



## **Assignment # 3**

**Presented By**

**M.Waqas (22I - 2469)**

**Abdullah Mansoor (22I - 8808)**

**To**

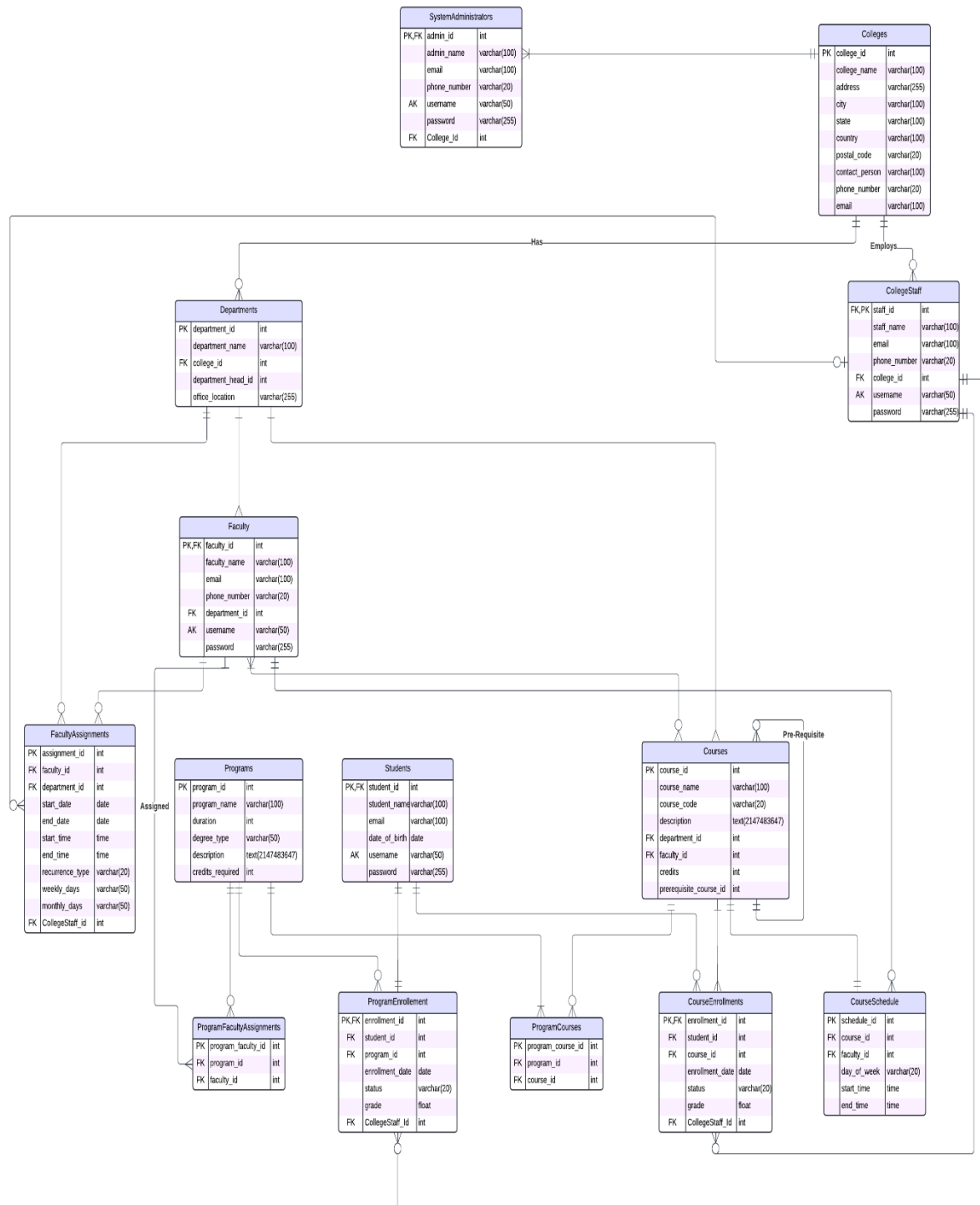
**Sir Bilal**

## Table of Contents

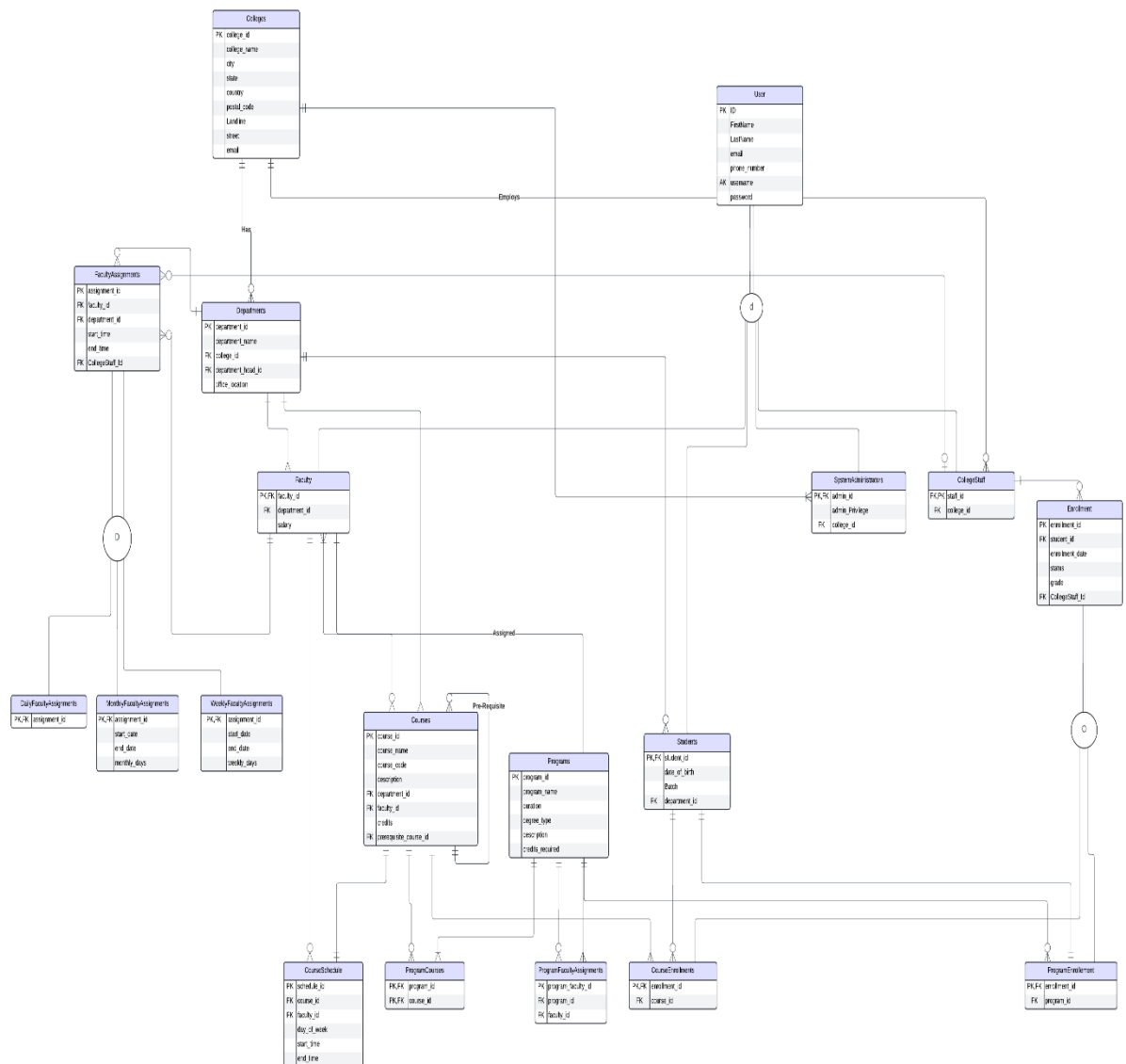
1 Question#1 .....	3
1.1 Entity Relation Diagram.....	3
1.2 Enhanced Entity Relation Diagram.....	4
1.3 Relational Data Model .....	5
1.4 Queries .....	6
2 Question#2 .....	9
2.1 Examples for Anomalies .....	9
2.1.1 Insertion Anomaly .....	9
2.1.2 Deletion Anomaly.....	9
2.1.3 Update Anomaly .....	10
2.2 Normalization.....	10

# 1 Question#1

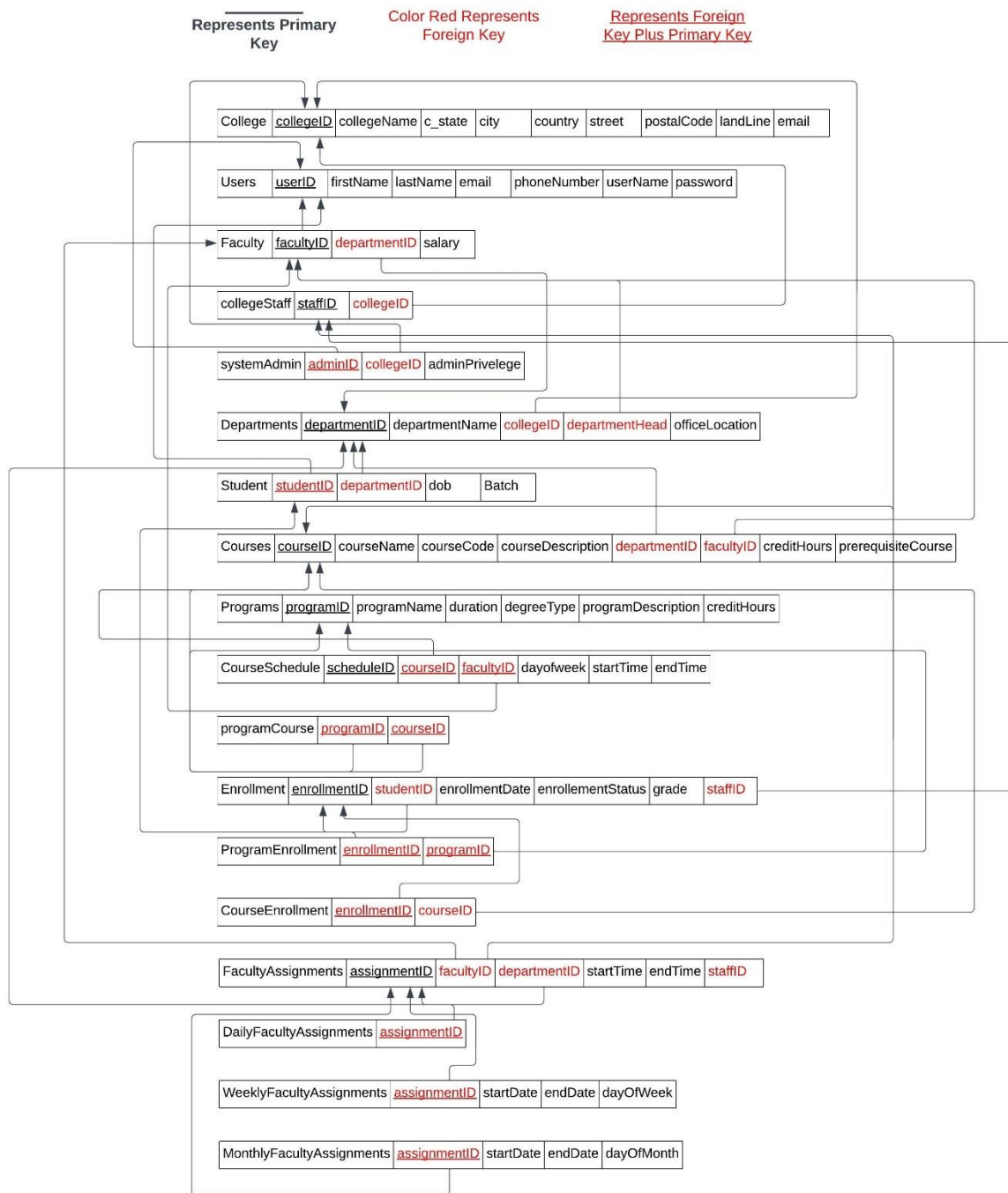
## 1.1 Entity Relation Diagram



## 1.2 Enhanced Entity Relation Diagram



## 1.3 Relational Data Model



## 1.4 Queries

Our Queries:

SQLQuery2.sql - WIKI\SQLEXPRESS.D83 (WIKI\wik8 (54)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

Connect - WIKI\SQLEXPRESS (SQL Server 16.0.1000 - WIKI\WIKI8 (54))

SQLQuery2.sql - WIKI\SQLEXPRESS.D83 (WIKI\wik8 (54))

```

SELECT * FROM Colleges;
SELECT * FROM Departments;

```

Results Messages

first_name	last_name	college_name
Alice	Johnson	Harbor College
Bob	Williams	Greenwood College
Carol	Brown	Hill Valley University
David	Jones	Riverside Institute
Eva	Miller	East Coast College
Frank	Davis	Northern University
Grace	Garcia	Southern College
Henry	Wilson	West Coast University

user_id	first_name	last_name	date_of_birth
6	David	Jones	2000-09-20
7	Eva	Miller	2001-10-22
8	Frank	Davis	2002-11-30
9	Grace	Garcia	2003-12-14
10	Henry	Wilson	2004-01-16

department_id	faculty_id	salary
1	101	75000.00
2	102	70000.00
3	103	72000.00
4	104	68000.00
5	105	65000.00
6	106	62000.00
7	107	61000.00
8	108	63000.00

course_id	faculty_id	start_time	end_time
1	101	09:00:00.0000000	11:00:00.0000000
5	102	09:00:00.0000000	11:00:00.0000000

department_id	faculty_id	salary
10	110	66000.00
9	109	64000.00
8	108	63000.00

Query executed successfully.

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time: 00:00:00.213

Finish time: 24/04/2024 23:48:24

Name: WIKI\SQLEXPRESS

Rows returned: 92

Start time: 24/04/2024 23:48:24

State: Open

Connection

Connection name: WIKI\SQLEXPRESS (WIKI\wik8 (54))

Connection Details

Connection elapsed: 00:00:00.213

Connection encrypt: Encrypted

Connection finish: 24/04/2024 23:48:24

Connection rows: 92

Connection start: 24/04/2024 23:48:24

Connection state: Open

Display name: WIKI\SQLEXPRESS

Login name: WIKI\wik8

Server name: WIKI\SQLEXPRESS

Server version: 16.0.1000

Session Tracing ID: 54

TDS protocol version: 0x74000004

Name: The name of the connection.

Snipping Tool

Screenshot copied to clipboard and saved. Select here to mark up and share.

SQLQuery2.sql - WIKI\SQLEXPRESS.D83 (WIKI\wik8 (54)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

Connect - WIKI\SQLEXPRESS (SQL Server 16.0.1000 - WIKI\WIKI8 (54))

SQLQuery2.sql - WIKI\SQLEXPRESS.D83 (WIKI\wik8 (54))

```

SELECT * FROM Colleges;
SELECT * FROM Departments;

```

Results Messages

department_name	student_count
Business Administration	2
Civil Engineering	2
Computer Science	2
Electrical Engineering	2
Mechanical Engineering	2

course_name	credits
Introduction to Programming	3
Algorithms	3
Data Structures	3
Machine Learning	4

user_id	first_name	last_name	email	phone_number	username	password
1	John	Doe	john.doe@example.com	555-1234	john.doe	newsecurepassword
2	Jane	Smith	jane.smith@example.com	555-9999	jane.smith	password456
3	Alice	Johnson	alice.johnson@example.com	555-9012	alice.joh	password789
4	Bob	Williams	bob.williams@example.com	555-3456	bob.will	password012
5	Carol	Brown	carol.brown@example.com	555-7890	carol.b	password345
6	David	Jones	david.jones@example.com	555-1235	david.j	password678
7	Eva	Miller	eva.miller@example.com	555-5679	eva.mill	password901
8	Frank	Davis	frank.davis@example.com	555-9013	frank.d	password234

first_name	last_name	college_name
John	Doe	Tech University
Jane	Smith	Metro State College
Alice	Johnson	Harbor College
Bob	Williams	Greenwood College
Carol	Brown	Hill Valley University
David	Jones	Riverside Institute

Query executed successfully.

Properties

Current connection parameters

Aggregate Status

Connection failures

Elapsed time: 00:00:00.213

Finish time: 24/04/2024 23:48:24

Name: WIKI\SQLEXPRESS

Rows returned: 92

Start time: 24/04/2024 23:48:24

State: Open

Connection

Connection name: WIKI\SQLEXPRESS (WIKI\wik8 (54))

Connection Details

Connection elapsed: 00:00:00.213

Connection encrypt: Encrypted

Connection finish: 24/04/2024 23:48:24

Connection rows: 92

Connection start: 24/04/2024 23:48:24

Connection state: Open

Display name: WIKI\SQLEXPRESS

Login name: WIKI\wik8

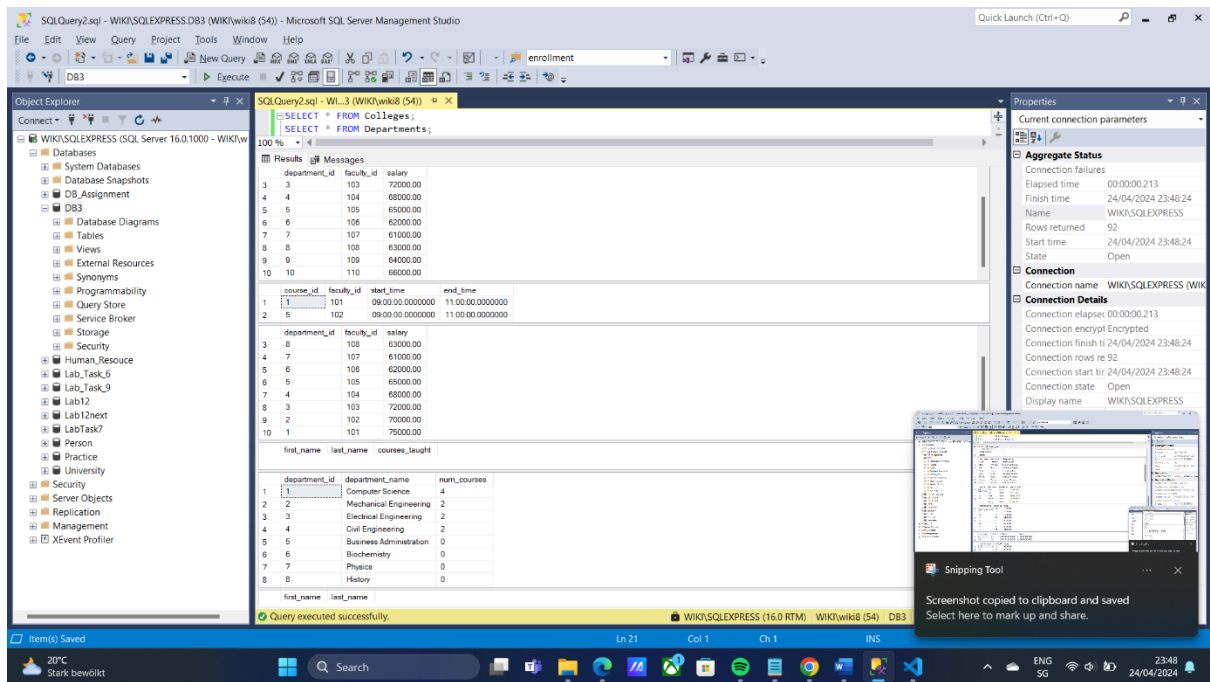
Server name: WIKI\SQLEXPRESS

Server version: 16.0.1000

Session Tracing ID: 54

TDS protocol version: 0x74000004

Name: The name of the connection.



Results Messages

	first_name	last_name	college_name	course_name	department_name	enrollment_date	
3	Alice	Johnson	Tech University	Data Structures	Computer Science	2022-08-25	
4	Bob	Williams	Tech University	Machine Learning	Computer Science	2022-08-25	
5	Carol	Brown	Tech University	Thermodynamics	Mechanical Engineering	2022-08-25	
6	David	Jones	Tech University	Fluid Mechanics	Mechanical Engineering	2022-08-25	
7	Eva	Miller	Metro State College	Circuit Analysis	Electrical Engineering	2022-08-25	
8	Frank	Davis	Metro State College	Electronics	Electrical Engineering	2022-08-25	
9	Grace	Garcia	Metro State College	Statics	Civil Engineering	2022-08-25	
10	Henry	Wilson	Metro State College	Dynamics	Civil Engineering	2022-08-25	

	course_name	course_code	description	department_name	college_name	day_of_week	start_time	end_time	assignment_type
1	Introduction to Programming	CS101	Learn the basics of programming.	Computer Science	Tech University	Monday	09:00:00.0000000	11:00:00.0000000	1
2	Algorithms	CS201	Study of algorithms	Computer Science	Tech University	Wednesday	10:00:00.0000000	12:00:00.0000000	1
3	Data Structures	CS202	Organizing data for efficient access and modificati...	Computer Science	Tech University	Friday	13:00:00.0000000	15:00:00.0000000	1
4	Machine Learning	CS303	Introduction to machine learning concepts and al...	Computer Science	Tech University	Tuesday	14:00:00.0000000	16:00:00.0000000	1

	first_name	last_name	course_name	description	enrollment_date	grade	Assignment_Frequency
1	John	Doe	Introduction to Programming	Learn the basics of programming.	2022-08-25	A	Daily
2	Jane	Smith	Algorithms	Study of algorithms	2022-08-25	B	Daily

	course_name	day_of_week	start_time	end_time	faculty_id	department_name	college_name
1	Dynamics	Thursday	12:00:00.0000000	14:00:00.0000000	104	Civil Engineering	Metro State College
2	Statics	Thursday	09:00:00.0000000	11:00:00.0000000	104	Civil Engineering	Metro State College
3	Circuit Analy...	Varies	12:00:00.0000000	14:00:00.0000000	103	Electrical Engine...	Metro State College
4	Electronics	Varies	12:00:00.0000000	14:00:00.0000000	103	Electrical Engine...	Metro State College
5	Algorithms	Wednesday	10:00:00.0000000	12:00:00.0000000	101	Computer Science	Tech University
6	Algorithms	Varies	08:00:00.0000000	10:00:00.0000000	101	Computer Science	Tech University
7	Data Structu...	Varies	08:00:00.0000000	10:00:00.0000000	101	Computer Science	Tech University
8	Introduction ...	Monday	09:00:00.0000000	11:00:00.0000000	101	Computer Science	Tech University

faculty_id	start_time	end_time	first_department	second_department

faculty_id	assignment_count	department_name	
1	105	1	Business Administration

faculty_id	assignment_count	department_name	
1	105	1	Business Administration

student_id	first_name	last_name	course_id	course_name	times_enrolled

Query executed successfully.

DESKTOP-GDUKF66\SQLEXPRESS ... DESKTOP-GDUKF66\Abdul... sss 00:00:00 32 rows



## 2 Question#2

### 2.1 Examples for Anomalies

#### 2.1.1 Insertion Anomaly

An insertion anomaly occurs when some data cannot be inserted into a table without the presence of other data.

1. If we want to insert a student into the table it would not be possible without inserting the course offer id and course id and course details will be repeated. This leads to inefficiency and redundancy.
2. For each course to be added there needs to be a student and course offer id otherwise the course cannot be inserted and in this way student data will be repeated many times leading to redundancy and inefficiency.
3. If we want to offer a course then it cannot be done without inserting data of the student and course leading to inefficiency and redundancy.

#### 2.1.2 Deletion Anomaly

A deletion anomaly occurs when one row is deleted and it leads to deletion of other rows also.

1. If a course being offered by course offer id is decided to be no longer offered and removed by let's say the HOD then all the rows with the course offer id leading to loss of data of students and courses.
2. If a course is decided to be removed from the program and decided to be never offered again let's say by the HEC then the student data and historical data will be lost.
3. If suppose in summer semester courses were offered but nobody took any courses then deleting the semester data will lead to loss of other data in table.

### 2.1.3 Update Anomaly

An update anomaly occurs when a modification is made in one row and as a result multiple rows are affected.

1. If the name of course is changed as a result all the rows with the course name will be affected.
2. If a student changes his city as a result all the rows with his data will be affected.
3. If by mistake at insertion time the course offered year was entered wrong and now has to be corrected so as a result multiple rows will be affected.

## 2.2 Normalization

### 1<sup>st</sup> Normal Form:

Requirements:

1. No multivalued attribute must exist if so, they will be broken and moved to a new table and a foreign key will be set in the original first table. (All fields scalar values)
2. A primary key must be defined based on the condition it could be composite primary key keeping minimum number of columns possible.

Composite Primary Key: Student\_Id, Course\_Offer\_Id

Reason: As student id will always be unique with the combination of course offer id because a student cannot take same course offer id more than once.

Now 1NF achieved.

<u>Student_Id</u>	Student_City	Student_Name	<u>Course_Offer_Id</u>	Course_Semester	Course_Offered_Year	Course_Grade	Course_Id	Course_Name
-------------------	--------------	--------------	------------------------	-----------------	---------------------	--------------	-----------	-------------

2NF:

Requirements:

1. Remove partial dependencies by making a new table if the primary key is composite and the primary key of new table will be foreign key in original old table.

Functional Dependencies:

1. {Student\_Id} -> {Student\_City}
2. {Student\_Id} -> {Student\_Name}

1. {Course\_Offer\_Id} -> {Course\_Semester}
  2. {Course\_Offer\_Id} -> {Course\_Offered\_Year}
  3. {Course\_Offer\_Id} -> {Course\_Id}
  4. {Course\_Offer\_Id} -> {Course\_Name}
5. 1. {Student\_Id, Course\_Offer\_Id} -> {Course\_Grade}

We have named the original table as Enrollment which has student id and course offer id as both primary and foreign keys coming from respective tables.

In 2NF two new tables have been created each with their primary key underlined.

Enrollment

<u>Student_Id</u> -----	<u>Course_Offer_Id</u> -----	Course_Grade
----------------------------	---------------------------------	--------------

Student

<u>Student_Id</u>	Student_City	Student_Name
-------------------	--------------	--------------

Course\_Offer

<u>Course_Offer_Id</u>	Course_Semester	Course_Offered_Year	Course_Id	Course_Name
------------------------	-----------------	---------------------	-----------	-------------

Now 2NF achieved.

3NF:

Requirements:

1. Remove transitive dependencies meaning no attribute depends on a non-primary key attribute which depends on a primary key. If so, the attributes which depend on the non-primary key attribute will be moved to a new table where the non-primary key attribute in the old table will be a primary key in the new table and foreign key in the old table.

The course id and course name attributes were moved to a new table and course id was made primary key in the new table namely Course. In the course offer table, the course id attribute is now a foreign key linked to Course table.

#### Enrollment

<u>Student_Id</u> -----	<u>Course_Offer_Id</u> -----	Course_Grade
----------------------------	---------------------------------	--------------

#### Student

<u>Student_Id</u>	Student_City	Student_Name
-------------------	--------------	--------------

#### Course\_Offer

<u>Course_Offer_Id</u>	Course_Semester	Course_Offered_Year	Course_Id -----
------------------------	-----------------	---------------------	--------------------

#### Course

<u>Course_Id</u>	Course_Name
------------------	-------------

Now 3NF achieved.