**1. Case Study**

An E-Learning platform aims to provide online courses and educational resources to a wide range of learners. Students can enrol in multiple courses, which are taught by one or many experience instructors. Each course has multiple lessons with different type of content like text file, video file etc. There is Q&A section for every course, where assigned instructor can provide feedback related to the student’s queries. Through the E-Learning platform, student can track their progress for each course’s lessons whether it is completed or in progress.

Fig 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| S.N | Student Name | Contact | DOB | Email | Country | Enrol Course | Enrol Date |
| 1 | Dipesh Maharjan | 98347362 | 1998-03-28 | [iamdipesh@gmail.com](mailto:iamdipesh@gmail.com) | Nepal | Java  Python | 2019-02-19  2020-05-18 |
| 2 | Ronal Silwal | 98343784 | 2000-01-09 | [ronaldo@gmail.com](mailto:ronaldo@gmail.com) | UK | SQL for BI | 2022-09-12 |
| 3 | Prithivi Tandulkar | 98343786 | 2002-09-13 | [pre@gmail.com](mailto:pre@gmail.com) | USA | Java  Angular | 2023-01-23  2023-02-12 |
| 4 | Dipeshor GC | 98384754 | 1998-02-14 | [gcdip@gmail.com](mailto:gcdip@gmail.com) | USA | Python for Data Analysis | 2022-01-29 |

UNF – Identifying Repeating Groups

Student (**StudID**, StudentName, Contact, DOB, Email, Country, {Enrol Course, Enrol Date})

1NF – Splitting repeating groups into separate tables.

Student-1(**StudID**, StudentName, Contact, DOB, Email, Country)

EnrolCourse-1(**CourseID**, CourseName, EnrolDate, **StudID\***)

2NF – Checking for partial dependency,

In Student-1

StudID->StudentName, Contact, DOB, Email, Country

In EnrolCourse-1

CourseID->CourseName

CourseID, StudentID-> EnrolDate

Now getting the tables in 2NF

Student-2(**StudID**, StudentName, Contact, DOB, Email, Country)

Course-2(**CourseID**, CourseName)

Student-Course-2(**StudID**\*, **CourseID\***, EnrolDate)

3NF – Checking for Transitive Dependency

All tables

Student-3(**StudID**, StudentName, Contact, DOB, Email, Country)

Course-3(**CourseID**, CourseName)

Student-Course-3(**StudID**\*, **CourseID\***, EnrolDate)

Fig 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Student Name | Course ID | Course Title | Lesson No | Lesson Title | Lesson Status | Last Accessed Date | Course Instructor ID | Course Instructor Name |
| Aadesh Lama | Java\_019 | Java | 1 | Class and Constructor | Completed | 2023-01-20 | Teach\_1 | Dolma Gurung |
| Aadesh Lama | Java\_019 | Java | 2 | Inheritance | In Progress | 2023-01-21 | Teach\_1 | Dolma Gurung |
| Aadesh Lama | PY\_12 | Python | 1 | Data Dictionary | Completed | 2023-03-01 | Teach\_2 | Samundra Karki |
| Aadesh Lama | PY\_12 | Python | 2 | DataFrame | Completed | 2023-03-04 | Teach\_2 | Samundra Karki |

UNF – Identifying Repeating Groups

Student (StudentName, {CourseID, CourseTitle, {LessonNo, LessonTitle, LessonStatus, Last Accessed}, InstructorID, InstructorName})

1NF – Splitting repeating groups into separate table,

Student - 1 (**StudID,** StudentName)

Course -1 (**CourseID**, CourseTitle, InstructorID, InstructorName, **StudID\***,)

Lesson-1(**LessonNo**, LessonTitle, LessonStatus, LastAccessedID, **StudID\***, **CourseID\*)**

**2NF – Identifying Partial Dependency,**

In Student-Course-1 there is a single primary key

In Course-1 there is composite primary key

CourseID, StudID->

In Lesson-1 there is composite Primary Key

***Final Tables in 2NF***

Student - 2 (**StudID,** StudentName)

Course – 2 (**CourseID**, CourseTitle, InstructorID, InstructorName)

Student-Course -2 (**StudID\***, **CourseID\***)

Lesson-Title -2 (**LessonID\***, **CourseID\*,** LessonTitle)

Lesson-Instructor-Student -2 (**LessonID\***, **StudID\***, **CourseID\*,**  LastAccessedDate, LessonStatus)

3NF – Identifying Transitive Dependency

In Course-2, CourseID -> InstructorID -> InstructorName

**Final Tables after 3NF,**

Student - 3 (**StudID,** StudentName)

Course – 3 (**CourseID**, CourseTitle)

Instructor – 3 (**InstructorID**, InstructorName, **CourseID\***)

Student-Course -3 (**StudID\***, **CourseID\***)

Lesson= -3 (**LessonID**, **CourseID\*,** LessonTitle)

Lesson-Student -3 (**LessonID\***, **StudID\***, LastAccessedDate, LessonStatus)