(23047617) Niran Bhatta database.docx

Islington College,Nepal

Document Details

Submission ID
trn:oid:::3618:79924192

Submission Date
Jan 23, 2025, 11:23 AM GMT+5:45

Download Date
Jan 23, 2025, 11:26 AM GMT+5:45

File Name
(23047617) Niran Bhatta database.docx

File Size
29.4 KB

71 Pages

4,021 Words

25,373 Characters

34% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

- **117** Not Cited or Quoted 33%
  Matches with neither in-text citation nor quotation marks
- **5** Missing Quotations 1%
  Matches that are still very similar to source material
- **1** Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

- 4% Internet sources
- 1% Publications
- 34% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- 117 Not Cited or Quoted 33%
  Matches with neither in-text citation nor quotation marks
- 5 Missing Quotations 1%
  Matches that are still very similar to source material
- 1 Missing Citation 0%
  Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

- 4% Internet sources
- 1% Publications
- 34% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Submitted works	islingtoncollege on 2025-01-03 — 7%
2	Submitted works	islingtoncollege on 2025-01-03 — 4%
3	Submitted works	islingtoncollege on 2025-01-02 — 3%
4	Submitted works	islingtoncollege on 2025-01-03 — 3%
5	Submitted works	islingtoncollege on 2025-01-02 — 3%
6	Submitted works	islingtoncollege on 2024-12-30 — 2%
7	Submitted works	islingtoncollege on 2025-01-03 — 1%
8	Submitted works	Softwarica College Of IT & E-Commerce on 2022-02-23 — 1%
9	Submitted works	Australian Catholic University on 2024-08-30 — <1%
10	Submitted works	University of Wales Institute, Cardiff on 2023-09-18 — <1%

11 Submitted works
University of Wollongong on 2023-03-26 <1%

12 Submitted works
Kingston University on 2024-01-09 <1%

13 Submitted works
Federation University on 2023-01-10 <1%

14 Submitted works
Middle East College of Information Technology on 2024-05-22 <1%

15 Submitted works
University College Birmingham on 2025-01-13 <1%

16 Submitted works
University of Warwick on 2024-06-24 <1%

17 Submitted works
islingtoncollege on 2024-12-30 <1%

18 Submitted works
islingtoncollege on 2024-12-30 <1%

19 Submitted works
Lincoln Institute of Higher Education Ltd on 2024-10-06 <1%

20 Submitted works
Middle East College on 2024-12-18 <1%

21 Submitted works
University of East London on 2024-09-06 <1%

22 Submitted works
York St John University on 2025-01-21 <1%

23 Submitted works
Aberystwyth University on 2017-11-09 <1%

24 Submitted works
Monash University on 2024-10-28 <1%

25 Submitted works
University of Bedfordshire on 2023-01-19 <1%

26 Submitted works
University of Lancaster on 2024-07-19 <1%

27 Submitted works
University of Sheffield on 2010-05-14 <1%

28 Publication
Sanjiv Purba. "High-Performance Web Databases - Design, Development, and Dep... <1%

29 Submitted works
University of Bedfordshire on 2023-09-06 <1%

30 Submitted works
islingtoncollege on 2024-12-22 <1%

31 Submitted works
Icon College of Technology and Management on 2019-06-13 <1%

32 Submitted works
Informatics Education Limited on 2010-10-27 <1%

33 Submitted works
Middle East College of Information Technology on 2016-05-04 <1%

34 Submitted works
Monash University on 2024-10-27 <1%

35 Submitted works
United Colleges Group on 2025-01-21 <1%

36 Submitted works
Informatics Education Limited on 2013-11-18 <1%

37 Submitted works
Softwarica College of IT & E-Commerce on 2015-04-21 <1%

38 Submitted works
The University of Wolverhampton on 2022-08-15 <1%

39 Submitted works

University of Salford on 2024-04-16 <1%

40 Submitted works

University of Suffolk on 2023-03-17 <1%

`

CC5051NI Databases

100% Individual Coursework

Autumn 2024

Credit: 15 Semester Long Module

Student Name: Niran Bhatta

London Met ID: 23047617

Assignment Submission Date: December 31, 2024

Word Count: 3676

I confirm that I understand my coursework needs to be submitted online via My Second Teacher Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non_submission and a mark of zero will be awarded.

Contents

Table of Tables

Table of Figures

Introduction

E Classroom Platform is a tool for online education, using technological aids to promote a regular, organized learning environment. The strong side of it is that this will contribute to a continuous educational journey by making use of modern technologies, whereby the content learned will be readily available to learners and easy to monitor and follow. With this comes more flexibility in learning to facilitate diversities in learning and educational needs requirements crucial in a contemporary academic landscape.

Miss Mary wishes to develop an eLearning platform known as the E Classroom System, which can connect student and teacher in a more organized and interactive manner. It would provide several academic programs, such as Computing and Multimedia, with several required modules that the student needs to be studied step by step, so that it becomes easy to follow their studies step by step. Teachers will be able to assign assessments and give feedback on students' work while also sharing helpful resources like video lectures and quizzes. The platform will be easy to use and accessible on different devices, ensuring that students can learn from anywhere. All the information regarding students, programs, modules, and assessment results will be

managed with the help of a strong database. Finally, Miss Mary wishes this platform would make learning fun and more effective for all parties.

Current business activities and operations

The business activities of the E Classroom Platform operational procedure is explained below in a short paragraph:

The student may get enrolled into different programs, and each program may have multiple modules that may fit the fields of study. Then teachers are allocated to different modules. Module allocations are assigned to educators within the module to provide content and support to students, so each module is well supported. Then the program module is managed. Every program consists of multiple modules, hence requiring proper organization, management, and allocation of education resources to present a structured curriculum. The resources are allocated in the platform. The platform is designed to allow access to a variety of learning resources such as video lectures, assignments, and quizzes that must be completed sequentially in order to facilitate progressive learning. After that, the students' work is marked, and results for each assessment are generated which students can then view. This activity is important in terms of reflecting student performance accurately.

Business Rule

Each student must enroll in just one program, which can have one or more students.

Students are not allowed to join multiple programs until they finish the first one.

A teacher can instruct several modules, and a module can be taught by different

teachers

Each module should include several assignments, and each assignment must relate to its specific module.

Programs are made up of various modules that students need to complete to graduate.

Teachers can access multiple resources related to the module

An assignment will only be graded if it is submitted by the deadline.

Teachers can only make announcements for the modules they are assigned to, and each announcement must relate to a specific module.

Results should be published according to the student's program and module.

Assumptions

One module can have many teachers assigned to it

Once submitted, students are not allowed to change or remove any assessments.

A single teacher can make multiple announcements.

38 Entities and attributes identification

Entities and attributes

Entity:

28 An entity is anything that exists in the real world with independent existence. Each entity is identifiable and can be differentiated from other objects. It has a unique identifier. For example, some entities in a gadget store database are customer, order, and product.

Attribute:

The characteristics of an entity are called attributes. Attributes consist of three types of keys named primary, foreign and non_key. For instance, in students, some of the attributes are Student_ID, Student_Name, and Student_PhoneNumber.

Initial ERD

The entities, attributes and primary key are displayed in table below

STUDENT

5 S.N.

Attribute

5

Data type

Size

Constraint

1

Student_ID

Number

18

Primary Key

2

Student_Name

Character

35

Not Null

3

Enrolled_Date

Date

10

Not null

4

Email

Character

35

Unique

5

DOB

Date

10

Not null

5 PROGRAM

S.N.

Attribute

Datatype

Size

Constraint

1

Program_Name

Character

22

5 Primary Key

2

Program_ID

Number

18

Not null

3

Program_Description

Character

200

Not null

4

Program_Duration

Number

6

MODULES

S.N.

Attribute

Datatype

Size

Constraint

1

Module_ID

Number

12

Not null

2

Module_Name

Character

22

Not null

3

Credit_Hours

Number

5

Not null

4

Teacher_Name

Character

30

Not null

5

Teacher_ID

Number

12

Not null

6

Contact

Number

15

Unique

7

Specialization

4 Character

20

Not null

8

Resource_ID

Number

12

Not null

9

Resource_Title

Character

30

Not null

10

Resource_Duration

Number

5

1 Not null

11

Resource_Type

Character

25

Not null

12

Resource_Status

Character

30

Not Null

13

Sequence

Character

10

Unique

14

Assessment_ID

Number

22

Not null

15

Asssment_Title

Character

25

Not null

16

Assessment_Description

Character

1200

Not null

17

Assessment_Status

Character

30

Not Null

18

Weightage

Number

5

Not null

19

Deadline

Date

10

Not null

20

Result_Marks

Number

5

Not null

21

Grade

Character

5

Not null

22

Result_Date

Date

10

Not null

23

Announcement_ID

Number

25

Not null

24

Date_Posted

Date

10

Not null

25

Content

Character

1200

Not null

13 Entity Relationship Diagram (ERD)

The entity relationship diagram of the business rule is shown below:

Figure 1 ERD (Entity Relationship) Diagram

Normalization

In databases, normalization is a method for arranging information in a smart way. The main purposes of normalization are to cut down on duplicate data and to make sure the data stays consistent. When we use normalization, it helps us manage tasks like adding, removing, or changing records easily, like how cleaning up your room helps you find your favorite book faster. Normalization means splitting your data into different tables and creating connections between them based on certain rules. You can think of each step in normalization to tidy up your room until everything is in its right place and easy to reach.

Unnormalized Form (UNF)

Description:

Unnormalized Form is the first state of data organization, often derived from a single source of requirements. In this form, data may include repeated groups and does not adhere to any structured format.

Characteristics:

Typically, the structure straightforward list of attributes.

Contains a collection of data with possible redundancy.

Repeating groups may exist, indicated by curly braces.

Student: { Student_Name, Student_ID, Enrolled_Date, Email, DOB, Program_Name,

Program_ID, Program_Description, Program_Duration, { Module_ID, Module_Name, Credit_Hours, { Teacher_Name, Teacher_ID, Contact, Specialization, {Announcement_ID, Date_Posted, Content }}, Resource_ID, Resource_Title, Resource_Duration, Resource_Type, Sequence} {Assessment_ID, Assessment_Title, Assessment_Info, Assessment_Status, Weightage, Deadline, Result_Marks, Grade, Result_Date}}}

What it looks like: Imagine you have a list with lots of repeated information about students, their courses, and assigned tasks all mixed.

The Problem: A lot of information might be redundant, making it difficult to retrieve or update later.

First Normal Form (1NF)

First Normal Form (1NF) introduces basic structure to the data by ensuring that each cell contains only one value and all entries in a column have same type

Now, let's take the first step towards organizing that closet by ensuring that each item has its place. This is First Normal Form (1NF). If something tends to repeat, like a student taking multiple courses, we need to create separate shelves for those items. For instance, instead of keeping every course listed under each student's name, we create a separate table that neatly lists each student alongside their assigned courses. This way, you can easily add or change courses without messing up other records.

Process to Achieve 1NF:

Separating the repeating groups. After separation, we get:

{ Module_ID, Module_Name, Credit_Hours, { Teacher_Name, Teacher_ID, Contact, Specialization, {Announcement_ID, Date_Posted, Content }}, Resource_ID, Resource_Title, Resource_Duration, Resource_Type, Sequence} {Assessment_ID, Assessment_Title, Assessment_Info, Assessment_Status ,Weightage, Deadline, Result_Marks, Grade, Result_Date}}}

Result After Normalization (1NF)

STUDENT (1NF): Student_Name, Student_ID, Enrolled_Date, Email, DOB, Program_Name, Program_ID, Program_Description, Program_Duration}

Student_Module (1NF): Student_ID, Module_ID, Module_Name, Credit_Hours}

Module_Teacher (1NF): Student_ID, Module_ID, Teacher_Name, Teacher_ID, Contact, Specialization}

Module_Teacher_Announcement (1NF): Student_ID, Module_ID, Teacher_ID,

Announcement_ID, Date_Posted, Content}

6 Student_Module_Resource (1NF): Student_ID, Module_ID, Resource_ID, Resource_Title, Resource_Duration, Resource_Type, Resource_Status, Sequence}

Student_Module_Assessment (1NF): Student_ID, Module_ID, Assessment_ID, Assessment_Title, Assessment_Info, Assessment_Status, Weightage, Deadline, Result_Marks, Grade, Result_Date}

S.N.

Attribute

Datatype

2 Size

Constraint

1

Module_ID

Number

12

Not null

2

Module_Name

Character

22

Not null

3

Credit_Hours

Number

5

3 Not null

4

Teacher_Name

Character

30

Not null

5

Teacher_ID

Number

12

Not null

6

Contact

Number

15

Unique

7

Specialization

4 Character

20

Not null

8

Resource_ID

Number

12

Not null

9

Resource_Title

Character

30

Not null

10

Resource_Duration

4 Number

5

Not null

11

Resource_Type

Character

25

Not null

12

Sequence

Character

10

Unique

13

Assessment_ID

Number

22

3 Not null

14

Assssment_Title

Character

25

Not null

15

Assessment_Info

Character

1200

Not null

16

Assessment_Status

Character

30

Not Null

17

Weightage

Number

5

Not null

18

Deadline

Date

10

Not null

19

Result_Marks

Number

5

Not null

20

Grade

Character

5

Not null

21

34

Result_Date

Date

10

Not null

22

Announcement_ID

Number

25

Not null

23

Date_Posted

Date

10

Not null

24

Content

Character

1200

Not null

Rules to Follow:

Each column must have unique values.

Repeating groups should be kept in separate tables, thus removing redundancy.

Second Normal Form (2NF)

Second Normal Form (2NF) takes 1NF a step further by eliminating partial dependencies. In this form, all non_key attributes must depend on the entire primary key.

Process to Achieve 2NF:

Student:

Since students have only one primary key, the 2NF of student is given as:

Student(2NF) : (Student_Name, Student_ID, Enrolled_Date, Email, DOB, Program_Name, Program_ID, Program_Description, Program_Duration)

Student_Module:

Primary key: Student_ID, Module_ID

Module_Name is partially dependent on Module_ID

Credit_Hours is partially dependent on Module_ID

2NF Scheme:

Module(2NF):

(Module_ID, Module_Name, Credit_Hours)

Student_Module(2NF):

(Module_ID, Student_ID)

Module_Teacher

Primary key: Module_ID, Teacher_ID

Teacher_Name is partially dependent on Teacher_ID

Contact is partially dependent on Teacher_ID

Specialization is partially dependent on Teacher_ID

2NF Scheme:

Teacher(2NF):

(Teacher_ID, Teacher_Name, Contact, Specialization)

Module_Teacher(2NF):

(Student_ID, Module_ID, Teacher_ID)

Module_Teacher_Announcement

Primary Key: Module_ID, Teacher_ID, Announcement_ID

Date_Posted is partially dependent on Announcement_ID

Content is partially dependent on Announcement_ID

2NF Scheme:

Announcement(2NF):

(Announcement_ID, Date_Posted, Content)

Teacher_Announcement(2NF):

(Student_ID, Module_ID, Teacher_ID, Announcement_ID)

Student_Module_Resource

Primary key: Student_ID, Module_ID, Resource_ID

Resource_Title is partially dependent on Resource_ID

Resource_Duration is partially dependent on Resource_ID

9 Resource_Type is partially dependent on Resource_ID

Sequence is partially dependent on Resource_ID

2NF Scheme:

Resource(2NF):

1 (Resource_ID, Resource_Title, Resource_Duration, Resource_Type, Sequence)

Student_Module_Resource(2NF) :

(Student_ID, Mouddle_ID, Resource_ID, Resource_Status)

Student_Module_Assessment

Primary key: Student_ID, Module_ID, Assessment_ID

Assessment_Title is partially dependent on Assessment_ID

Assessment_Info is partially dependent on Assessment_ID

Assessment_Status is partially dependent on Assessment_ID

Weightage is partially dependent on Assessment_ID

Deadline is partially dependent on Assessment_ID

Result_Marks is partially dependent on Assessment_ID

Grade is partially dependent on Assessment_ID

Result_Date is partially dependent on Assessment_ID

2NF Scheme:

Assessment(2NF):

(Assessment_ID, Assessment_Title, Assessment_Info, Weightage, Deadline)

Module_Asst(2NF):

(Student_ID, Module_ID, Assessment_ID, Assessment_Status , Result_Marks, Grade, Date_Posted)

Final Results after 2NF

Student(2NF): (Student_Name, Student_ID, Enrolled_Date, Email, DOB,

Program_Name, Program_ID, Program_Description, Program_Duration)

Module(2NF):(Module_ID, Module_Name, Credit_Hours)

Student_Module(2NF): (Module_ID, Student_ID)

Teacher(2NF): (Teacher_ID, Teacher_Name, Contact, Specialization)

Module_Teacher(2NF): (Student_ID, Module_ID, Teacher_ID)

Announcement(2NF): (Announcement_ID, Date_Posted, Content)

Teacher_Announcement(2NF): (Student_ID, Module_ID, Teacher_ID,

Announcement_ID)

Resource(2NF): (Resource_ID, Resource_Title, Resource_Duration, Resource_Type,

Sequence)

Student_Module_Resouce(2NF): (Student_ID, Module_ID, Resource_ID)

Assessment(2NF) : (Assessment_ID, Assessment_Title, Assessment_Info, Weightage,

Deadline

Module_Asst(2NF): (Student_ID, Module_ID, Assessment_ID, Assessment_Status ,

Result_Marks, Grade, Date_Posted)

Rules to Follow:

Ensure that no non_key attribute is partially dependent on the primary key.

All attributes should directly relate to the primary key.

Third Normal Form(3NF)

Third Normal Form (3NF) aims to eliminate transitive dependencies, which occur when a non_key attribute depends on another non_key attribute. Let's refine the organization even further! Second Normal Form (2NF) helps eliminate redundancy that might still exist.

Rules to Achieve 3NF:

Ensure the database is in 2NF

Eliminate transitive dependencies by ensuring all non_key attributes are directly related to the primary key.

Process to Achieve 3NF

Student:

Program_Name directly depends on Program_ID

Program_Description directly depends on Program_ID

Program_Duration directly depends on Program_ID

Thus, the required 3NF will be:

Student(3NF):

(Student_Name, Student_ID, Enrolled_Date, Email, DOB, Program_Name,

Program_ID)

Program(3NF):

(Program_ID, Program_Name, Program_Description, Program_Duration)

Module

All the non key attributes directly depends on Module_ID resulting in absence of transitive dependency.

Thus, the required 3NF will be:

Module(3NF):

(Module_ID, Module_Name, Credit_Hours)

Student_Module

Non key attributes in Student_Module is absent. Thus, the required 3NF will be

Student_Module(3NF):

(Student_ID,Module_ID)

Teacher

All the non key attributes directly depend on Teacher_ID resulting in absence of transitive dependency

Thus, the required 3NF will be

Teacher(3NF):

(Teacher_ID, Teacher_Name, Contact, Specialization)

Module_Teacher

Non key attributes in Module_Teacher is absent. Thus, the required 3NF will be

Module_Teacher(3NF):

(Student_ID, Module_ID, Teacher_ID)

Announcement

All the non key attributes directly depend on Announcement_ID resulting in absence of transitive dependency

Thus, the required 3NF will be

Announcement(3NF):

(Announcement_ID, Date_Posted, Content)

Teacher_Announcement

Non key attributes in Teacher_Announcement is absent. Thus, the required 3NF will be

Teacher_Announcement(3NF):

(Student_ID, Module_ID, Teacher_ID, Announcement_ID)

Resource

All the non key attributes directly depend on Resource_ID resulting in absence of transitive dependency

Thus, the required 3NF will be:

Resource(3NF):

(Resource_ID, Resource_Title, Resource_Duration, Resource_Type, Sequence)

Stud_Module_Resource

All the non key attributes directly depend on Resource_Status resulting in absence of transitive dependency

Thus, the required 3NF will be:

Stud_Module_Resource(3NF):

(Student_ID, Module_ID, Resource_ID)

Assessment

All the non key attributes directly depend on Assessment_ID resulting in absence of transitive dependency

Thus, the required 3NF will be

Assessment(3NF):

(Assessment_ID, Assessment_Title, Assessment_Info, Weightage, Deadline)

Module_Assessment

Result_Marks, Grade, Date_Posted are directly dependent to Student_ID, Module_ID and Assessment_ID resulting in absence of transitive dependency

Thus, the required 3NF will be

Module_Asst(3NF):

(Student_ID, Module_ID, Assessment_ID,Asessment_Status, Result_Marks, Grade, Date_Posted)

Final Results after 3NF:

Student(3NF) :(Student_Name, Student_ID, Enrolled_Date, Email, DOB, Program_Name, Program_ID)

Program(3NF) :(Program_ID, Program_Name, Program_Description, Program_Duration)

Module(3NF) :(Module_ID, Module_Name, Credit_Hours)

Student_Module(3NF) :(Student_ID, Module_ID)

Teacher(3NF) :(Teacher_ID, Teacher_Name, Contact, Specialization)

Module_Teacher(3NF) :(Student_ID, Module_ID, Teacher_ID)

Announcement(3NF) :(Announcement_ID, Date_Posted, Content)

Teacher_Announcement(3NF) :(Student_ID, Module_ID, Teacher_ID, Announcement_ID)

Resource(3NF) :(Resource_ID, Resource_Title, Resource_Duration, Resource_Type, Sequence)

Stud_Module_Resource(3NF) :(Student_ID, Module_ID, Resource_ID,)

Assessment(3NF) :(Assessment_ID, Assessment_Title, Assessment_Info, Weightage, Deadline)

Module_Asst(3NF) :(Student_ID, Module_ID, Assessment_ID, Asessment_Status, Result_Marks, Grade, Date_Posted)

12) Conclusion

By following these normalization steps—starting from Unnormalized Form through First, Second, and Third Normal Forms—you ensure that your database remains organized, minimizes redundancy, and maintains data integrity. This systematic approach makes your data easier to manage, reduces errors, and enhances overall efficiency. By organizing data through these normalization steps, we can manage our databases like a tidy closet where everything is accessible and efficient.

13) Data Dictionary

Student Table:

S.N.

Attribute

4 Data type

Size

Constraint

1

Student_ID

Number

18

Primary Key

2

Student_Name

Character

35

Not Null

3

Enrolled_Date

Date

10

Not null

4

Email

Character

35

Unique

5

DOB

Date

10

Not null

6

5 Program_ID

Number

18

Foreign Key

Table 1 Student table after 3NF

1 Program Table:

S.N.

Attribute

Data type

Size

Constraint

1

Program_Name

Character

18

Primary Key

2

Program_ID

Number

22

Not Null

3

Program_Description

Character

200

Not null

4

Program_Duration

Number

6

Unique

Table 2 Program table after 3NF

1 Module Table:

S.N.

Attribute

Data type

Size

Constraint

1

Module_ID

Number

12

Primary Key

2

Module_Name

Character

22

Not Null

3

Credit_Hours

Number

5

Not null

Table 3 Module table after 3NF

4 Teacher Table:

S.N.

Attribute

Data type

Size

Constraint

1

Teacher_Name

Character

30

Not null

2

Teacher_ID

Number

12

Primary Key

3

Contact

Number

15

Unique

4

Specialization

Character

20

Not null

Table 4 Teacher table after 3NF

Assessment Table:

S.N.

Attribute

Data type

Size

Constraint

1

Assessment_ID

Number

22

Primary_Key

2

Assssment_Title

Character

25

Not null

3

Assessment_Info

Character

1200

Not null

4

Weightage

Number

5

Not null

5

Deadline

Date

10

Not Null

6

Assessment_Status

Character

30

Not Null

Table 5 Assessment table after 3NF

Announcement Table:

S.N.

Attribute

Data type

Size

Constraint

1

Announcement_ID

Number

25

Primary Key

2

Date_Posted

Date

10

Not null

3

Content

Character

1200

Not null

Table 6 Announcement table after 3NF

Resource Table:

S.N.

Attribute

Data type

Size

Constraint

1

Resource_ID

Number

12

PrimaryKey

2

Resource_Title

Character

30

Not null

3

Resource_Duration

Number

5

Not null

4

Resource_Type

Character

25

Not null

5

Sequence

Character

10

Unique

Table 7 Resource table after 3NF

Bridging Entities

5 Student_Module

S.N.

Attribute

Data type

Size

Constraint

Composite Constraint

1

Student_ID

Number

18

1 Foreign Key

Primary Key

2

Module_ID

Number

12

Foreign Key

Primary Key

Table 8 Student_Module table after 3NF

5 Module_Teacher

S.N.

Attribute

Data type

Size

Constraint

Composite Constraint

1

Student_ID

Number

18

1 Foreign Key

Primary Key

2

Module_ID

Number

12

Foreign Key

Primary Key

3

Teacher_ID

Number

12

Foreign Key

Primary Key

Table 9 Module_Teacher table after 3NF

5 Teacher_Announcement

S.N.

Attribute

Data type

Size

Constraint

Composite Constraint

1

Student_ID

Number

18

1 Foreign Key

Primary Key

2

Module_ID

Number

12

Foreign Key

Primary Key

3

Teacher_ID

Number

22

Foreign Key

Primary Key

4

5 Announcement_ID

Number

25

Foreign Key

Primary Key

Table 10 Teacher_Announcement table after 3NF

5 Module_Resource

S.N.

Attribute

Data type

Size

Constraint

Composite Constraint

1

Student_ID

Number

18

1 Foreign Key

Primary Key

2

Module_ID

Number

12

Foreign Key

Primary Key

3

Resource_ID

Number

12

Foreign Key

Primary Key

Table 11 Module_Resource table after 3NF

Module_Ass

S.N.

Attribute

Data type

Size

Constraint

Composite Constraint

1

Module_ID

Number

12

Foreign Key

Primary Key

2

Student_ID

Number

18

Foreign Key

Primary Key

3

Assessment_ID

Number

22

Foreign Key

Primary Key

4

Assessment_Status

Character

30

24 Not Null

5

Result_Marks

Number

5

Not null

6

Grade

Character

5

Not null

7

Result_Date

Date

10

Not null

Table 12 Module_Assessment table after 3NF

5 Final ERD

Figure 1 Final ERD

14) Implementation

3 14.1) Creating new user.

Figure 2 Creating new user

14.2) Creating student table and describing table

Figure 3 Creating student table.

7 14.3) Creating program table and describing table

Figure 4 Creating program table

14.4) Creating module table and describing table

Figure 5 Creating module table

14.5) Creating Student_Module table and describing table

Figure 6 Creating Student_Module table

14.6) Creating teacher table and describing table

Figure 7 Creating teacher table

14.7) Creating Module_Teacher table and describing table

Figure 8 Creating Module_Teacher table

14.8) Creating announcement table and describing table

Figure 9 Creating announcement

14.9) Creating Teacher_Announcement table and describing table

Figure 10 Creating Teacher_Announcement table

14.10) Creating Resources table and describing table

Figure 11 Creating Resources table

37 14.11) Creating Stud_Module_Resource and describing table

Figure 12 Creating Stud_Module_Resource

3 14.12) Creating assessment table and describing table

Figure 13 Creating assessment table

14.13) Creating Module_Asst table and describing table

Figure 14 Creating Module_Asst table

Inserting the values in table we created

14.14) Inserting the values in program table:

Figure 15 Inserting values in program table

14.15) Inserting values in module table

Figure 16 Inserting values in module tabe

14.16) Inserting values in student table:

12 Figure 17 17 Inserting values in student table:

14.17) Inserting values in teacher table:

Figure 18 Inserting values in teacher table

10 14.18) Inserting values in Student_Module table:

Figure 19 Inserting values in student_module

14.19) Inserting values in Module_Teacher

Figure 20 Inserting values in Module_Teacher

14.20) Inserting values in Announcement table.

Figure 21 Inserting values in Announcement table.

14.21) Inserting values in Techer_Announcement

Figure 22 Inserting values in Techer_Announcement

14.22) Inserting values in Resource table

Figure 23 Inserting values in Resource table

14.23) Inserting values in assessment

Figure 24 Inserting values in assessment

14.24) Inserting values in Module_Asst table

Figure 25 Inserting values in Module_Asst table

15) Querying of Database

11 15.1) List the programs that are available in the college and the total number of students enrolled in each.

Query- SELECT

P.PROGRAM_ID AS Program_ID,

11 P.PROGRAM_NAME AS Program_Name,

COUNT(S.STUDENT_ID) AS Total_Students

FROM

PROGRAM P

LEFT JOIN

STUDENT S

ON

P.PROGRAM_ID = S.PROGRAM_ID

GROUP BY

P.PROGRAM_ID, P.PROGRAM_NAME

ORDER BY

```
P.PROGRAM_ID;
```

This query makes sure that all the programs are listed in a database. Even if the students are not enrolled, all the programs that are listed in the database are displayed.

15.2) List all the announcements made for a particular module starting from 1st May 2024 to 28th May 2024.

17) Critical Evaluation

17.1) Evaluation of the Module, Its Purpose, and Connections to Other Subjects:

The modules in the system play a key role in structuring academic programs. Tit is designed to cover a wide range of modules like programming, data analysis, cybersecurity, and digital marketing. Each module has its own attributes such as its name, ID, and credit hours.

Modules are the glue that holds everything together. They connect teachers to their areas of expertise, students to their courses, and resources like lectures, quizzes, and assignments. This makes it easy to manage what each person needs to do within the system.

What makes modules even more important is how they relate to other parts of the curriculum. For example, they're linked to specific programs, which create a logical flow for students to follow step by step. Modules also integrate multiple disciplines, like combining digital marketing with data analysis to evaluate marketing strategies—showing how interconnected knowledge can be.

17.2) Assessment of the Coursework:

This coursework helps to tackle problems such as managing students, teachers, and assessments in a structured way.

The database design follows proper normalization steps i.e the data is stored in a

efficient way with no unnecessary repetition.

The relationship between entities like students, modules, and teachers are defined, which makes it easy to add more entities in future.

It mirrors real-life challenges of managing academic platforms, preparing you for practical applications.

Areas for Improvement:

Add tools for analyzing data, such as tracking student progress or module success rates, to make the system more insightful.

Include features that allow students to progress at their own pace, like marking modules by difficulty level or prerequisites.

In summary, the module system is well-structured and has a lot of potential for practical use. It connects the dots between different elements of education, creating a strong foundation for an interactive and scalable learning platform.

Bibliography

39 Chris, K. (2022, 12 21). Database. Retrieved from FreeCodeCamp: https://www.freecodecamp.org/news/database-normalization-1nf-2nf-3nf-table-examples/

1 GFG. (2024, 6 3). Retrieved from geeksforgeeks: https://www.geeksforgeeks.org/how-to-draw-entity-relationship-diagrams/

13 Schools. (n.d.). SqlTutorial. Retrieved from W3Schools: https://www.w3schools.com/sql/sql_insert.asp

30 team, O. (2023, 5 3). Ovaledge. Retrieved from Ovaledgr: https://www.ovaledge.com/blog/what-is-data-dictionary