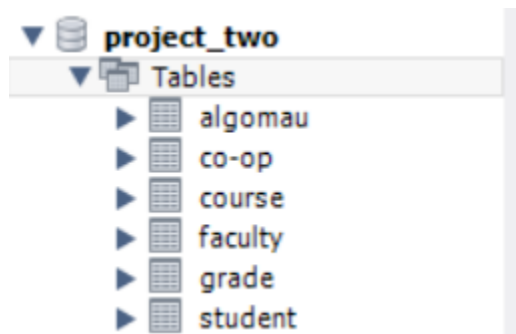


COSC 4606 Project 2

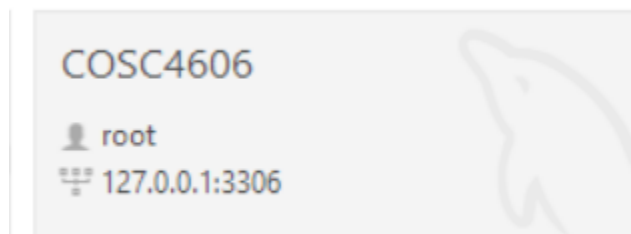
David Asare

Username: Dasare

Tables exported from MS Access to MYSQL workbench



I am using the root user as the database administrator since it has all the administrator privileges.



JDBC Demo

Below is a screenshot of the jdbc driver printing out the first 3 columns from the COURSE database using the IntelliJ programming app.

```
JDBC_Pro2.java x
1 import com.mysql.cj.x.protobuf.MysqlxDatatypes;
2
3 import java.sql.Connection;
4 import java.sql.DriverManager;
5 import java.sql.ResultSet;
6 import java.sql.Statement;
7
8 public class JDBC_Pro2 {
9     public static void main(String[] args) {
10         try{
11             Class.forName("com.mysql.jdbc.Driver");
12             Connection con= DriverManager.getConnection(
13                 url: "jdbc:mysql://localhost:3306/project_two", user: "root", password: "pswd");
14             Statement stmt=con.createStatement();
15             ResultSet rs=stmt.executeQuery( sql: "select * from course");
16             while(rs.next())
17                 System.out.println(rs.getInt( columnIndex: 1)+" "+rs.getString( columnIndex: 2)+" "+rs.getString( columnIndex: 3));
18             con.close();
19         }catch(Exception e){ System.out.println(e);}
20     }
21 }
```

```
Run: JDBC_Pro2 x
C:\Users\david\jars\openjdk-17.0.1\bin\java.exe -javaagent:
Loading class 'com.mysql.jdbc.Driver'. This is deprecated. Th
2006 Data Structures One Tech
2007 Data Structures Two Tech
2056 Discrete Math Two Math_Dep
2406 Assembly Language Business_Tech
2706 Mobile Technology Tech
2956 Internet Tools Business_Tech

Process finished with exit code 0
```

Full COURSE table in MySQL workbench

project2_DavidAsare course x

Limit to 1000 rows

1 • `SELECT * FROM project_two.course;`

Result Grid Filter Rows: Export: Wrap Cell Content:

	course_ID	course_Name	department	section_ID	teacher teaching course	teacher_ID	team_teaching
▶	2006	Data Structures One	Tech	9	Tony Stark	261	0
	2007	Data Structures Two	Tech	4	Felicity Smoak	264	1
	2056	Discrete Math Two	Math_Dep	7	Roy Harper	266	0
	2406	Assembly Language	Business_Tech	3	John Wick	262	0
	2706	Mobile Technology	Tech	5	Slade Wilson	277	0
	2956	Internet Tools	Business_Tech	8	Oliver Queen	263	1

Registrar User

project2_DavidAsare Administration - Users and Privileges x

COSC4606 Users and Privileges

User Accounts

User	From Host
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
registrar	%
root	localhost

Details for account registrar@%

Login Account Limits Administrative Roles Schema Privileges

Schema	Privileges
project_two	INSERT, SELECT, UPDATE

Schema and Host fields may use % and _ wildcards.
The server will match specific entries before wildcarded ones.

Revoke All Privileges Delete Entry Add Entry...

Object Rights

☐ SELECT
☐ INSERT
☐ UPDATE
☐ DELETE
☐ EXECUTE
☐ SHOW VIEW

DDL Rights

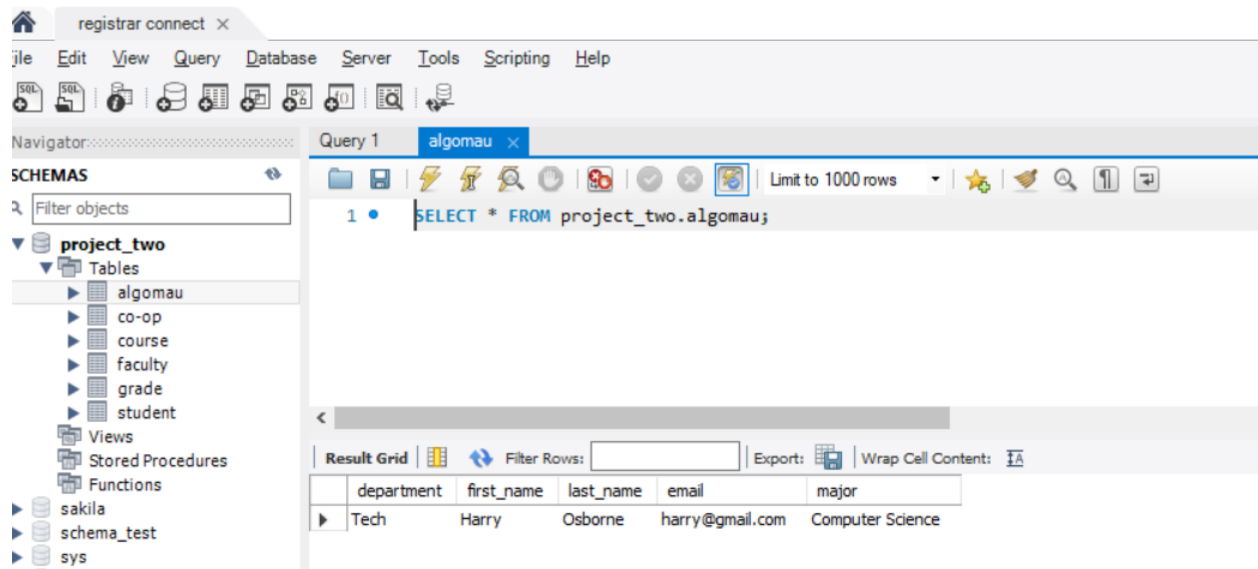
☐ CREATE
☐ ALTER
☐ REFERENCES
☐ INDEX
☐ CREATE VIEW
☐ CREATE ROUTINE
☐ ALTER ROUTINE
☐ EVENT
☐ DROP

Other Rights

☐ GRANT OPTION
☐ CREATE TEMPORARY TABLES
☐ LOCK TABLES

Add Account Delete Refresh Revert Apply

This is the ALGOMAU table below showing a student registered in the university.



After the insert code was successful in the JDBC, the student “Barbra Gordon” was in enrolled into the Algoma university in the ALGOMAU table.

```

public class JDBC_Registrar {
    public static void main(String[] args) {
        /* Insert statement demonstration */
        try{
            Class.forName("com.mysql.jdbc.Driver");
            //1. Get connection
            Connection con= DriverManager.getConnection(
                url: "jdbc:mysql://localhost:3306/project_two", user: "registrar", password: "PSreg");
            //2. Get statement
            Statement stmt=con.createStatement();
            // 3. Execute query
            String sql = "insert into algomau"
                + "(department,first_name,last_name,email,major)"
                + "values( 'Medic', 'Barbra', 'Gordon', 'barbra@gmail.com', 'Biology')";

            stmt.executeUpdate(sql);

            System.out.println("Insert complete");
            con.close();
        }catch(Exception e){ System.out.println(e);}
    }
}

```

```

Run: JDBC_Registrar x
C:\Users\david\.jdk\openjdk-17.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ
Loading class 'com.mysql.jdbc.Driver'. This is deprecated. The new driver class is 'com.mysql.c
Insert complete
Process finished with exit code 0

```

registrar connect x

Edit View Query Database Server Tools Scripting Help

Query 1 algomau x

Limit to 1000 rows

1 • SELECT * FROM project_two.algomau;

Result Grid Filter Rows: Export: Wrap Cell Content:

department	first_name	last_name	email	major
Tech	Harry	Osborne	harry@gmail.com	Computer Science
Medic	Barbra	Gordon	barbra@gmail.com	Biology

project_two

- Tables
 - algomau
 - co-op
 - course
 - faculty
 - grade
 - student
- Views
- Stored Procedures
- Functions
- sakila
- schema_test
- sys
- world

Current grade table.

Result Grid					
Filter Rows: <input type="text"/>					
Export: <input type="text"/> Wrap Cell Content: <input type="text"/>					
	grading_ID	final_grades	course_name	teachers_name	student_ID
▶	501	100	Data Structures One	Tony Stark	161
	502	95	Data Structures Two	Felicity Smoak	162
	503	50	Discrete Math Two	Roy Harper	166
	504	86	Assembly Language	John Wick	163
	505	78	Internet Tools	Oliver Queen	164
	506	90	Internet Tools	Oliver Queen	165
	507	95	Mobile Technology	Slade Wilson	167

Updating final grade 95 to 96 where “grading_ID” is 502 successfully.

```
public class JDBC_Registrar {
    public static void main(String[] args) {
        /* Insert statement demonstration */
        try{
            Class.forName("com.mysql.jdbc.Driver");
            //1. Get connection
            Connection con= DriverManager.getConnection(
                url: "jdbc:mysql://localhost:3306/project_two", user: "registrar", password: "PSreg");
            //2. Get statement
            Statement stmt=con.createStatement();
            // 3. Execute query
            /* String sql = "insert into algomau"
                + "(department,first_name,last_name,email,major)"
                + "values( 'Medic','Barbra','Gordon','barbra@gmail.com','Biology')";

            stmt.executeUpdate(sql);

            System.out.println("Insert complete"); */

            String sql = "update grade"
                + " set final_grades= '96'"
                + "where grading_ID= '502'";
            stmt.executeUpdate(sql);
            System.out.println("Update complete");
            con.close();
        }catch(Exception e){ System.out.println(e);}
    }
}
```

registrat connect x

File Edit View Query Database Server Tools Scripting Help

Navigator: Query 1 grade x

Limit to 1000 rows

1 • SELECT * FROM project_two.grade;

SCHEMAS

Filter objects

project_two

- Tables
 - algomau
 - co-op
 - course
 - faculty
 - grade
 - student
- Views
- Stored Procedures
- Functions

sakila

schema_test

sys

world

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	grading_ID	final_grades	course_name	teachers_name	student_ID
▶	501	100	Data Structures One	Tony Stark	161
	502	96	Data Structures Two	Felicity Smoak	162
	503	50	Discrete Math Two	Roy Harper	166
	504	86	Assembly Language	John Wick	163
	505	78	Internet Tools	Oliver Queen	164
	506	90	Internet Tools	Oliver Queen	165
	507	95	Mobile Technology	Slade Wilson	167

Students User

First I created a view for the student Clark Kent for his academic summary, joining the student and grade tables with dba.

```
/* Creating view for student Clark Kent academic summary */
create VIEW `Clark Kent Academic Summary` AS
select project_two.student.student_ID, first_name, last_name, courses_ID, course_name, final_grades
from student, grade
where student.student_ID = grade.student_ID
and grading_id='501';
```

project2_DavidAsare* clark kent academic summary x

Limit to 1000 rows

1 • SELECT * FROM project_two.`clark kent academic summary`;

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	student_ID	first_name	last_name	courses_ID	course_name	final_grades
▶	161	Clark	Kent	2006	Data Structures One	100

This is the student user with select privilege.

The screenshot shows the MySQL 'Users and Privileges' window. On the left, a table lists user accounts:

User	From Host
mysql.infoschema	localhost
mysql.session	localhost
mysql.sys	localhost
registrar	%
root	localhost
student	%

The 'Details for account student@%' window is open, showing the 'Schema Privileges' tab. It displays a table with the following data:

Schema	Privileges
project_two	SELECT

Below this table, there are sections for 'Object Rights', 'DDL Rights', and 'Other Rights', each with a list of checkboxes for various privileges. The 'student' user has the 'SELECT' privilege on the 'project_two' schema.

Viewing academic summary for Clark Kent with JDBC successfully.

```
public class JDBC_student {
    public static void main(String[] args) {

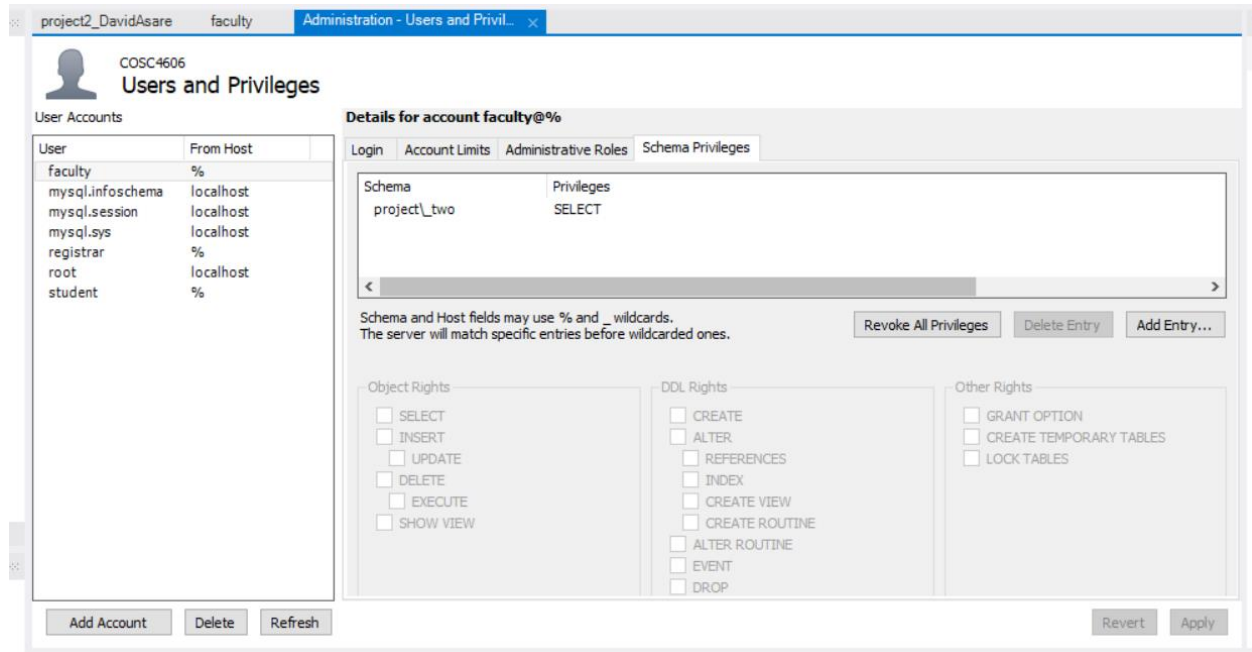
        try {
            Class.forName("com.mysql.jdbc.Driver");
            //1. Get connection
            Connection con = DriverManager.getConnection(
                url: "jdbc:mysql://localhost:3306/project_two", user: "student", password: "PSstudent");
            //2. Get statement
            Statement stmt = con.createStatement();
            // 3. Execute query
            ResultSet rs=stmt.executeQuery( sql: "select * from clark_kent_academinc_summary");
            while(rs.next())
                System.out.println(rs.getInt( columnIndex: 1)+" "+rs.getString( columnIndex: 2)+" "+rs.getString( columnIndex: 3)+
                    " "+rs.getInt( columnIndex: 4)+ " "+rs.getString( columnIndex: 5)+ " "+rs.getInt( columnIndex: 6));
            con.close();
        }catch(Exception e){ System.out.println(e);}
    }
}
```



```
Run: JDBC_student x
C:\Users\david\.jdk\openjdk-17.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBr
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is
161 Clark Kent 2006 Data Structures One 100
Process finished with exit code 0
```

Faculty User

```
/*Creating view for faculty member Tony Stark */
• create VIEW `Tony_Stark_Courses-Taught` AS
  select teacher_ID, first_name, last_name, course_offered, second_course_offered, Term
  from faculty
  where teacher_ID= '261';
```



This is going with the scenario of Tony Stark viewing his class list below successfully.

```

public static void main(String[] args) {

    try {
        Class.forName("com.mysql.jdbc.Driver");
        //1. Get connection
        Connection con = DriverManager.getConnection(
            url: "jdbc:mysql://localhost:3306/project_two", user: "faculty", password: "PSfaculty");
        //2. Get statement
        Statement stmt = con.createStatement();
        // 3. Execute query
        ResultSet rs=stmt.executeQuery( sql: "select * from tony_stark_courses_taught");
        while(rs.next())
            System.out.println(rs.getInt( columnIndex: 1)+" "+rs.getString( columnIndex: 2)+" "+rs.getString( columnIndex: 3)+
                " "+rs.getInt( columnIndex: 4)+ " "+rs.getInt( columnIndex: 5)+ " "+rs.getString( columnIndex: 6));
        con.close();
    }catch(Exception e){ System.out.println(e);}
}

```

Run: JDBC_Faculty x

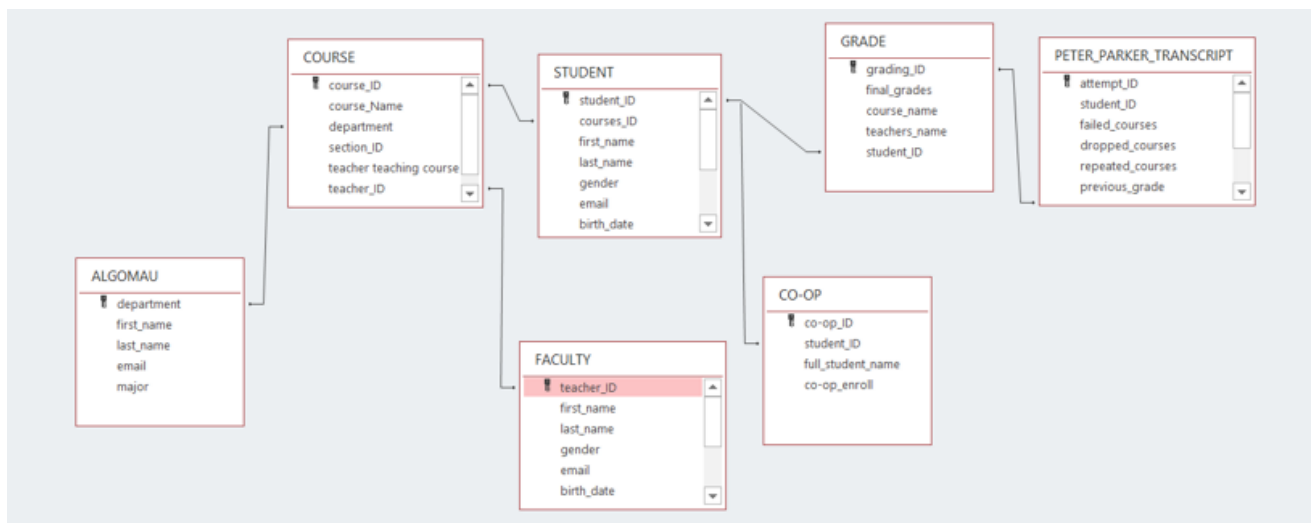
```

C:\Users\david\.jdk\openjdk-17.0.1\bin\java.exe "-javaagent:C:\Users\david\
Loading class 'com.mysql.jdbc.Driver'. This is deprecated. The
261 Tony Stark 2006 2007 Winter
Process finished with exit code 0

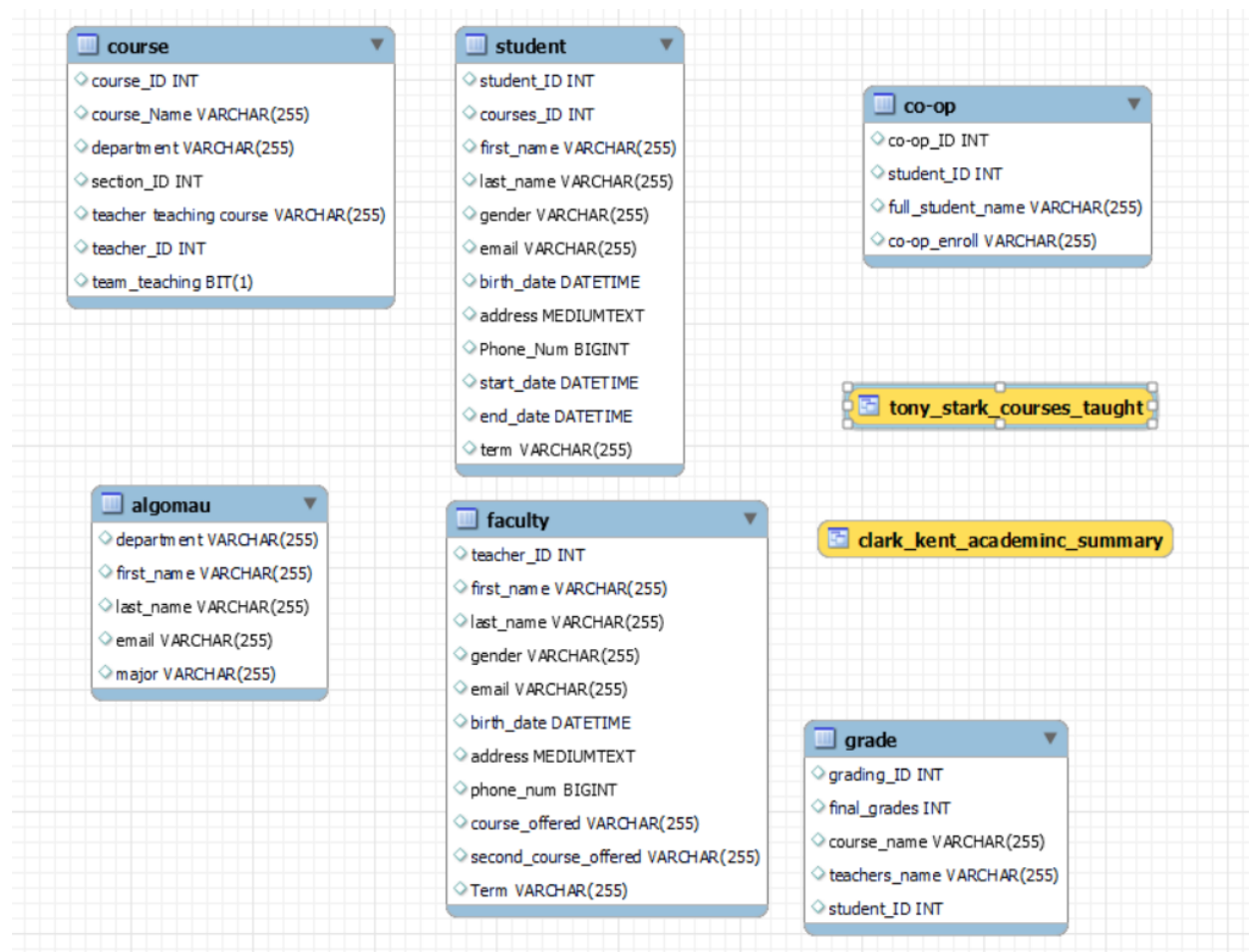
```

ER-Diagram

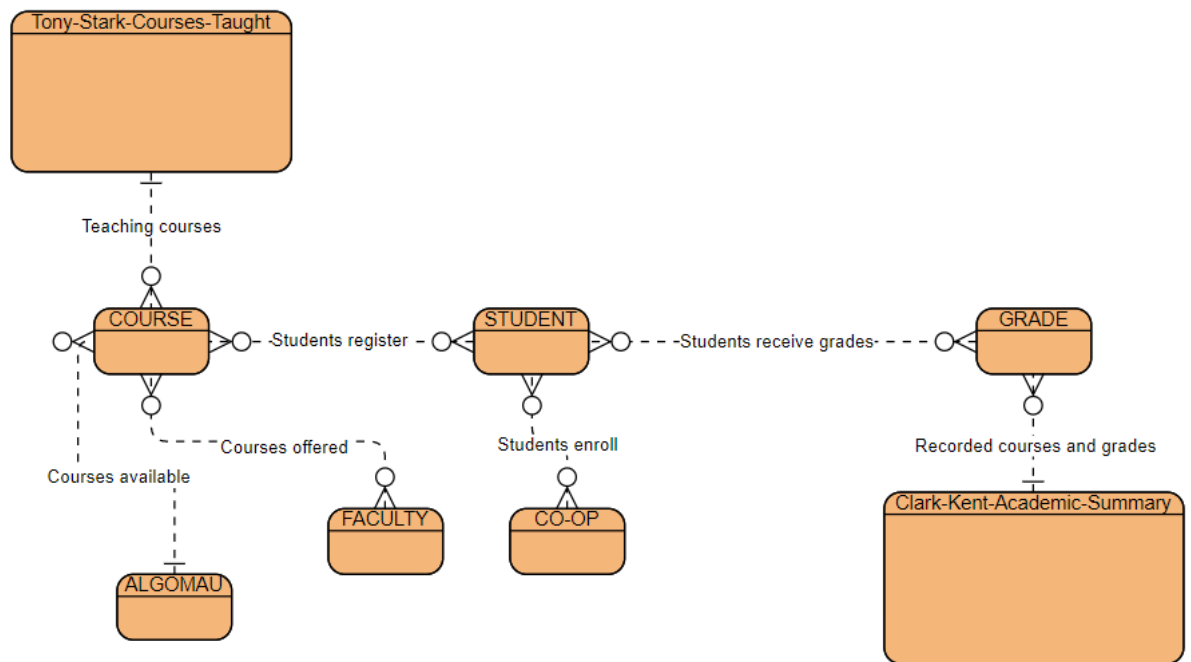
Database Schema below.



Tables and views of diagram generated in the workbench.



Relationship of the ER-Diagram below.



Normal Forms

Co-op table.

co-op_ID	student_ID	full_student_name	co-op_enroll
1001	161	Clark Kent	yes
1002	162	Lois Lane	yes
1003	163	Peter Parker	no
1004	164	Bruce Wayne	no
1005	165	Steve Rogers	yes
1007	167	Cassandra Cain	yes

The table is in 1NF because the values are not divided. The columns contain values of the same type and each column has a unique name. This is 2NF as well because the columns depend on the "co-op_ID" attribute.

Algoma Table

	department	first_name	last_name	email	major
▶	Tech	Harry	Osborne	harry@gmail.com	Computer Science
	Medic	Barbra	Gordon	barbra@gmail.com	Biology

This is 1NF because values are not divided, columns have unique names and values of the same type. It is also 2NF because all the columns can depend on “department” column.

Course Table

	course_ID	course_Name	department	section_ID	teacher teaching course	teacher_ID	team_teaching
▶	2006	Data Structures One	Tech	9	Tony Stark	261	0
	2007	Data Structures Two	Tech	4	Felicity Smoak	264	1
	2056	Discrete Math Two	Math_Dep	7	Roy Harper	266	0
	2406	Assembly Language	Business_Tech	3	John Wick	262	0
	2706	Mobile Technology	Tech	5	Slade Wilson	277	0
	2956	Internet Tools	Business_Tech	8	Oliver Queen	263	1

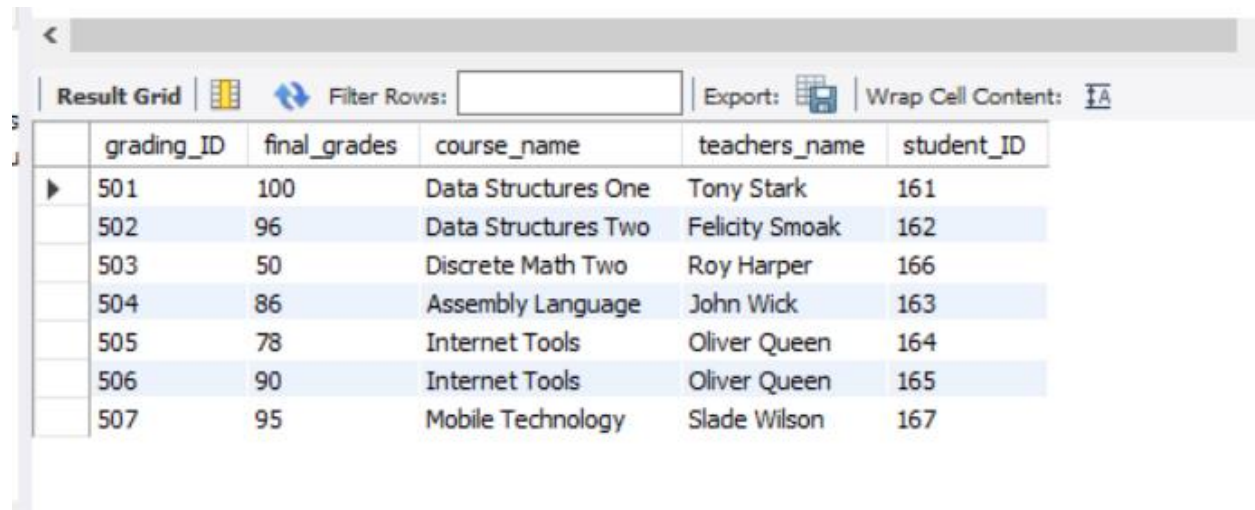
This is 1NF because values are not divided, columns have unique names and values of the same type.

Faculty Table

	teacher_ID	first_name	last_name	gender	email	birth_date	address	phone_num	course_offered	second_course_offered	Term
▶	261	Tony	Stark	male	tony@gmail.com	1976-10-06 00:00:00	26 Jmp St	6135550242	2006	2007	Winter
	262	John	Wick	male	john@gmail.com	1970-04-09 00:00:00	27 Jmp St	6135550034	2406	HOLL	Fall
	263	Oliver	Queen	male	oliver@gmail.com	1980-03-02 00:00:00	28 Jmp St	6135550426	2956	HOLL	Fall
	264	Felicity	Smoak	female	felicity@gmail.com	1988-12-06 00:00:00	29 Jmp St	6135550147	2006	2007	Winter
	265	Thea	Queen	female	thea@gmail.com	1990-07-20 00:00:00	30 Jmp St	6135550788	2956	HOLL	Fall
	266	Roy	Harper	male	roy@gmail.com	1989-10-06 00:00:00	31 Jmp St	6135550255	2056	HOLL	Fall
	277	Slade	Wilson	male	slade@gmail.com	1974-10-10 00:00:00	32 Jmp St	6135550244	2706	HOLL	Fall

This is 1NF because values are not divided, columns have unique names and values of the same type. It is also 2NF because all the columns can depend on the “teacher_ID” attribute.

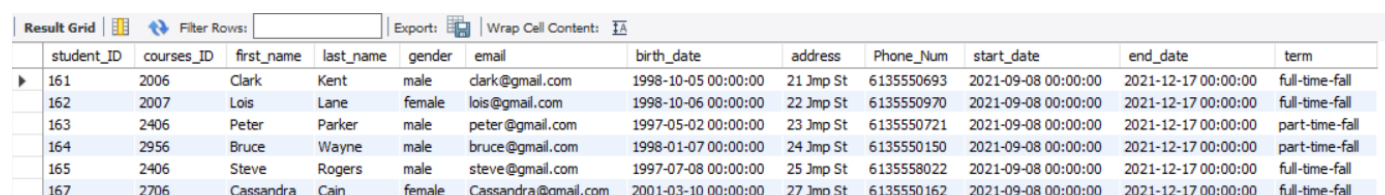
Grade



	grading_ID	final_grades	course_name	teachers_name	student_ID
▶	501	100	Data Structures One	Tony Stark	161
	502	96	Data Structures Two	Felicity Smoak	162
	503	50	Discrete Math Two	Roy Harper	166
	504	86	Assembly Language	John Wick	163
	505	78	Internet Tools	Oliver Queen	164
	506	90	Internet Tools	Oliver Queen	165
	507	95	Mobile Technology	Slade Wilson	167

This is 1NF because values are not divided, columns have unique names and values of the same type. It is also 2NF because all the columns can depend on the “grading_ID” attribute.

Student



	student_ID	courses_ID	first_name	last_name	gender	email	birth_date	address	Phone_Num	start_date	end_date	term
▶	161	2006	Clark	Kent	male	clark@gmail.com	1998-10-05 00:00:00	21 Jmp St	6135550693	2021-09-08 00:00:00	2021-12-17 00:00:00	full-time-fall
	162	2007	Lois	Lane	female	lois@gmail.com	1998-10-06 00:00:00	22 Jmp St	6135550970	2021-09-08 00:00:00	2021-12-17 00:00:00	full-time-fall
	163	2406	Peter	Parker	male	peter@gmail.com	1997-05-02 00:00:00	23 Jmp St	6135550721	2021-09-08 00:00:00	2021-12-17 00:00:00	part-time-fall
	164	2956	Bruce	Wayne	male	bruce@gmail.com	1998-01-07 00:00:00	24 Jmp St	6135550150	2021-09-08 00:00:00	2021-12-17 00:00:00	part-time-fall
	165	2406	Steve	Rogers	male	steve@gmail.com	1997-07-08 00:00:00	25 Jmp St	6135558022	2021-09-08 00:00:00	2021-12-17 00:00:00	full-time-fall
	167	2706	Cassandra	Cain	female	Cassandra@gmail.com	2001-03-10 00:00:00	27 Jmp St	6135550162	2021-09-08 00:00:00	2021-12-17 00:00:00	full-time-fall

This is 1NF because values are not divided, columns have unique names and values of the same type. It is also 2NF because all the columns can depend on the “student_ID” column.

Forms

Below are the forms viewed in MYSQL workbench which have the same functionality as MS Access in project 1.

Algomau Form

Form Editor | Navigate: ⏮ ⏪ 1 / 2 ⏩ ⏭

Department:	Tech
First_name:	Harry
Last_name:	Osborne
Email:	harry@gmail.com
Major:	Computer Science

Result Grid
Form Editor
Field Types
Query Stats
Execution Plan

CO-OP Form

Form Editor | Navigate: ⏮ ⏪ ⏩ ⏭

Co-op_ID:	1001
Student_ID:	161
Full_student_name:	Clark Kent
Co-op_enroll:	yes

Course Form

Form Editor | Navigate: ⏮ ⏪ ⏩ ⏭

Course_ID:	2006
Course_Name:	Data Structures One
Department:	Tech
Section_ID:	9
Teacher teaching course:	Tony Stark
Teacher_ID:	261
Team_teaching:	0

Faculty Form

Form Editor

Navigate:
 ⏮
⏪
1 / 7
⏩
⏭

Teacher_ID:	261
First_name:	Tony
Last_name:	Stark
Gender:	male
Email:	tony@gmail.com
Birth_date:	1976-10-06 00:00:00
Address:	26 Jmp St
Phone_num:	6135550242
Course_offered:	2006
Second_course_offered:	2007
Term:	Winter

Result Grid

Form Editor

Field Types

Query Stats

Execution Plan

Grade Form

Form Editor

Navigate:
 ⏮
⏪
1 / 7
⏩
⏭

Grading_ID:	501
Final_grades:	100
Course_name:	Data Structures One
Teachers_name:	Tony Stark
Student_ID:	161

Student Form

Form Editor | Navigate: 1 / 6

Student_ID: 161

Courses_ID: 2006

First_name: Clark

Last_name: Kent

Gender: male

Email: clark@gmail.com

Birth_date: 1998-10-05 00:00:00

Address: 21 Jmp St

Phone_Num: 6135550693

Start_date: 2021-09-08 00:00:00

End_date: 2021-12-17 00:00:00

Term: full-time-fall

student 1 x

Read Only

Reports from Project 1

CO-OP_REPORT				
full_student_name	course_ID	section_ID	term	
Bruce Wayne	2956	8	part-time-fall	
Cassandra Cain	2706	5	full-time-fall	
Clark Kent	2006	9	full-time-fall	
Lois Lane	2007	4	full-time-fall	
Peter Parker	2406	3	part-time-fall	
Steve Rogers	2406	3	full-time-fall	
October 11, 2021			Page 1 of 1	

DEGREE_ENROLLMENT_REPORT

full_student_name	student_ID	Phone_Num	department
Bruce Wayne	164	6135550150	Business_Tech
Cassandra Cain	167	6135550162	Tech
Clark Kent	161	6135550693	Tech
Lois Lane	162	6135550970	Tech
Peter Parker	163	6135550721	Business_Tech
Steve Rogers	165	6135558022	Business_Tech

October 12, 2021

Page 1 of 1

INSTRUCTOR_REPORT

teacher_ID	first_name	last_name	course_ID	phone_num	student_ID	Term
261	Tony	Stark	2006	6135550242	161	Winter
264	Felicity	Smoak	2007	6135550147	162	Winter
262	John	Wick	2406	6135550034	163	Fall
263	Oliver	Queen	2956	6135550426	164	Fall
262	John	Wick	2406	6135550034	165	Fall
277	Slade	Wilson	2706	6135550244	167	Fall

October 12, 2021

Page 1 of 1

INSTRUCTOR_COURSES_REPORT

Term	course_ID	course_Name	teacher_ID
Fall		Discrete Math Two	266
	2406	Assembly Language	262
	2706	Mobile Technology	277
	2956	Internet Tools	263
Winter	2006	Data Structures One	261
	2007	Data Structures Two	264

October 12, 2021

Page 1 of 1

References

<https://www.javatpoint.com/example-to-connect-to-the-mysql-database>

<https://dev.mysql.com/doc/connector-odbc/en/connector-odbc-examples-tools-with-access-export.html>