Assignment 2

Assignment description

Individually or in a group of at most two, create a distributed heterogeneous database environment comprising three sites with three different participating database platforms, and at least two different operating systems.

Use the above environment to demonstrate your grasp of fragmentation and reconstruction. Think of a domain area comprising of at least four distributed relations. Write out your applications/reports and use them to perform fragmentation. Write out the applications/reports into calculus queries. Come up with appropriate query access frequencies of your choice. After working out the fragmentation, allocate the fragments by implementing them physically in the participating databases. Choose one of the sites to be the decision site and perform reconstruction using either views, functions, stored procedures or any other technique.

Submission outputs

- A video recording of at most seven (7 minutes). First few seconds of video to have your registration number(s) displayed
- A pdf document summarizing the work. Include your registration number(s) on the first page

Group members

1. Wisdom Makokha – P15/81777/2017
2. Joel Ndeda - P15/103081/2017

Project process

Containers were used to simulate the operating systems and to satisfy the capability of the databases running on different operating systems.

A container provides an isolated, consistent environment that can mimic the behavior of running applications on separate OS instances.

Containers also make the installation and running of the targeted

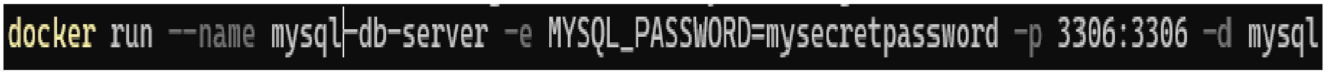
So we used containers to simulate the environment of the different operating systems.

Container setup

The technologies used for the system were

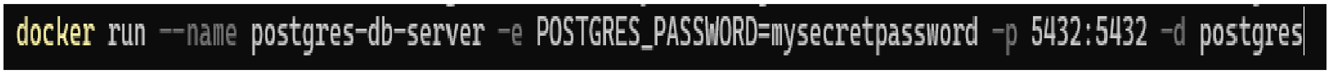
1. MySQL used on site 1.
2. Postgres SQL used on site 2.
3. SQL Server used on site 3.

MySQL server container setup command:



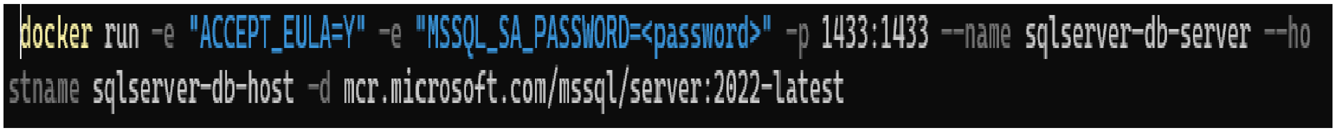
```
docker run --name mysql-db-server -e MYSQL_PASSWORD=mysecretpassword -p 3306:3306 -d mysql
```

Postgres server container setup command



```
docker run --name postgres-db-server -e POSTGRES_PASSWORD=mysecretpassword -p 5432:5432 -d postgres
```

SQL Server container setup command:



```
docker run -e "ACCEPT_EULA=Y" -e "MSSQL_SA_PASSWORD=<password>" -p 1433:1433 --name sqlserver-db-server --hostname sqlserver-db-host -d mcr.microsoft.com/mssql/server:2022-latest
```

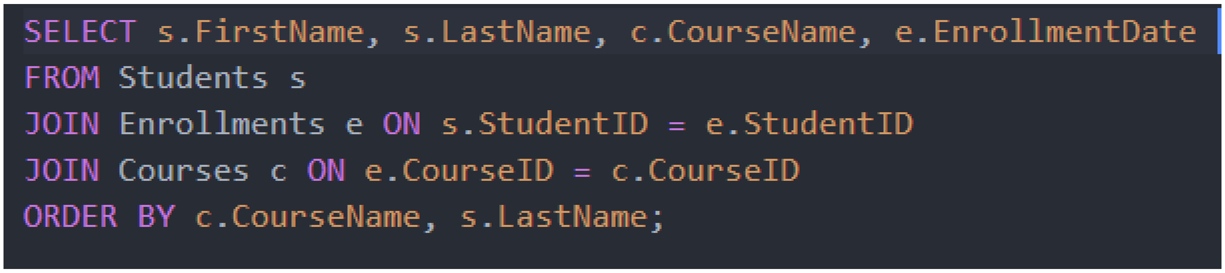
Database design

TABLE_NAME	COLUMN_NAME	DATA_TYPE
Courses	CourseID	int
	CourseName	varchar
	Department	varchar
	Credits	int
Enrollments	EnrollmentID	int
	StudentID	int
	CourseID	int
	EnrollmentDate	date
	Grade	varchar
Professors	ProfessorID	int
	FirstName	varchar
	LastName	varchar
	Department	varchar
Students	StudentID	int
	FirstName	varchar
	LastName	varchar
	DateOfBirth	date
	Gender	varchar
	Major	varchar

Applications / reports

The applications, their equivalent calculus query and a sample of the results from each report.

1. Generate a report of all students enrolled in each course



```
SELECT s.FirstName, s.LastName, c.CourseName, e.EnrollmentDate
FROM Students s
JOIN Enrollments e ON s.StudentID = e.StudentID
JOIN Courses c ON e.CourseID = c.CourseID
ORDER BY c.CourseName, s.LastName;
```

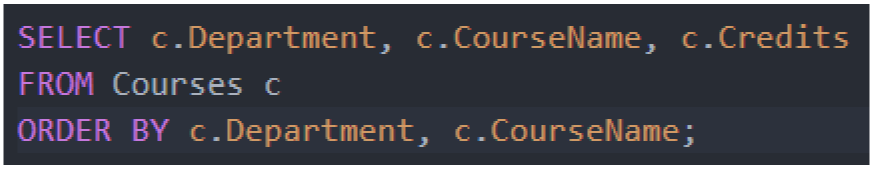
FirstName	LastName	CourseName	EnrollmentDate
Ben	Gusikowski	Abstract algebra	12/08/2018
Rodrigo	Gutkowski	Abstract algebra	25/06/2004
Vernie	Mills	Abstract algebra	21/06/1976
Barbara	Olson	Abstract algebra	20/08/1993
Will	Stracke	Abstract algebra	24/03/1996
Magnus	Berge	Biochemistry	10/09/1971
Marco	Koss	Biochemistry	27/05/2006
Darwin	Lemke	Biochemistry	01/02/1997
Will	Stracke	Biochemistry	30/08/1998
Dena	Windler	Biochemistry	06/09/2024
Rico	Bartell	Calculus 1	08/07/2007
Valentin	Denesik	Calculus 1	28/12/1977
Ben	Gusikowski	Calculus 1	26/06/1998
Daryl	Kuhlman	Calculus 1	27/09/1974
Vinnie	Mills	Calculus 1	05/05/2017
Harley	Ortiz	Calculus 1	08/02/1990
Dianna	Jakubowski	Cell Biology	17/06/2004
Darwin	Lemke	Cell Biology	06/04/1992
Oran	Leuschke	Cell Biology	18/10/1975
Bart	Marks	Cell Biology	20/08/1999

Fragmentation.

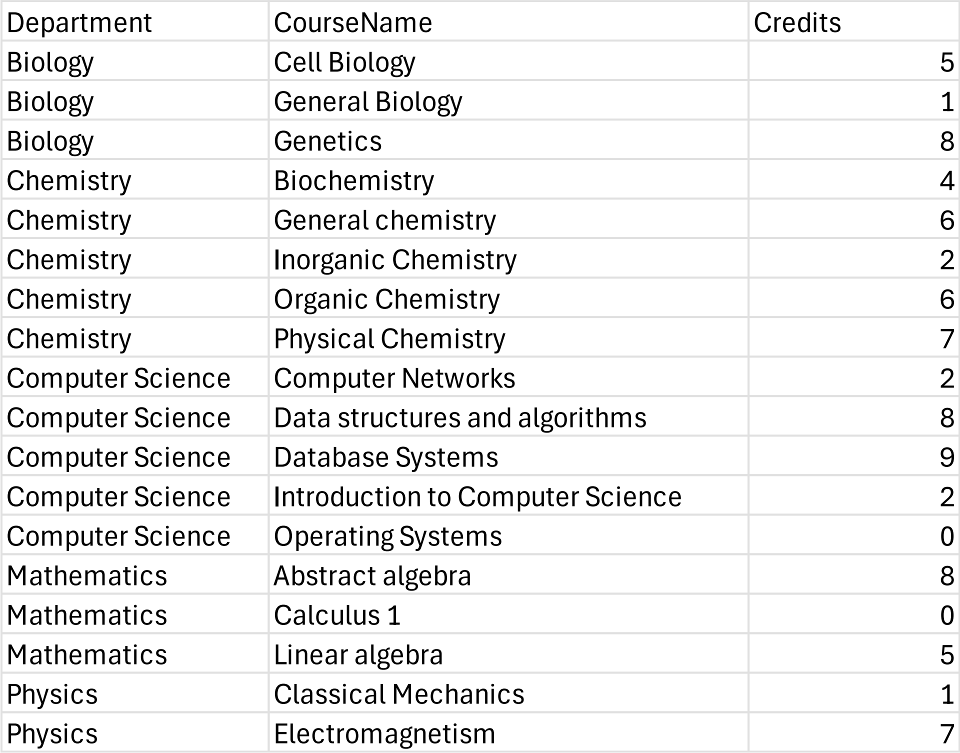
Horizontal fragmentation of Courses table by department.

Vertical fragmentation of Students table to separate student enrollment details from student personal details

2. List all courses offered by each department, including course names and credits



```sql
SELECT c.Department, c.CourseName, c.Credits
FROM Courses c
ORDER BY c.Department, c.CourseName;
```



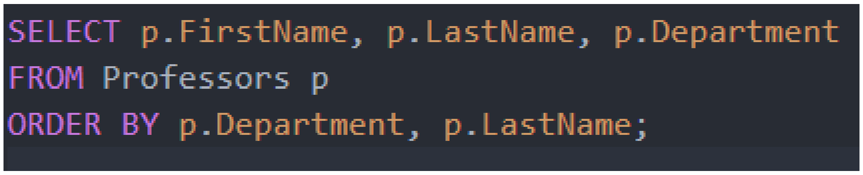
Department	CourseName	Credits
Biology	Cell Biology	5
Biology	General Biology	1
Biology	Genetics	8
Chemistry	Biochemistry	4
Chemistry	General chemistry	6
Chemistry	Inorganic Chemistry	2
Chemistry	Organic Chemistry	6
Chemistry	Physical Chemistry	7
Computer Science	Computer Networks	2
Computer Science	Data structures and algorithms	8
Computer Science	Database Systems	9
Computer Science	Introduction to Computer Science	2
Computer Science	Operating Systems	0
Mathematics	Abstract algebra	8
Mathematics	Calculus 1	0
Mathematics	Linear algebra	5
Physics	Classical Mechanics	1
Physics	Electromagnetism	7

Fragmentation.

Horizontal fragmentation of courses table by department

Vertical fragmentation of courses table to separate credits details from other course details.

3. Generate a report of all professors and their respective departments



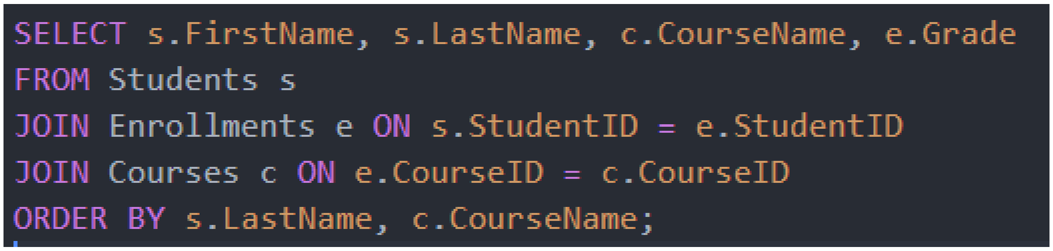
```sql
SELECT p.FirstName, p.LastName, p.Department
FROM Professors p
ORDER BY p.Department, p.LastName;
```

FirstName	LastName	Department
Keely	Ortiz	Biology
Einar	Haag	Chemistry
Rowena	Rogahn	Chemistry
Salvatore	Wuckert	Chemistry
Icie	Cummings	Computer Science
Luigi	Dare	Computer Science
Jayden	Keebler	Computer Science
Brandi	Mills	Mathematics
Aditya	Sawayn	Mathematics
Kevon	Kshlerin	Physics

Fragmentation.

Vertical fragmentation of professors table to separate professor information from departmental information

4. Generate a report of student grades in each course



```
SELECT s.FirstName, s.LastName, c.CourseName, e.Grade
FROM Students s
JOIN Enrollments e ON s.StudentID = e.StudentID
JOIN Courses c ON e.CourseID = c.CourseID
ORDER BY s.LastName, c.CourseName;
```

FirstName	LastName	CourseName	Grade
Erich	Adams	Classical Mechanics	b
Erich	Adams	General Biology	a
Erich	Adams	Quanutm Mechanics	e
Marc	Armstrong	Introduction to Computer Science	d
Marc	Armstrong	Organic Chemistry	a
Rico	Bartell	Calculus 1	b
Brain	Bauch	Operating Systems	e
Magnus	Berge	Biochemistry	d
Kelvin	Bernier	Inorganic Chemistry	c
Kelvin	Bernier	Linear algebra	e
Kelvin	Bernier	Operating Systems	a
Jacinthe	Borer	Data structures and algorithms	d
Juwan	Christiansen	Operating Systems	e
Fleta	Cremin	Computer Networks	c
Fleta	Cremin	Quanutm Mechanics	b
Haven	Daugherty	General Biology	e
Valentin	Denesik	Calculus 1	e
Valentin	Denesik	Computer Networks	c

Fragmentation.

Vertical fragmentation of student table to separate enrollment details from personal student details.

Access frequencies

	Report 1	Report 2	Report 3	Report 4
Site 1(MySQL)	40	10	25	25
Site 2 (SQL Server)	15	50	15	20
Site 3 (Postgres)	40	10	10	40

Site 1

- Horizontal fragment students by their major for a specific group of majors - PHF
- Store enrollment dates - VF
- Store basic professor details – VF
- Store course credits - VF

Site 2

- Fragments from the course table separating the table horizontally by department – PHF
- General student info
- Store professor department information - VF

Site 3

- Horizontal fragmentation of students by major for a specific group of majors– PHF
- Horizontal fragmentation of enrollments by major - PHF

Actual Fragment distribution

Site	Fragments
Site 1	F1 Student Major Science
	F2 Course Credits
	F3 Enrollment Dates
	F4 Professor Basic Details
Site 2	F1 Student General Info
	F2 Courses Computer Science
	F3 Courses Mathematics
	F4 Courses Physics
	F5 Courses Biology
	F6 Courses Chemistry
	F7 Professors Department
Site 3	F1 Student Computer Math
	F2 Science Enrollments
	F3 Enrollment Computer Math

Decision site

Site 1 was selected as the decision site to perform reconstruction