

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

Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

Theory:

Data Replication is the process of storing data in more than one site or node. It is useful in improving the availability of data. It is simply copying data from a database from one server to another server so that all the users can share the same data without any inconsistency. The result is a distributed database in which users can access data relevant to their tasks without interfering with the work of others. There can be full replication, in which the whole database is stored at every site. There can also be partial replication, in which some frequently used fragment of the database are replicated and others are not replicated.

Types of Data Replication –

1. **Transactional Replication** – In Transactional replication users receive full initial copies of the database and then receive updates as data changes. Data is copied in real time from the publisher to the receiving database in the same order as they occur with the publisher therefore in this type of replication, transactional consistency is guaranteed. Transactional replication is typically used in server-to-server environments. It does not simply copy the data changes, but rather consistently and accurately replicates each change.
2. **Snapshot Replication** – Snapshot replication distributes data exactly as it appears at a specific moment in time does not monitor for updates to the data. The entire snapshot is generated and sent to Users. Snapshot replication is generally used when data changes are infrequent. It is bit slower than transactional because on each attempt it moves multiple records from one end to the other end. Snapshot replication is a good way to perform initial synchronization between the publisher and the subscriber.
3. **Merge Replication** – Data from two or more databases is combined into a single database. Merge replication is the most complex type of

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

Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

replication because it allows both publisher and subscriber to independently make changes to the database. Merge replication is typically used in server-to-client environments.

Program :

```
import mysql.connector
import tkinter as tk
from tkinter import *
def fullreplication():
    my_w3 = tk.Tk()
    my_w3.title("Duplicate Table Fragment1")
    my_w3.geometry("600x200")
    my_connect3 = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="root",
        database="student"
    )
    my_conn3 = my_connect3.cursor()
    my_conn3.execute("""CREATE TABLE IF NOT EXISTS studentdup (
                        Rollno integer PRIMARY KEY,
                        Name varchar(45),
                        Address varchar(45),
                        Mobileno varchar(45)
                    )""")
    ##### end of connection #####
    my_conn3.execute("INSERT INTO studentdup (Rollno,Name,Address,Mobileno)
    SELECT Rollno,Name,Address,Mobileno FROM students where Rollno<=63 ")
    my_conn3.execute("SELECT * FROM studentdup")
    i=0
    for student in my_conn3:
        for j in range(len(student)):
            e = Entry(my_w3, width=25, fg='blue')
            e.grid(row=i, column=j)
            e.insert(END, student[j])
        i=i+1
```

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

Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

```
my_w4 = tk.Tk()
my_w4.title("Duplicate Table Fragment2")
my_w4.geometry("600x200")
my_connect4 = mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="root",
    database="student"
)
my_conn4 = my_connect4.cursor()
my_conn4.execute("""CREATE TABLE IF NOT EXISTS studentdup (
                    Rollno integer PRIMARY KEY,
                    Name varchar(45),
                    Address varchar(45),
                    Mobilenno varchar(45)
                    )""")
##### end of connection #####
my_conn4.execute("INSERT INTO studentdup (Rollno,Name,Address,Mobilenno)
SELECT Rollno,Name,Address,Mobilenno FROM students where Rollno>63 ")
my_conn4.execute("SELECT * FROM studentdup")
i=0
for student in my_conn4:
    for j in range(len(student)):
        e = Entry(my_w4, width=25, fg='blue')
        e.grid(row=i, column=j)
        e.insert(END, student[j])
    i=i+1
my_w4.mainloop()
my_w3.mainloop()

def display():
    my_w = tk.Tk()
    my_w.title("Original Table")
    my_w.geometry("600x200")
    my_connect = mysql.connector.connect(
        host="localhost",
        user="root",
        passwd="root",
        database="student"
    )
    my_conn = my_connect.cursor()
```

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

Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

```
##### end of connection #####
my_conn.execute("SELECT * FROM students")
i=0
for student in my_conn:
    for j in range(len(student)):
        e = Entry(my_w, width=25, fg='blue')
        e.grid(row=i, column=j)
        e.insert(END, student[j])
    i=i+1

my_w1 = tk.Tk()
my_w1.title("Fragment1")
my_w1.geometry("600x200")
my_connect1 = mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="root",
    database="student"
)
my_conn1 = my_connect1.cursor()
##### end of connection #####
my_conn1.execute("SELECT * FROM students where Rollno<=63")
i=0
for student in my_conn1:
    for j in range(len(student)):
        e = Entry(my_w1, width=25, fg='blue')
        e.grid(row=i, column=j)
        e.insert(END, student[j])
    i=i+1

my_w2 = tk.Tk()
my_w2.title("Fragment2")
my_w2.geometry("600x200")
my_connect2 = mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="root",
    database="student"
)
```

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

Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

```
my_conn2 = my_connect2.cursor()
##### end of connection #####
my_conn2.execute("SELECT * FROM students where Rollno>63")
i=0
for student in my_conn2:
    for j in range(len(student)):
        e = Entry(my_w2, width=25, fg='blue')
        e.grid(row=i, column=j)
        e.insert(END, student[j])
    i=i+1
my_w2.mainloop()
my_w1.mainloop()

my_w.mainloop()

root = Tk()
root.title("Full Replication")
root.geometry("600x400")
lbl = Label(root, text="Full Replication", font=("Times New Roman Bold", 20))
lbl.grid(column=25, row=20)
#lbl.configure(background="#306754")
btn = Button(root, text="Display Tables", font=("Arial", 10), command=display)
btn.grid(column=30, row=30)
btn = Button(root, text="Full
Replication", font=("Arial", 10), command=fullreplication)
btn.grid(column=50, row=30)
root['bg'] = '#CBC3E3'
root.mainloop()
```

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

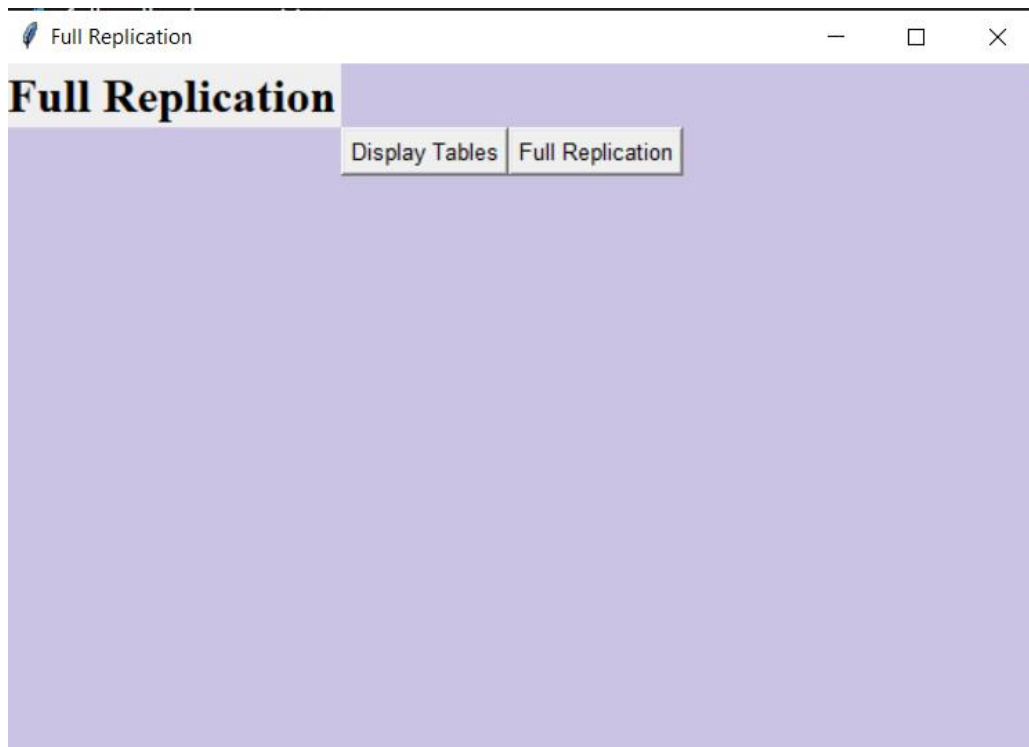
Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

Screenshots:



Original Table			
14	Ishita	Delhi	9856457854
19	Madhuri	Solapur	9685321478
62	Alaikya	Solapur	8956471258
63	Ishita	Solapur	7845123698
64	Pujitha	Solapur	8546213975

Fragment1			
14	Ishita	Delhi	9856457854
19	Madhuri	Solapur	9685321478
62	Alaikya	Solapur	8956471258
63	Ishita	Solapur	7845123698

Fragment2			
64	Pujitha	Solapur	8546213975

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

Name: Alaikya S Yemul

Roll No: 62

ASSIGNMENT NO: 2

Title: Implement Full Replication for created fragments.

Duplicate Table Fragment1			
14	Ishita	Delhi	9856457854
19	Madhuri	Solapur	9685321478
62	Alaikya	Solapur	8956471258
63	Ishita	Solapur	7845123698
Duplicate Table Fragment2			
64	Pujitha	Solapur	8546213975