Theory :

**Fragmentation :**

Fragmentation is the task of dividing a table into a set of smaller tables

these subsets of the table are called Fragments.

Fragmentation should be done in a way so that the original table can be

reconstructed from the fragments whenever required this is called “re

constructiveness.”

Fragmentation can be of three types: Horizontal, Vertical, and Mixed

(Combination of horizontal and vertical).

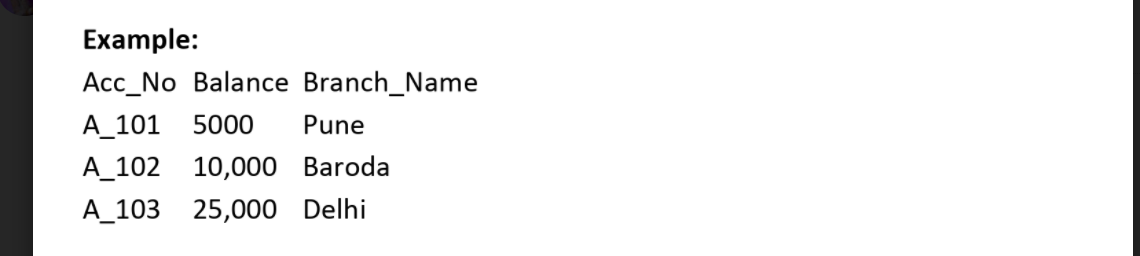
**1. Vertical Fragmentation:**

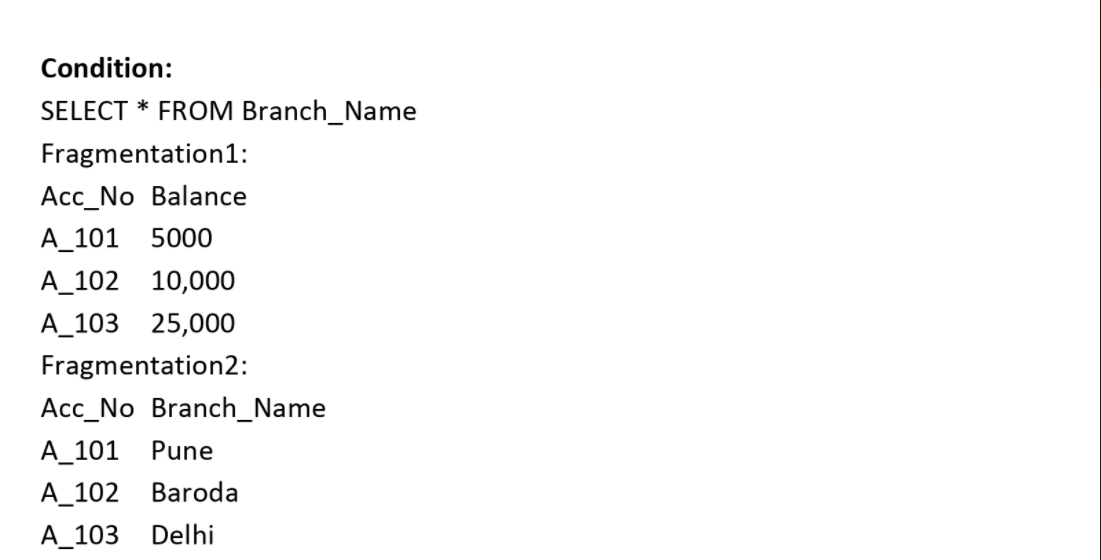
In vertical fragmentation, the fields or columns of a table are grouped into

fragments. In order to maintain reconstructiveness, each fragment should

contain the primary key field(s) of the table. Vertical fragmentation can be

used to enforce privacy of data.





**2. Horizontal Fragmentation:**

Horizontal fragmentation groups the tuples of a table to values of one or

more fields. Horizontal fragmentation should also confirm to the rule of re

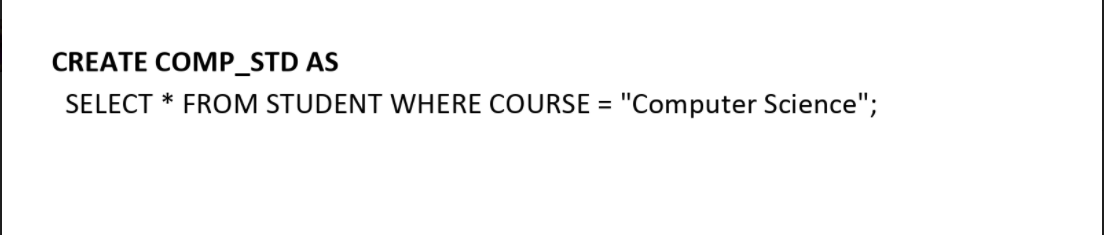
constructiveness. Each horizontal fragment must have all columns of the

original base table.

Example: In the student schema, if the details of all students of Computer

Science Course needs to be maintained at the School of Computer Science,

then the designer will horizontally fragment the database as follows –



**3. Mixed Fragmentation:**

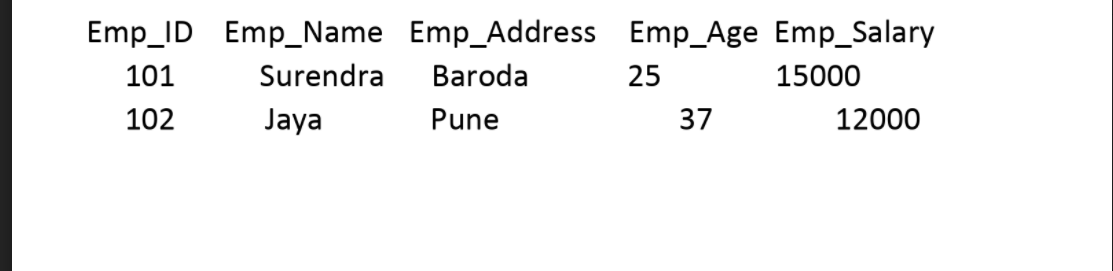
Mixed fragmentation can be achieved by performing horizontal

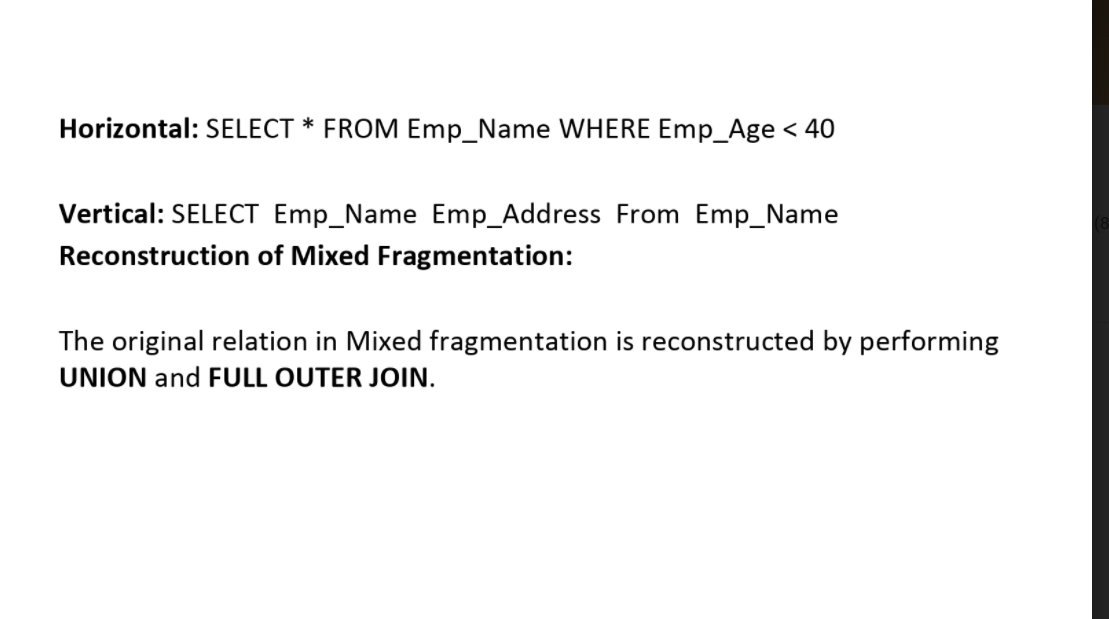
and vertical partition together.

Mixed fragmentation is group of rows and columns in relation.

Example: Consider the following table which consists of employee

information.





Program Code :

import mysql.connector

import tkinter  as tk

from tkinter import \*

def vertical():

    my\_w = tk.Tk()

    my\_w.title("Fragment1")

    my\_w.geometry("600x200")

    my\_connect = mysql.connector.connect(

        host="localhost",

        user="root",

        passwd="root",

        database="student"

    )

    my\_conn = my\_connect.cursor()

####### end of connection ####

    my\_conn.execute("SELECT Rollno,Name,Address FROM students limit 0,10")

    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("Fragment2")

    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 Rollno,Mobileno FROM students limit 0,10")

    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\_w1.mainloop()

    my\_w.mainloop()

def horizontal():

    my\_w = tk.Tk()

    my\_w.title("Fragment1")

    my\_w.geometry("600x200")

    my\_connect = mysql.connector.connect(

        host="localhost",

        user="root",

        passwd="root",

        database="student"

    )

    my\_conn = my\_connect.cursor()

####### end of connection ####

    my\_conn.execute("SELECT \* FROM students where Rollno<=63")

    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("Fragment2")

    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\_w1.mainloop()

    my\_w.mainloop()

def mixed():

    my\_w = tk.Tk()

    my\_w.title("Horizontal")

    my\_w.geometry("600x200")

    my\_connect = mysql.connector.connect(

        host="localhost",

        user="root",

        passwd="root",

        database="student"

    )

    my\_conn = my\_connect.cursor()

####### end of connection ####

    my\_conn.execute("SELECT \* FROM students WHERE Rollno<63")

    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("Mixed Table1")

    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 Rollno,Name,Address 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("Mixed Table2")

    my\_w2.geometry("600x200")

    my\_connect2 = mysql.connector.connect(

        host="localhost",

        user="root",

        passwd="root",

        database="student"

    )

    my\_conn2 = my\_connect2.cursor()

####### end of connection ####

    my\_conn2.execute("SELECT Rollno,Mobileno FROM students WHERE Rollno<64")

    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("Fragmentation")

root.geometry("580x200")

lbl = Label(root, text="Fragmentation", font=("Times New Roman Bold", 20))

lbl.grid(column=4, row=5)

lbl.configure(background="#306754")

menubar = Menu(root)

filemenu = Menu(menubar, tearoff=0)

filemenu.add\_command(label="Vertical Fragmentation", command=vertical)

filemenu.add\_command(label="Horizontal Fragmentation", command=horizontal)

filemenu.add\_command(label="Mixed Fragmentation", command=mixed)

filemenu.add\_separator()

filemenu.add\_command(label="Exit", command=root.quit)

menubar.add\_cascade(label="Menu", menu=filemenu)

editmenu = Menu(menubar, tearoff=0)

editmenu.add\_separator()

root['bg'] = '#306754'

root.config(menu=menubar)

root.mainloop()

Screenshots :

