	Database
2	Database

COURSES_REGD_0

Partition 1

RegNo	SName	Gen	Phone
1	Sundar	M	9898786756
3	Karthik	M	8798987867

STUDENT_1

RegNo	Courses
3	Data Structures
1	Multimedia

COURSES_REGD_1

From the above table, it is very clear that the same RegNo values of both tables STUDENT and COURSES_REGD are sent to same partitions. Now, join can be performed locally at every processor in parallel.

One more interesting fact about this join is, only 4 (2 Student records X 2 Courses_regd records) comparisons need to be done in every partition for our example. Hence, we need total of 8 comparisons in partitioned join against 16 (4 X 4) in conventional join.

WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.

Program :

Join using Hash Partitioning on join Attribute

```
import mysql.connector
conn = mysql.connector.connect(host='localhost',
    database='exammanagement',
    user='root',
    password='root')
cur = conn.cursor()

def hashpartition(tablename, numberofpartitions):

        list_partition = range(numberofpartitions)

        for j in list_partition:
                cur.execute("DROP TABLE IF EXISTS " +tablename+ str(j))
                cur.execute("CREATE TABLE "+ tablename+ str(j) + " AS SELECT * FROM " + tablename + " WHERE dept_id % " + str(numberofpartitions) + " = " + str(j) + ";")
def jointable(numberofpartitions):
        list_partition = range(numberofpartitions)

        for j in list_partition:
                cur.execute("DROP TABLE IF EXISTS sub_dept"+ str(j))
                cur.execute("CREATE TABLE sub_dept"+ str(j) + " AS SELECT * FROM subject"+str(j) +" natural join department"+ str(j)+";")
def result(numberofpartitions):
        cur.execute("DROP TABLE IF EXISTS final")
        query = "CREATE TABLE final AS (SELECT * FROM sub_dept0"
        for j in range(1,numberofpartitions):
                query+=(" union SELECT * FROM sub_dept"+str(j))
        query+= " ) ;"
        cur.execute(query)

def display():
        #cur.execute("select * from final")
        cur.execute("SELECT * FROM final")

        # fetch all the matching rows
        result = cur.fetchall()
```

WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

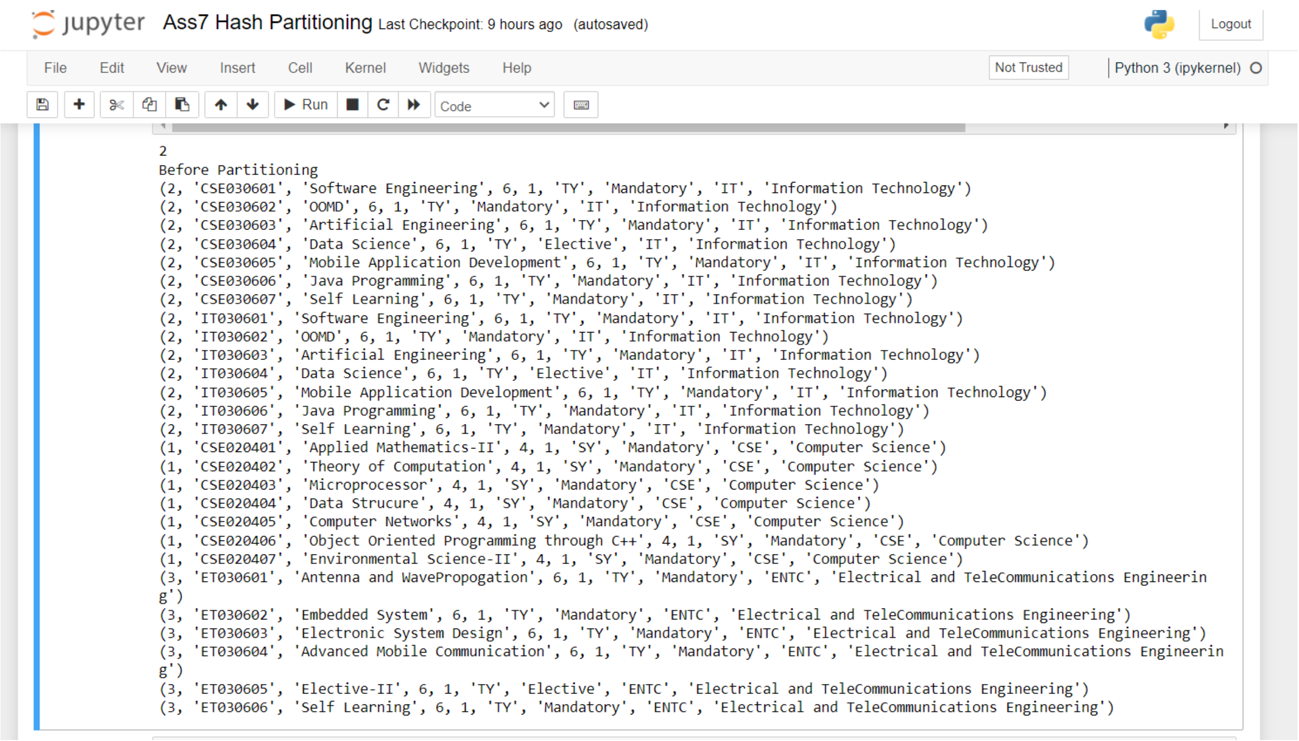
ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.

```
        print("Before Partitioning")

        # loop through the rows
        for row in result:
                print(row)

n=int(input())
hashpartition('subject',n)
hashpartition('department',n)
jointable(n)
result(n)
display()
```

Screenshots:

Join using Hash Partitioning on join Attribute



```
2
Before Partitioning
(2, 'CSE030601', 'Software Engineering', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'CSE030602', 'OOMD', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'CSE030603', 'Artificial Engineering', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'CSE030604', 'Data Science', 6, 1, 'TY', 'Elective', 'IT', 'Information Technology')
(2, 'CSE030605', 'Mobile Application Development', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'CSE030606', 'Java Programming', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'CSE030607', 'Self Learning', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'IT030601', 'Software Engineering', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'IT030602', 'OOMD', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'IT030603', 'Artificial Engineering', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'IT030604', 'Data Science', 6, 1, 'TY', 'Elective', 'IT', 'Information Technology')
(2, 'IT030605', 'Mobile Application Development', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'IT030606', 'Java Programming', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(2, 'IT030607', 'Self Learning', 6, 1, 'TY', 'Mandatory', 'IT', 'Information Technology')
(1, 'CSE020401', 'Applied Mathematics-II', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(1, 'CSE020402', 'Theory of Computation', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(1, 'CSE020403', 'Microprocessor', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(1, 'CSE020404', 'Data Strucure', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(1, 'CSE020405', 'Computer Networks', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(1, 'CSE020406', 'Object Oriented Programming through C++', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(1, 'CSE020407', 'Environmental Science-II', 4, 1, 'SY', 'Mandatory', 'CSE', 'Computer Science')
(3, 'ET030601', 'Antenna and WavePropogation', 6, 1, 'TY', 'Mandatory', 'ENTC', 'Electrical and TeleCommunications Engineerin
g')
(3, 'ET030602', 'Embedded System', 6, 1, 'TY', 'Mandatory', 'ENTC', 'Electrical and TeleCommunications Engineering')
(3, 'ET030603', 'Electronic System Design', 6, 1, 'TY', 'Mandatory', 'ENTC', 'Electrical and TeleCommunications Engineering')
(3, 'ET030604', 'Advanced Mobile Communication', 6, 1, 'TY', 'Mandatory', 'ENTC', 'Electrical and TeleCommunications Engineerin
g')
(3, 'ET030605', 'Elective-II', 6, 1, 'TY', 'Elective', 'ENTC', 'Electrical and TeleCommunications Engineering')
(3, 'ET030606', 'Self Learning', 6, 1, 'TY', 'Mandatory', 'ENTC', 'Electrical and TeleCommunications Engineering')
```

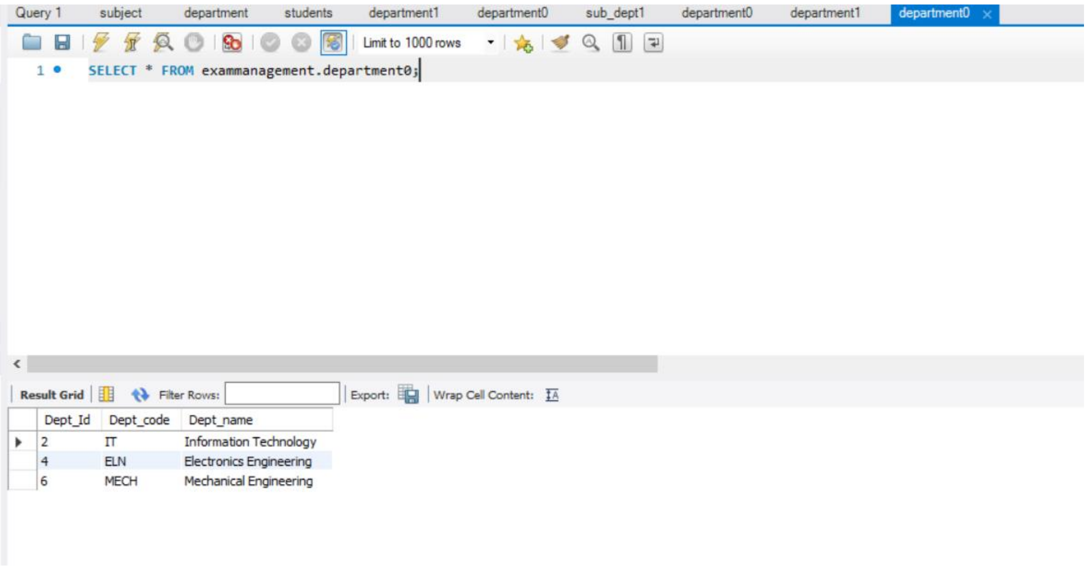
WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

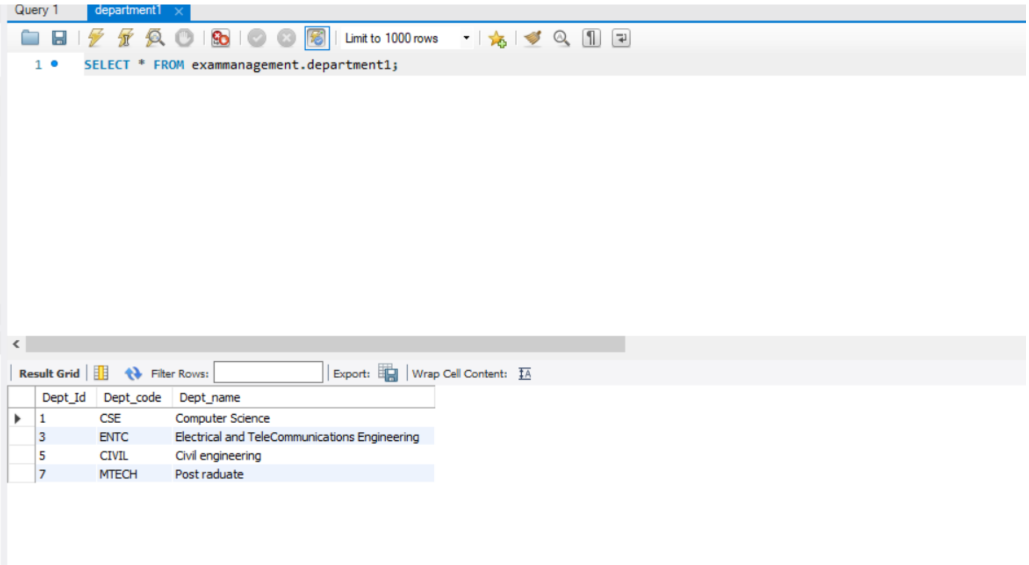
ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.



```
SELECT * FROM exammanagement.department0;
```

Dept_Id	Dept_code	Dept_name
2	IT	Information Technology
4	ELN	Electronics Engineering
6	MECH	Mechanical Engineering



```
SELECT * FROM exammanagement.department1;
```

Dept_Id	Dept_code	Dept_name
1	CSE	Computer Science
3	ENTC	Electrical and TeleCommunications Engineering
5	CIVIL	Civil engineering
7	MTECH	Post raduate

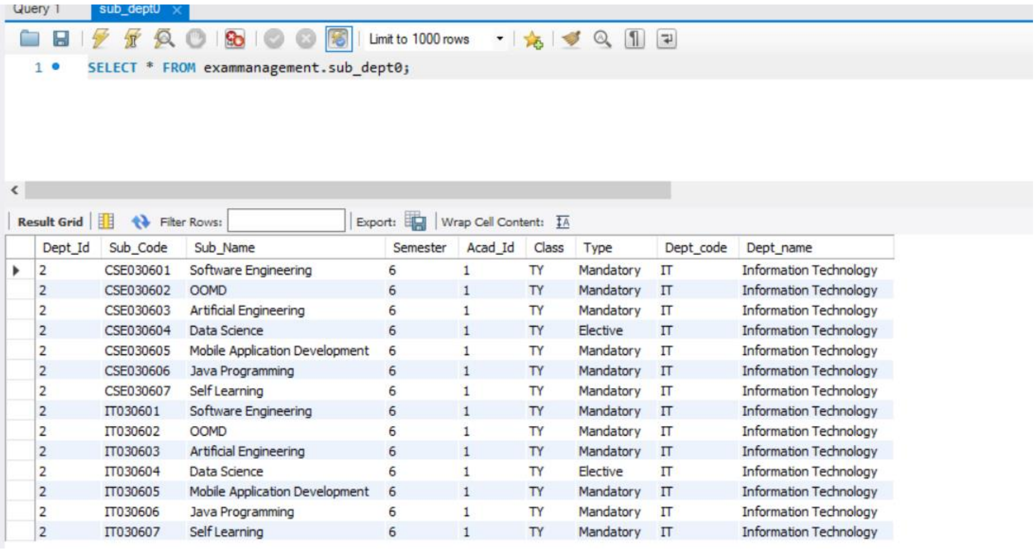
WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

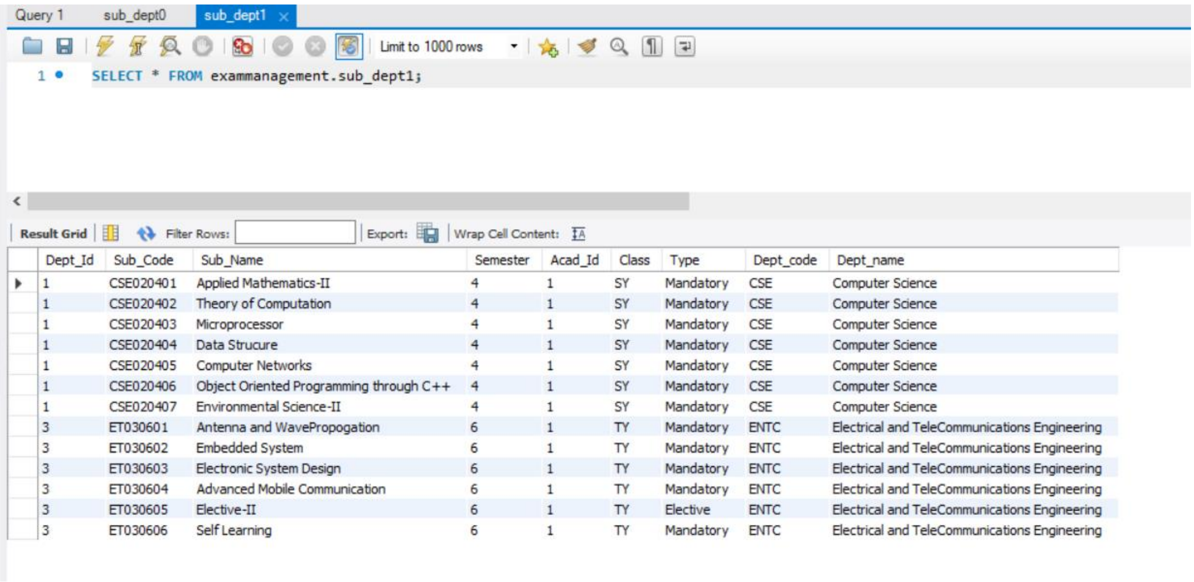
ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.



```
SELECT * FROM exammanagement.sub_dept0;
```

Dept_Id	Sub_Code	Sub_Name	Semester	Acad_Id	Class	Type	Dept_code	Dept_name
2	CSE030601	Software Engineering	6	1	TY	Mandatory	IT	Information Technology
2	CSE030602	OOMD	6	1	TY	Mandatory	IT	Information Technology
2	CSE030603	Artificial Engineering	6	1	TY	Mandatory	IT	Information Technology
2	CSE030604	Data Science	6	1	TY	Elective	IT	Information Technology
2	CSE030605	Mobile Application Development	6	1	TY	Mandatory	IT	Information Technology
2	CSE030606	Java Programming	6	1	TY	Mandatory	IT	Information Technology
2	CSE030607	Self Learning	6	1	TY	Mandatory	IT	Information Technology
2	IT030601	Software Engineering	6	1	TY	Mandatory	IT	Information Technology
2	IT030602	OOMD	6	1	TY	Mandatory	IT	Information Technology
2	IT030603	Artificial Engineering	6	1	TY	Mandatory	IT	Information Technology
2	IT030604	Data Science	6	1	TY	Elective	IT	Information Technology
2	IT030605	Mobile Application Development	6	1	TY	Mandatory	IT	Information Technology
2	IT030606	Java Programming	6	1	TY	Mandatory	IT	Information Technology
2	IT030607	Self Learning	6	1	TY	Mandatory	IT	Information Technology



```
SELECT * FROM exammanagement.sub_dept1;
```

Dept_Id	Sub_Code	Sub_Name	Semester	Acad_Id	Class	Type	Dept_code	Dept_name
1	CSE020401	Applied Mathematics-II	4	1	SY	Mandatory	CSE	Computer Science
1	CSE020402	Theory of Computation	4	1	SY	Mandatory	CSE	Computer Science
1	CSE020403	Microprocessor	4	1	SY	Mandatory	CSE	Computer Science
1	CSE020404	Data Strucure	4	1	SY	Mandatory	CSE	Computer Science
1	CSE020405	Computer Networks	4	1	SY	Mandatory	CSE	Computer Science
1	CSE020406	Object Oriented Programming through C++	4	1	SY	Mandatory	CSE	Computer Science
1	CSE020407	Environmental Science-II	4	1	SY	Mandatory	CSE	Computer Science
3	ET030601	Antenna and WavePropogation	6	1	TY	Mandatory	ENTC	Electrical and TeleCommunications Engineering
3	ET030602	Embedded System	6	1	TY	Mandatory	ENTC	Electrical and TeleCommunications Engineering
3	ET030603	Electronic System Design	6	1	TY	Mandatory	ENTC	Electrical and TeleCommunications Engineering
3	ET030604	Advanced Mobile Communication	6	1	TY	Mandatory	ENTC	Electrical and TeleCommunications Engineering
3	ET030605	Elective-II	6	1	TY	Elective	ENTC	Electrical and TeleCommunications Engineering
3	ET030606	Self Learning	6	1	TY	Mandatory	ENTC	Electrical and TeleCommunications Engineering

Program :

Join using Range Partitioning on join Attribute

```python
import mysql.connector
conn = mysql.connector.connect(host='localhost',
                       database='miniexammanagement',
                       user='root',
                       password='root')
```

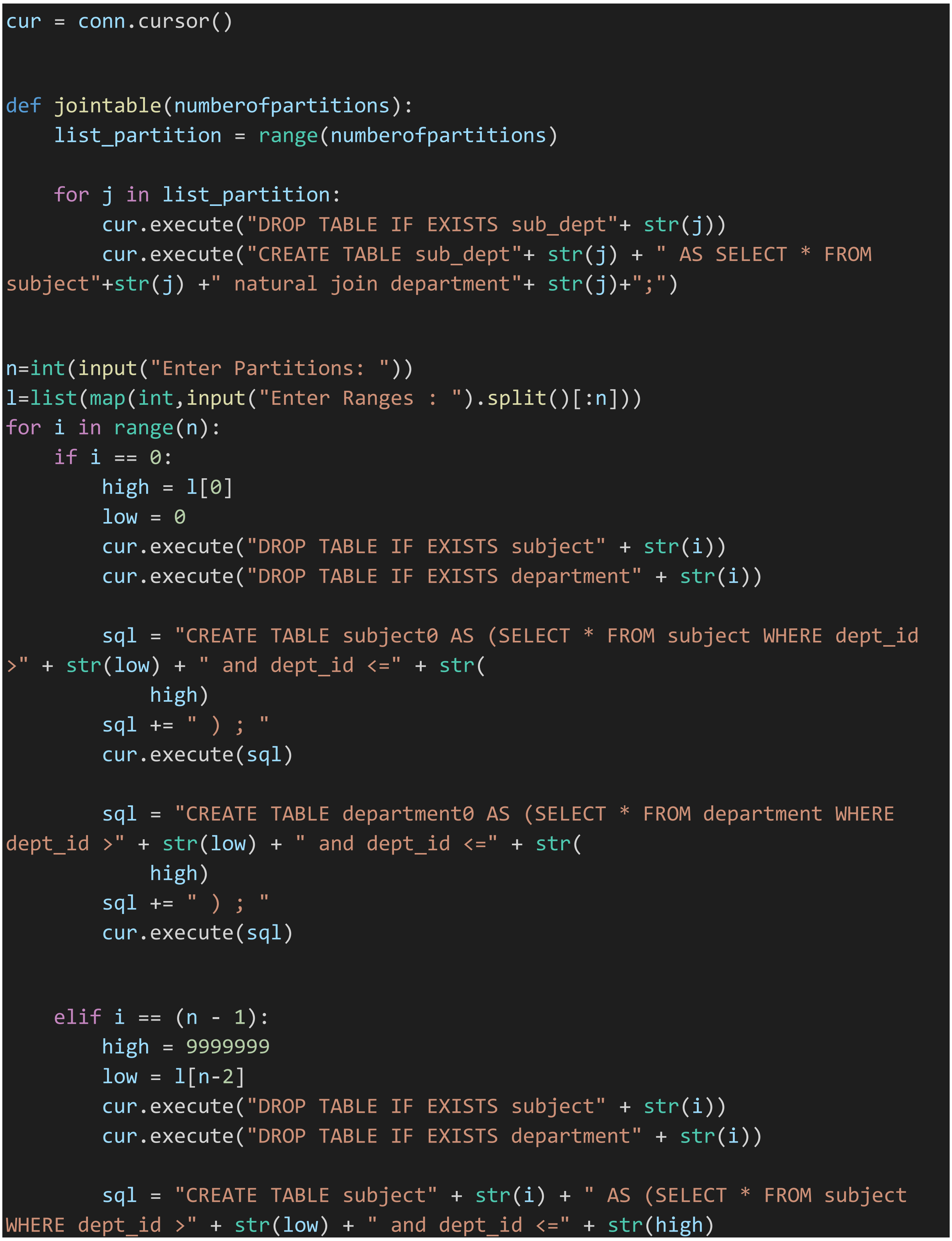
WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.

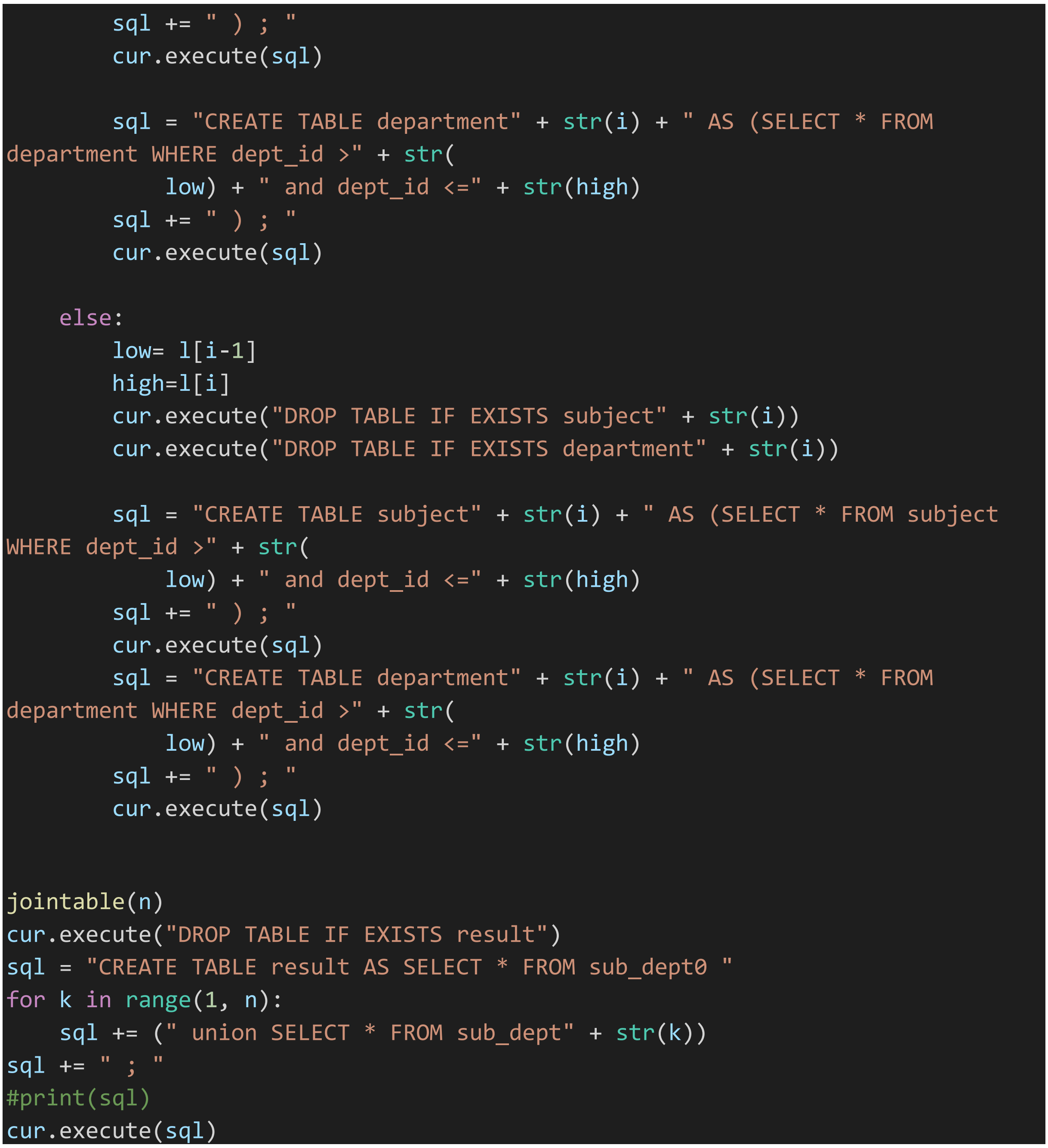


```
cur = conn.cursor()


def jointable(numberofpartitions):
    list_partition = range(numberofpartitions)

    for j in list_partition:
        cur.execute("DROP TABLE IF EXISTS sub_dept"+ str(j))
        cur.execute("CREATE TABLE sub_dept"+ str(j) + " AS SELECT * FROM 
subject"+str(j) +" natural join department"+ str(j)+";")


n=int(input("Enter Partitions: "))
l=list(map(int,input("Enter Ranges : ").split()[:n]))
for i in range(n):
    if i == 0:
        high = l[0]
        low = 0
        cur.execute("DROP TABLE IF EXISTS subject" + str(i))
        cur.execute("DROP TABLE IF EXISTS department" + str(i))

        sql = "CREATE TABLE subject0 AS (SELECT * FROM subject WHERE dept_id 
>" + str(low) + " and dept_id <=" + str(
            high)
        sql += " ) ; "
        cur.execute(sql)

        sql = "CREATE TABLE department0 AS (SELECT * FROM department WHERE 
dept_id >" + str(low) + " and dept_id <=" + str(
            high)
        sql += " ) ; "
        cur.execute(sql)


    elif i == (n - 1):
        high = 9999999
        low = l[n-2]
        cur.execute("DROP TABLE IF EXISTS subject" + str(i))
        cur.execute("DROP TABLE IF EXISTS department" + str(i))

        sql = "CREATE TABLE subject" + str(i) + " AS (SELECT * FROM subject 
WHERE dept_id >" + str(low) + " and dept_id <=" + str(high)
```

Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.



```
        sql += " ) ; "
        cur.execute(sql)

        sql = "CREATE TABLE department" + str(i) + " AS (SELECT * FROM 
department WHERE dept_id >" + str(
            low) + " and dept_id <=" + str(high)
        sql += " ) ; "
        cur.execute(sql)

    else:
        low= l[i-1]
        high=l[i]
        cur.execute("DROP TABLE IF EXISTS subject" + str(i))
        cur.execute("DROP TABLE IF EXISTS department" + str(i))

        sql = "CREATE TABLE subject" + str(i) + " AS (SELECT * FROM subject 
WHERE dept_id >" + str(
            low) + " and dept_id <=" + str(high)
        sql += " ) ; "
        cur.execute(sql)
        sql = "CREATE TABLE department" + str(i) + " AS (SELECT * FROM 
department WHERE dept_id >" + str(
            low) + " and dept_id <=" + str(high)
        sql += " ) ; "
        cur.execute(sql)


jointable(n)
cur.execute("DROP TABLE IF EXISTS result")
sql = "CREATE TABLE result AS SELECT * FROM sub_dept0 "
for k in range(1, n):
    sql += (" union SELECT * FROM sub_dept" + str(k))
sql += " ; "
#print(sql)
cur.execute(sql)
```

WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I

Advanced Database System

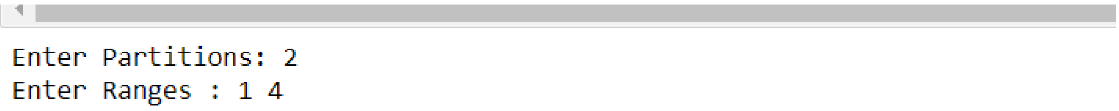
Name: Alaikya S Yemul Roll No: 62

ASSIGNMENT NO: 7

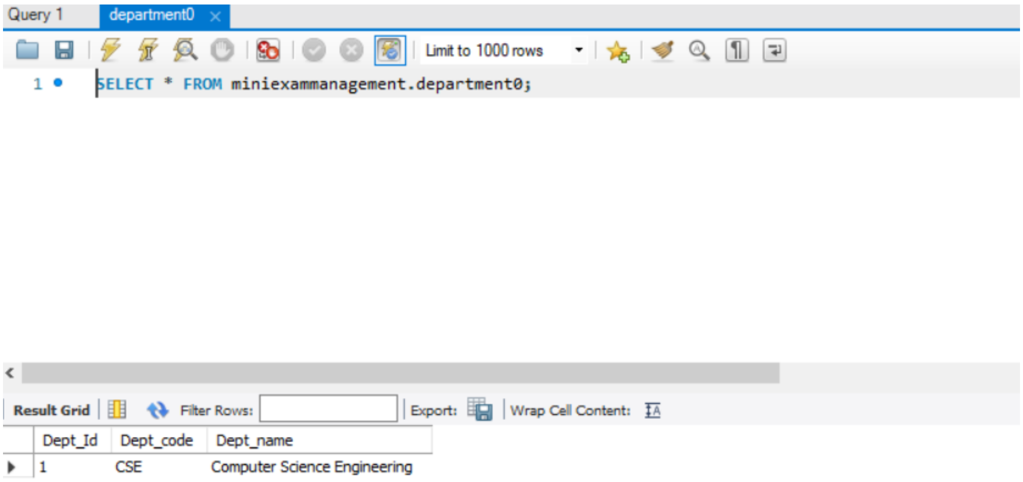
Title: Implement join operation on n relations using parallelism approach.

Screenshots :

Join using Range Partitioning on join Attribute

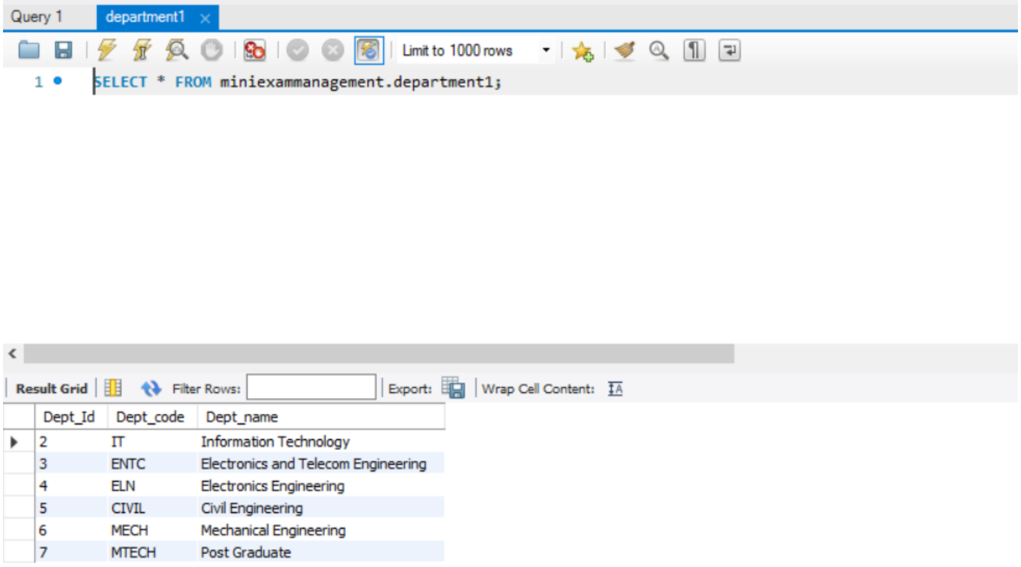


```
Enter Partitions: 2
Enter Ranges : 1 4
```



```
SELECT * FROM miniexammanagement.department0;
```

Dept_Id	Dept_code	Dept_name
1	CSE	Computer Science Engineering



```
SELECT * FROM miniexammanagement.department1;
```

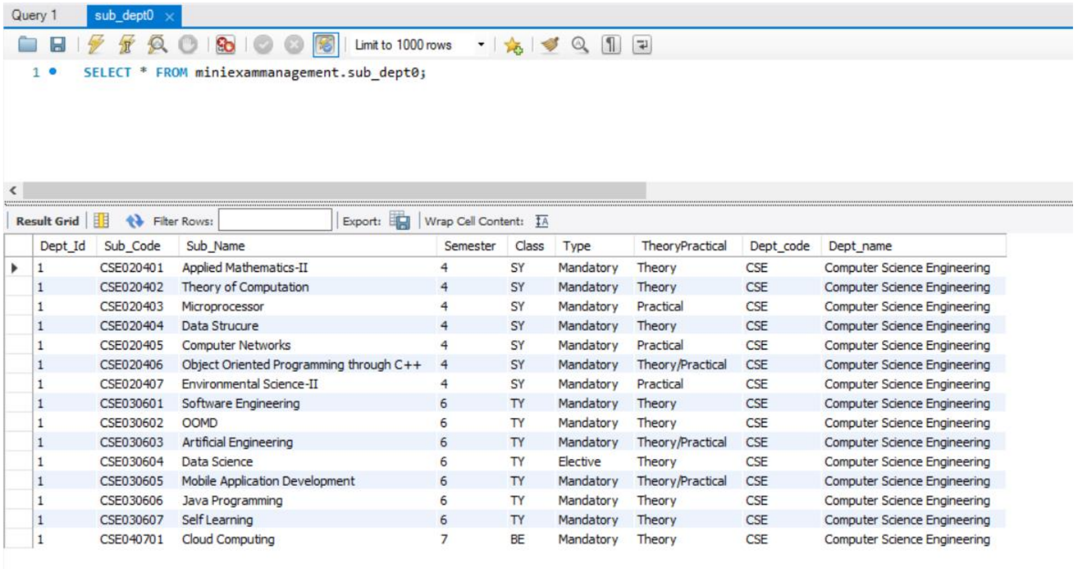
Dept_Id	Dept_code	Dept_name
2	IT	Information Technology
3	ENTC	Electronics and Telecom Engineering
4	ELN	Electronics Engineering
5	CIVIL	Civil Engineering
6	MECH	Mechanical Engineering
7	MTECH	Post Graduate

WALCHAND INSTITUTE OF TECHNOLOGY, SOLAPUR
INFORMATION TECHNOLOGY
2021-22 SEMESTER –I
Advanced Database System

Name: Alaikya S Yemul Roll No: 62

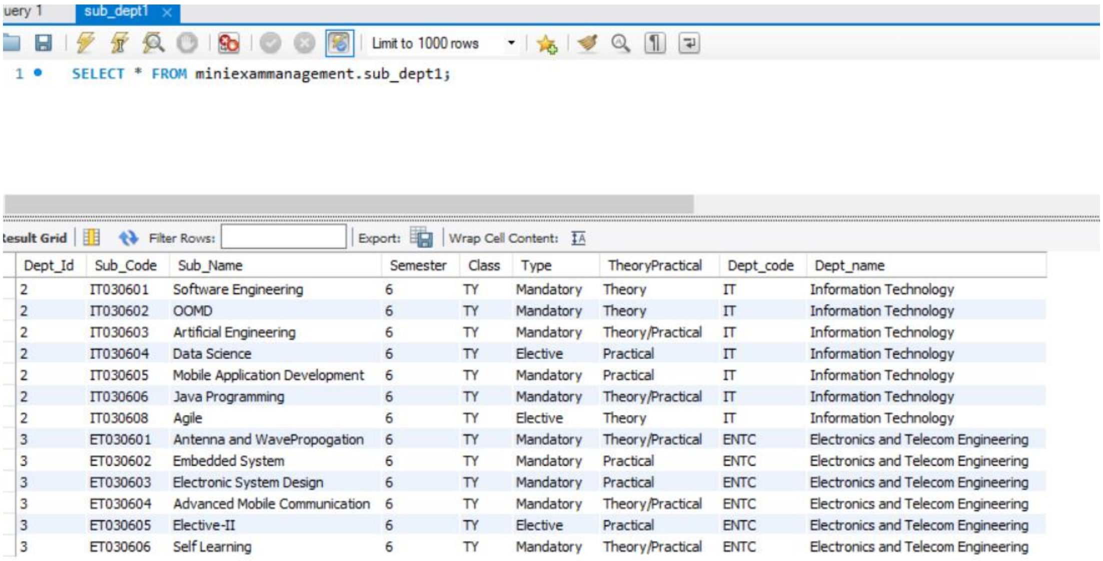
ASSIGNMENT NO: 7

Title: Implement join operation on n relations using parallelism approach.



```
SELECT * FROM miniexammanagement.sub_dept0;
```

Dept_Id	Sub_Code	Sub_Name	Semester	Class	Type	TheoryPractical	Dept_code	Dept_name
1	CSE020401	Applied Mathematics-II	4	SY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE020402	Theory of Computation	4	SY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE020403	Microprocessor	4	SY	Mandatory	Practical	CSE	Computer Science Engineering
1	CSE020404	Data Strucure	4	SY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE020405	Computer Networks	4	SY	Mandatory	Practical	CSE	Computer Science Engineering
1	CSE020406	Object Oriented Programming through C++	4	SY	Mandatory	Theory/Practical	CSE	Computer Science Engineering
1	CSE020407	Environmental Science-II	4	SY	Mandatory	Practical	CSE	Computer Science Engineering
1	CSE030601	Software Engineering	6	TY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE030602	OOMD	6	TY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE030603	Artificial Engineering	6	TY	Mandatory	Theory/Practical	CSE	Computer Science Engineering
1	CSE030604	Data Science	6	TY	Elective	Theory	CSE	Computer Science Engineering
1	CSE030605	Mobile Application Development	6	TY	Mandatory	Theory/Practical	CSE	Computer Science Engineering
1	CSE030606	Java Programming	6	TY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE030607	Self Learning	6	TY	Mandatory	Theory	CSE	Computer Science Engineering
1	CSE040701	Cloud Computing	7	BE	Mandatory	Theory	CSE	Computer Science Engineering



```
SELECT * FROM miniexammanagement.sub_dept1;
```

Dept_Id	Sub_Code	Sub_Name	Semester	Class	Type	TheoryPractical	Dept_code	Dept_name
2	IT030601	Software Engineering	6	TY	Mandatory	Theory	IT	Information Technology
2	IT030602	OOMD	6	TY	Mandatory	Theory	IT	Information Technology
2	IT030603	Artificial Engineering	6	TY	Mandatory	Theory/Practical	IT	Information Technology
2	IT030604	Data Science	6	TY	Elective	Practical	IT	Information Technology
2	IT030605	Mobile Application Development	6	TY	Mandatory	Practical	IT	Information Technology
2	IT030606	Java Programming	6	TY	Mandatory	Theory/Practical	IT	Information Technology
2	IT030608	Agile	6	TY	Elective	Theory	IT	Information Technology
3	ET030601	Antenna and WavePropogation	6	TY	Mandatory	Theory/Practical	ENTC	Electronics and Telecom Engineering
3	ET030602	Embedded System	6	TY	Mandatory	Practical	ENTC	Electronics and Telecom Engineering
3	ET030603	Electronic System Design	6	TY	Mandatory	Practical	ENTC	Electronics and Telecom Engineering
3	ET030604	Advanced Mobile Communication	6	TY	Mandatory	Theory/Practical	ENTC	Electronics and Telecom Engineering
3	ET030605	Elective-II	6	TY	Elective	Practical	ENTC	Electronics and Telecom Engineering
3	ET030606	Self Learning	6	TY	Mandatory	Theory/Practical	ENTC	Electronics and Telecom Engineering