



islington college

(इस्लिंग्टन कॉलेज)

CC6001NI Advanced Database System Development

40% Individual Coursework

Student Name: Aadesh Shrestha

London Met ID: 22015630

College ID: np01cp4s220163

Assignment Due Date: 20th March, 2024

Assignment Submission Date: 20th March, 2024

Word Count: 5034

I confirm that I understand my coursework needs to be submitted online via MySecondTeacher 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 marks of zero will be awarded.

Table of Contents

1	Introduction.....	1
2	Textual Analysis.....	2
2.1	Student-Course	2
2.2	Student-LessonProgress	2
2.3	Student-QA.....	2
2.4	Course-QA	3
2.5	Course-Instructor.....	3
2.6	Course-Lesson.....	3
2.7	Instructor-QA	4
2.8	Lesson-Content.....	4
2.9	Lesson-Progress	4
3	ERD from Case Study.....	5
4	Normalization	6
4.1	Normalization of fig 1	6
4.2	Normalization of fig 2	8
5	Integration and Assumption	11
5.1	Assumptions	11
5.2	Integration	11
6	Final ERD	12
7	Data Dictionary	13
8	Script.....	21
8.1.1	Student Table.....	21
8.1.2	Course Table	21

8.1.3	Enrollment Table.....	22
8.1.4	Instructor Table	22
8.1.5	Course Instructor Table.....	23
8.1.6	QA Table	24
8.1.7	Lesson table	25
8.1.8	Content Table.....	26
8.1.9	Progress Table.....	26
9	Insert Statement	30
10	Select Statement.....	38
11	Forms	47
11.1	Dashboard.....	47
11.2	Complex Form and Queries	48
11.2.1	SQL Queries.....	48
11.2.2	Dashboard Queries.....	51
11.2.3	Complex Forms.....	52
11.3	Simple Forms	53
12	User Manual.....	56
13	Testing.....	73
13.1	Basic Forms	73
13.1.1	Student Page.....	73
13.1.2	Courses Page.....	77
13.1.3	Instructor Pages.....	81
13.1.4	Lesson Page	85
13.1.5	Progress Page.....	89
13.2	Complex Forms	93

13.2.1	Enrollment Page.....	93
13.2.2	Course Instructors Page	95
13.2.3	Top Courses Page.....	97
13.3	Test Cases	99
13.3.1	Case 1.....	99
13.3.2	Case 2.....	101
14	Further Discussion	105
15	Conclusion	106
16	References.....	107

Table of Figures

Figure 1. Student-Course Relation.....	2
Figure 2. Student-LessonProgress Relation	2
Figure 3. Student-QA relation.....	2
Figure 4. Course-QA Relation	3
Figure 5. Course-Instructor Relation	3
Figure 6. Course-Lesson Instructor.....	3
Figure 7. Instructor-QA Relation	4
Figure 8. Lesson-Content Relation	4
Figure 9. Lesson-LessonProgress Relation.....	4
Figure 10. Initial ERD.....	5
Figure 11. Final ERD	12
Figure 12. Connecting User	28
Figure 13. Table Descriptions(1)	28
Figure 14. Table Descriptions(2)	29
Figure 15. Table Descriptions(3)	29
Figure 16. Student table insertion	30
Figure 17. Course table insertion	30
Figure 18. Instructor table insertion.....	31
Figure 19. Course_Instructor table insertion	32
Figure 20. Enrollment Table insertion	33
Figure 21. Lesson Table insertion	34
Figure 22. Content Table insertion.....	35
Figure 23. QA table insertion.....	36
Figure 24. Progress Table insertion.....	37
Figure 25. Student data	38
Figure 26. Course Data	39
Figure 27. Enrollment Data.....	40
Figure 28. Insturctor Data	41

Figure 29. Course _ Instructor Data.....	42
Figure 30. QA data.....	43
Figure 31. Lesson data	44
Figure 32. Content Data	45
Figure 33. Progress Data.....	46
Figure 34. Home Page.....	47
Figure 35. Student Enrollment Page	52
Figure 36. Course Instructor Page	52
Figure 37. Top Course Enrolled Page	53
Figure 38. Student Page	53
Figure 39. Courses Page	54
Figure 40. Instructors Page	54
Figure 41. Lesson Page	55
Figure 42. Progress Page.....	55
Figure 43. Folder with .sln file.....	56
Figure 44. Website Server Launcher.....	57
Figure 45. Home Page.....	58
Figure 46. Sidebar for navigation	59
Figure 47. Navigated to Top Courses Page.....	60
Figure 48. Table Page 1 of view	61
Figure 49. Table page 2.....	62
Figure 50. Dropdown menu for Course Instructors	63
Figure 51. Dropdown after click	63
Figure 52. New table according to Course.....	64
Figure 53. Edit Button.....	65
Figure 54. Updating Values.....	66
Figure 55. Updated Value.....	67
Figure 56. Deleting Data.....	68
Figure 57. Alert on Delete.....	69
Figure 58. Deleted Value.....	69
Figure 59. Button to add new data	70

Figure 60. Form for data insertion	71
Figure 61. Inserting the data	71
Figure 62. New Instructor Created.....	72
Figure 63. Student Edit	73
Figure 64. Student Update	74
Figure 65. Updated Student	74
Figure 66. Inserting Student.....	75
Figure 67. Inserted New Student	75
Figure 68. Deleting Student data	76
Figure 69. Student Data Deleted.....	76
Figure 70. Course Edit	77
Figure 71. Course Update	77
Figure 72. Updated Course	78
Figure 73. Inserting Course.....	78
Figure 74. New Course Inserted	79
Figure 75. Deleting Course Data	79
Figure 76. Deleted Course Data.....	80
Figure 77. Instructor Edit.....	81
Figure 78. Instructor Update	81
Figure 79. Updated Instructor	82
Figure 80. Only one value on page 4	82
Figure 81. Adding new Instructor	83
Figure 82. Newly Added Instructor	83
Figure 83. Deleting Instructor Data	84
Figure 84. Deleted Instructor Data.....	84
Figure 85. Editing Lesson	85
Figure 86. Lesson Update	85
Figure 87. Updated Lesson	86
Figure 88. Inserting new Lesson	86
Figure 89. New lesson Inserted.....	87
Figure 90. Deleting Lesson Data	87

Figure 91. Deleted Lesson Data.....	88
Figure 92. Editing Progress.....	89
Figure 93. Updating Progress	89
Figure 94. Updated Progress.....	90
Figure 95. Adding new Progress	90
Figure 96. New Progress added	91
Figure 97. Deleting Progress data.....	91
Figure 98. Deleted Progress data	92
Figure 99. Enrollment Dropdown	93
Figure 100. Dropdown List.....	94
Figure 101. Updated Table.....	94
Figure 102. Course Instructor Dropdown	95
Figure 103. Course Instructor Dropdown list	95
Figure 104. New view table	96
Figure 105. Top Course Dropdown menu.....	97
Figure 106. Top Courses Dropdown List.....	98
Figure 107. New table view for Top Courses	98
Figure 108. Student with the Same Name	99
Figure 109. Displaying ID with the Student Name.....	100
Figure 110. Inserting into Progress	101
Figure 111. Parent Key Not Found	102
Figure 112. Dropdown to show parent key.....	103
Figure 113. Successful Insertion	104

List of Table

Table 1. Sample Table 1	6
Table 2. Sample Table 2	8
Table 3. Student Table.....	13
Table 4. Course Table.....	14
Table 5. Enrollment Table.....	15
Table 6. Instructor Table	16
Table 7. Course_Instructor Table	17
Table 8. Lesson Table.....	17
Table 9. Progress Table	18
Table 10. Content Table	19
Table 11. QA Table.....	20
Table 12. Testing Failure 1.....	99
Table 13. Testing Failure 2.....	101

1 Introduction

A wide range of students can access online learning courses and resources through e-learner, an online learning platform designed specifically for that purpose. Any student can sign up for several classes taught by admired teachers. A wide variety of materials and content are offered to the students so they understand each component better. In addition to tracking their progress in enrolled courses, enrolled students get the chance to learn from the finest. The project shows the use of WebForms for its implementation and showcases the ASP.NET framework.

The report showcases the whole process of building the project from scratch including topics such as normalization, database creation, data insertion and frontend development with user manual, testing and further discussions. Overall, the report provides an overview of the projects development process from start to finish.

2 Textual Analysis

2.1 Student-Course

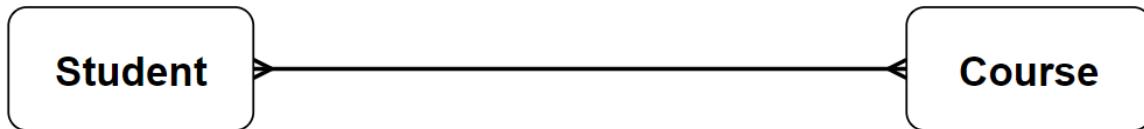


Figure 1. Student-Course Relation

Many student can enroll in many courses

2.2 Student-LessonProgress

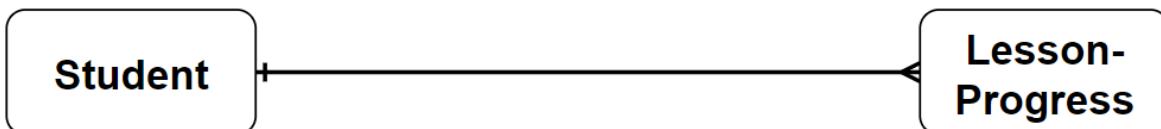


Figure 2. Student-LessonProgress Relation

One Student has many progress

2.3 Student-QA

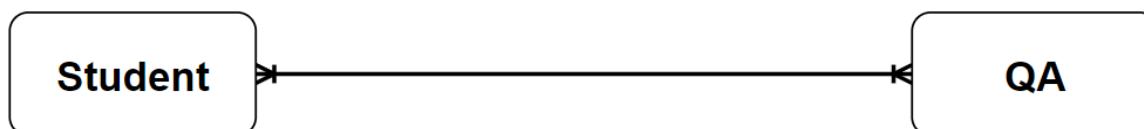


Figure 3. Student-QA relation

One or many students can have one or many QA

2.4 Course-QA

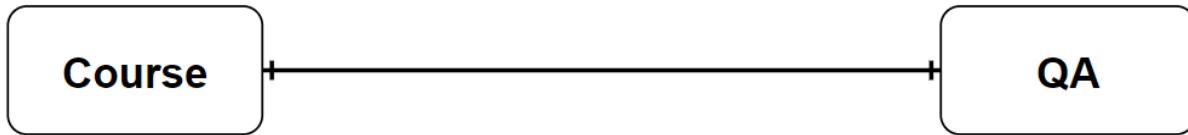


Figure 4. Course-QA Relation

A course can have one qa session

2.5 Course-Instructor

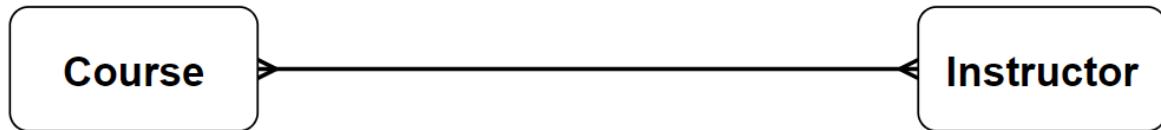


Figure 5. Course-Instructor Relation

Many courses can have many instructors

2.6 Course-Lesson

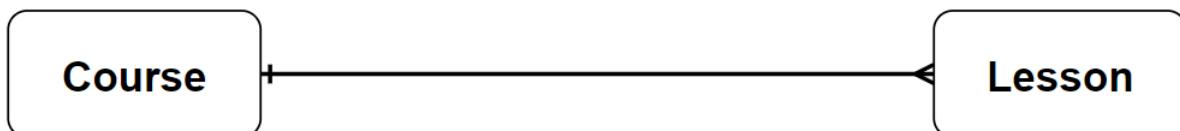


Figure 6. Course-Lesson Instructor

One course can have many lessons

2.7 Instructor-QA

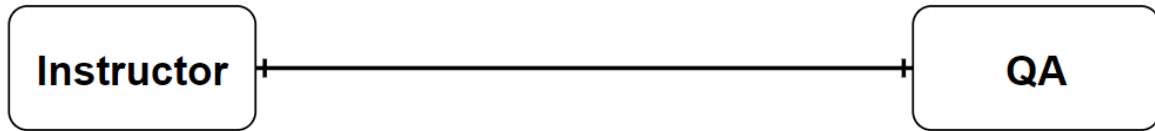


Figure 7. Instructor-QA Relation

One instructor can be involved in one qa session

2.8 Lesson-Content

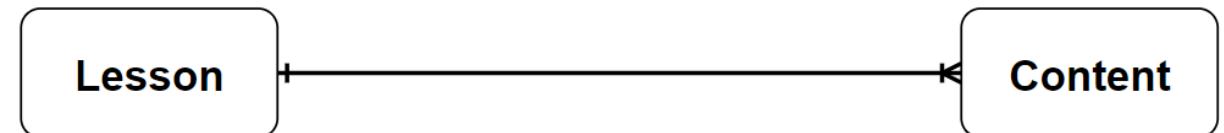


Figure 8. Lesson-Content Relation

One Lesson can have one content

2.9 Lesson-Progress

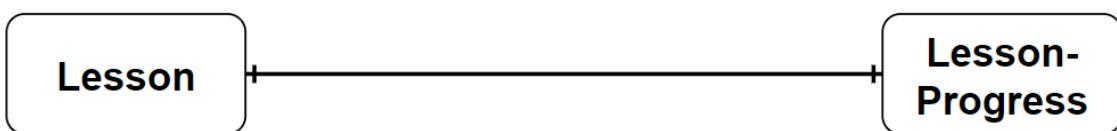


Figure 9. Lesson-LessonProgress Relation

One Lesson has one progress

3 ERD from Case Study

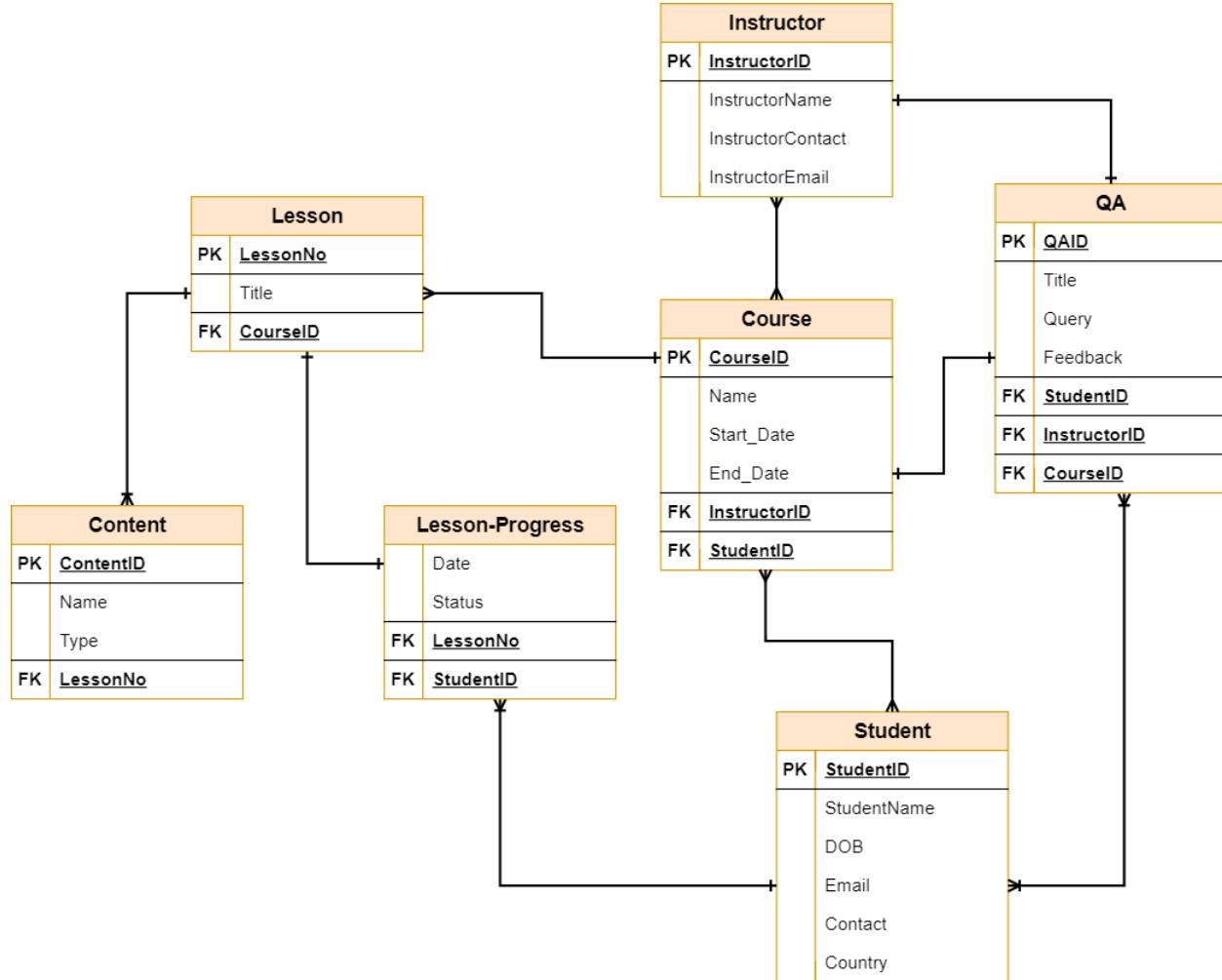


Figure 10. Initial ERD

4 Normalization

4.1 Normalization of fig 1

S.N .	Student Name	Contact	DOB	Email Address	Country	Enrol Course	Enrol Date
1	Dipesh Maharjan	98347362	1998-03-28	iamdipesh@gmail.com	Nepal	Java	2019-02-19
						Python	2020-05-18
2	Ronal Silwal	98436784	2000-01-09	ronaldo@gmail.com	UK	SQL for BI	2022-09-12
3	Prithivi Tandulkar	98343786	2002-09-13	pre@gmail.com		Java	2023-01-23
						Angular	2023-02-12
4	Dipeshor GC	98384754	1998-02-14	gcdip@gmail.com	USA	Python for Data Analysis	2022-01-29

Table 1. Sample Table 1

i. UNF

Student (StudentID, Student Name, Contact, DOB, Email, Address, Country {CourseName, EnrolDate, CourseID})

ii. 1NF

Separating repeating groups

Student (StudentID, Student Name, Contact, DOB, Email, Address, Country)

Course (StudentID *, CourseID, CourseName, EnrolDate)

iii. 2NF

Identifying Partial Functional Dependency

Student (StudentID, StudentName, Contact, DOB, Email, Contact, Country)

Course (StudentID *, CourseID, CourseName, EnrolDate)

For table Course:

StudentID, CourseID > EnrolDate

CourseID > CourseName

StudentID>

Course (CourseID, CourseName)

Enrollment (StudentID *, CourseID*, EnrolDate)

iv. 3NF

Student (StudentID, StudentName, Contact, DOB, Email, Contact, Country)

Course (CourseID, CourseName)

Enrollment (StudentID *, CourseID*, EnrolDate)

Already in 3NF

4.2 Normalization of 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_01 9	Java	1	Class and Constructor	Completed	2023-01-20	Teach_1	Dolma Gurung
Aadesh Lama	Java_01 9	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	Data Frame	Completed	2023-03-04	Teach_2	Samundra Karki

Table 2. Sample Table 2

i. UNF

Student (StudentID, Name, {CourseID, CourseName, Start_Date, End_Date, InstructorID, InstructorName, InstructorContact, InstructorEmail, {LessonNo, {Title, Status, Last_Accessed_Date} }})

ii. 1NF

Student (StudentID, Name)

Course (StudentID*, CourseID, CourseName, Start_Date, End_Date, InstructorID, InstructorName, InstructorContact, InstructorEmail)

Lesson-Progress (StudentID*, CourseID*, LessonNo, Title, Status, Last_Accessed_Date)

Separting Repeating Group

Each column has a primary key

iii. 2NF

For Course Table:

StudentID, CourseID >

CourseID > CourseName, Start_Date, End_Date

StudentID >

Assuming multiple courses have multiple instructors

For Lesson-Progress Table

CourseID, StudentID, LessonNo > Title, Status, LAD

Title, Status and LAD is dependent on the primary key

CourseID, StudentID >

Final Tables after 2NF

Student (StudentID, Name)

Student-Course (StudentID*, CourseID*)

Course (CourseID, CourseName, Start_Date, End_Date)

Course-Instructor (CourseID*, InstructorID*)

Instructor (InstructorID, InstructorName, InstructorContact, InstructorEmail)

Progress (StudentID*, CourseID*, LessonNo*, Title, Status, LAD)

iv. 3NF

For table Progress

CourseID > LessonNo > Title

StudentID >

StudentID , CourseID , LessonNo > Status, LAD

Final Tables after 3NF

Student (StudentID, Name)

Student-Course (StudentID*, CourseID*)

Course (CourseID, CourseName, Start_Date, End_Date)

Course-Instructor (CourseID*, InstructorID*)

Instructor (InstructorID, InstructorName, InstructorContact, InstructorEmail)

Progress (StudentID*, CourseID, LessonNo,Status, LAD)

Lesson (LessonNo, CourseID, Title)

5 Integration and Assumption

5.1 Assumptions

After normalizing the figures, the following assumptions can be drawn:

1. A student is enrolled into one or multiple courses
2. An instructor can teach multiple courses
3. A course can have multiple instructors
4. A lesson has one or many content
5. A QA session has only one query from one student
6. A QA session has only one feedback from one instructor
7. A QA session is taken by one instructor for one student
8. A QA session one feedback for one query
9. A student has one progress for a lesson
10. A lesson progress is completely dependent on the student, course and lesson no.

5.2 Integration

Student (StudentID, StudentName, DOB, Contact, Email, Country)

Student-Course (StudentID, CourseID)

Course (CourseID, CourseName, Start_Date, End_Date)

Course-Instructor (CourseID, InstructorID)

Instructor (InstructorID, InstructorName, InstructorContact, InstructorEmail)

Progress (StudentID, CourseID*, LessonNo, Status, LAD)

Lesson (CourseID, LessonNo, Title)

QA (QAIID, Title, Query, Feedback, StudentID, InstructorID, CourseID)

Contents (LessonNo*, ContentID, Name, Type, Lesson-Course*)

6 Final ERD

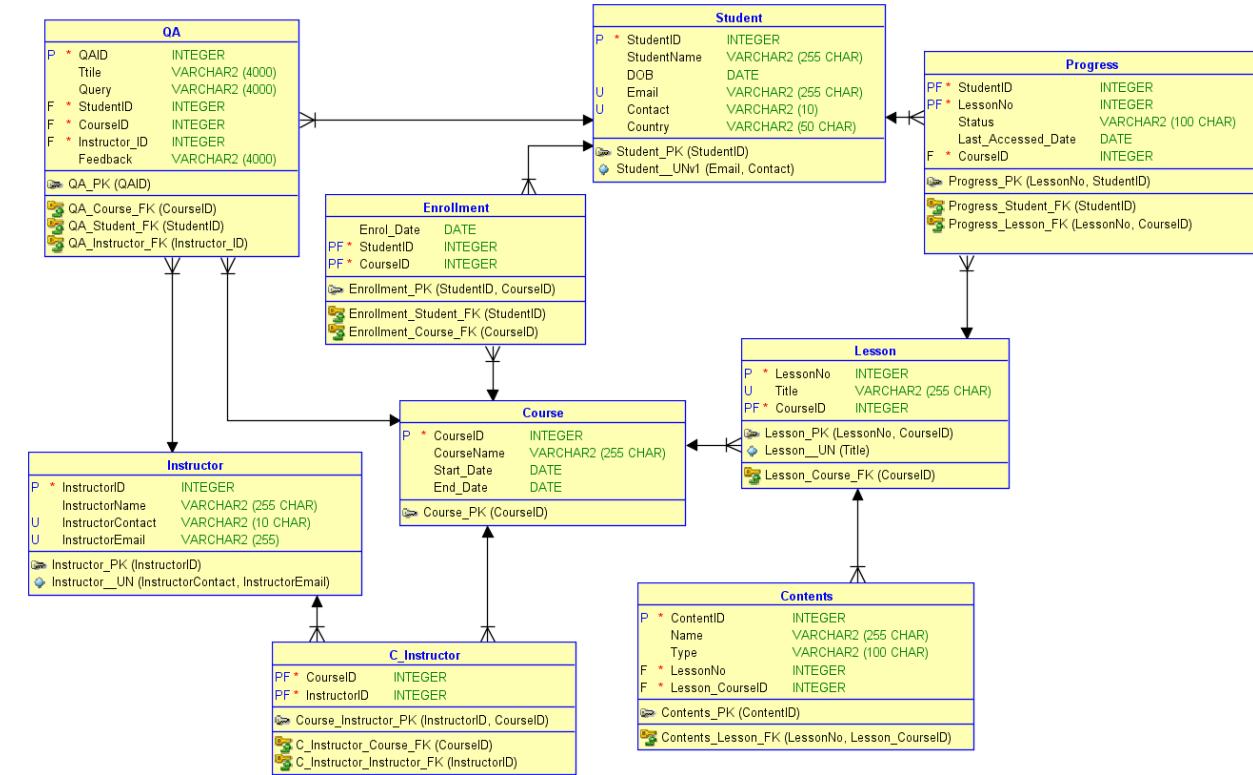


Figure 11. Final ERD

7 Data Dictionary

Table: Student

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
StudentID	INTEGER	10	PK			Unique Identifier for a student	1
StudentName	VARCHAR	255				To store name of the student	Aadesh Shrestha
DOB	DATE					To store birth date of the student	04-05-2003
Email	VARCHAR	255	UNIQUE			To store email address of student	aadeshshrestha 306@gmail.com
Contact	VARCHAR	10	UNIQUE			To store contact number of student	9797979797
Country	VARCHAR	50				To store country of origin of the student	Monaco

Table 3. Student Table

Table: Course

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
CourseID	INTEGER	10	PK			To store unique identifier for course	7
CoursetName	VARCHAR	255				To store Course Title	Java
Start_Date	DATE					To store the date when the course started	01-Jan-2002
End_Date	DATE					To store the date when the course ends	09-Dec-2002

Table 4. Course Table

Table: Enrollment

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
Enrol_Date	DATE					To store the date student enrolled for a course	11-Sept-2011
StudentID	INTEGER	10	PK, FK	Student	StudentID	To store the student id being enrolled	1
CourseID	INTEGER	10	PK, FK	Course	CourseID	To store the course id the student is being enrolled in	2

Table 5. Enrollment Table

Table: Instructor

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
InstructorID	INTEGER	10	PK			To store unique identifier for Instructor	101
InstructorName	VARCHAR	255				To store the name of the instructor	Madhav Kumar
InstructorContact	VARCHAR	10	UNIQUE			To store the contact number of the instructor	1212121212
InstructorEmail	VARCHAR	255	UNIQUE			To store the email address of the instructor	madhav@example.com

Table 6. Instructor Table

Table: C_Instructor

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
CourseID	INTEGER	10	PK, FK	Course	CourseID	To store the course id the instructor teaches	1
InstructorID	INTEGER	10	PK, FK	Instructor	InstructorID	To store the instructor of given courseid	101

*Table 7. Course_Instructor Table***Table: Lesson**

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
LessonNo	INTEGER	10	PK			To store unique no for each lesson	1
Title	VARCHAR	255				To store the name of the lesson	Python Basics
CourseID	INTEGER	10	PK, FK	Course	CourseID	To store the course	1

Table 8. Lesson Table

Table: Progress

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
StudentID	INTEGER	10	PK, FK	Student	StudentID	To store the student id	1
LessonNo	INTEGER	10	PK, FK	Lesson	LessonNo	To store the lesson for the student's progress	1
Status	VARCHAR	100				To store the status of the student in given course and lesson	Completed
Last_Accessed_Date	DATE					To store when the student last accessed the lesson	20-Jan-2024
CourseID	INTEGER	10	FK	Lesson	CourseID	To store the course id of the lesson the student is enrolled in	1

Table 9. Progress Table

Table: Contents

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
ContentID	INTEGER	10	PK			To store the unique number of contents	1
Name	VARCHAR	255				To store the name of the content	Introduction to Logic Video
Type	VARCHAR	100				To store the type of content	Video
LessonNo	INTEGER	10	FK	Lesson	Lesson_No	To store the lesson id for the respective content	1
Lesson_Course	INTEGER	10	FK	Lesson	CourseID	To store the course id of the lesson the content belongs to	1

Table 10. Content Table

Table: QA

Column Name	Datatype	Size	Constraint	Reference Table	Reference Column	Description	Example Data
QAID	INTEGER	10	PK			To store the unique identifier of the qa session	1
Title	VARCHAR	4000				To store the name for the qa session	Java Basics
Query	VARCHAR	4000				To store the question asked by students	What are the basic concepts of Java?
Feedback	INTEGER	10				To store the answers given by teachers	Java is an Object Oriented Language
StudentID	INTEGER	10	FK	Student	StudentID	To store the student id asking query	1
CourseID	INTEGER	10	FK	Course	CourseID	To store the course id	1
Instructor_ID	INTEGER	10	FK	Instructor	InstructorID	To store the id of instructor	101

Table 11. QA Table

8 Script

8.1.1 Student Table

```
CREATE TABLE student (
    studentid INTEGER NOT NULL,
    studentname VARCHAR2(255 CHAR),
    dob      DATE,
    email    VARCHAR2(255 CHAR),
    contact  VARCHAR2(10),
    country  VARCHAR2(50 CHAR)
);
```

```
ALTER TABLE student ADD CONSTRAINT student_pk PRIMARY KEY ( studentid );
```

```
ALTER TABLE student ADD CONSTRAINT student_unv1 UNIQUE ( email,
    contact );
```

8.1.2 Course Table

```
CREATE TABLE course (
    courseid INTEGER NOT NULL,
    coursename VARCHAR2(255 CHAR),
    start_date DATE,
    end_date  DATE
);
```

```
ALTER TABLE course ADD CONSTRAINT course_pk PRIMARY KEY ( courseid );
```

8.1.3 Enrollment Table

```
CREATE TABLE enrollment (
```

```
    enrol_date DATE,  
    studentid INTEGER NOT NULL,  
    courseid INTEGER NOT NULL  
);
```

```
ALTER TABLE enrollment ADD CONSTRAINT enrollment_pk PRIMARY KEY ( studentid,
```

```
    courseid );
```

```
ALTER TABLE enrollment
```

```
    ADD CONSTRAINT enrollment_course_fk FOREIGN KEY ( courseid )
```

```
        REFERENCES course ( courseid );
```

```
ALTER TABLE enrollment
```

```
    ADD CONSTRAINT enrollment_student_fk FOREIGN KEY ( studentid )
```

```
        REFERENCES student ( studentid );
```

8.1.4 Instructor Table

```
CREATE TABLE instructor (
```

```
    instructorid INTEGER NOT NULL,  
    instructorname VARCHAR2(255 CHAR),
```

```
instructorcontact VARCHAR2(10 CHAR),  
instructoremail VARCHAR2(255)  
);
```

```
ALTER TABLE instructor ADD CONSTRAINT instructor_pk PRIMARY KEY ( instructorid );
```

```
ALTER TABLE instructor ADD CONSTRAINT instructor_un UNIQUE ( instructorcontact,  
instructoremail );
```

8.1.5 Course Instructor Table

```
CREATE TABLE c_instructor (  
courseid INTEGER NOT NULL,  
instructorid INTEGER NOT NULL  
);
```

```
ALTER TABLE c_instructor ADD CONSTRAINT course_instructor_pk PRIMARY KEY (   
instructorid,  
courseid );
```

```
ALTER TABLE c_instructor  
ADD CONSTRAINT c_instructor_course_fk FOREIGN KEY ( courseid )  
REFERENCES course ( courseid );
```

```
ALTER TABLE c_instructor
```

```
    ADD CONSTRAINT c_instructor_instructor_fk FOREIGN KEY ( instructorid )
```

```
        REFERENCES instructor ( instructorid );
```

8.1.6 QA Table

```
CREATE TABLE qa (
```

```
    qaid      INTEGER NOT NULL,
```

```
    ttitle    VARCHAR2(255),
```

```
    query     VARCHAR2(4000),
```

```
    studentid  INTEGER NOT NULL,
```

```
    courseid   INTEGER NOT NULL,
```

```
    instructor_id INTEGER NOT NULL,
```

```
    feedback   VARCHAR2(4000)
```

```
);
```

```
ALTER TABLE qa ADD CONSTRAINT qa_pk PRIMARY KEY ( qaid );
```

```
ALTER TABLE qa
```

```
    ADD CONSTRAINT qa_course_fk FOREIGN KEY ( courseid )
```

```
        REFERENCES course ( courseid );
```

```
ALTER TABLE qa
```

```
    ADD CONSTRAINT qa_instructor_fk FOREIGN KEY ( instructor_id )
```

```
REFERENCES instructor ( instructorid );
```

```
ALTER TABLE qa
```

```
ADD CONSTRAINT qa_student_fk FOREIGN KEY ( studentid )
```

```
REFERENCES student ( studentid );
```

8.1.7 Lesson table

```
CREATE TABLE lesson (
```

```
lessonno INTEGER NOT NULL,
```

```
title VARCHAR2(255 CHAR),
```

```
courseid INTEGER NOT NULL
```

```
);
```

```
ALTER TABLE lesson ADD CONSTRAINT lesson_pk PRIMARY KEY ( lessonno,
```

```
courseid );
```

```
ALTER TABLE lesson ADD CONSTRAINT lesson_un UNIQUE ( title );
```

```
ALTER TABLE lesson
```

```
ADD CONSTRAINT lesson_course_fk FOREIGN KEY ( courseid )
```

```
REFERENCES course ( courseid );
```

8.1.8 Content Table

```
CREATE TABLE contents (
    contentid      INTEGER NOT NULL,
    name           VARCHAR2(255 CHAR),
    type           VARCHAR2(100 CHAR),
    lessonno       INTEGER NOT NULL,
    lesson_courseid INTEGER NOT NULL
);
```

```
ALTER TABLE contents ADD CONSTRAINT contents_pk PRIMARY KEY ( contentid );
```

```
ALTER TABLE contents
```

```
    ADD CONSTRAINT contents_lesson_fk FOREIGN KEY ( lessonno,
                                                lesson_courseid )
    REFERENCES lesson ( lessonno,
                        courseid );
```

8.1.9 Progress Table

```
CREATE TABLE progress (
    studentid      INTEGER NOT NULL,
    courseid       INTEGER NOT NULL,
    lessonno       INTEGER NOT NULL,
    status          VARCHAR2(100 CHAR),
```

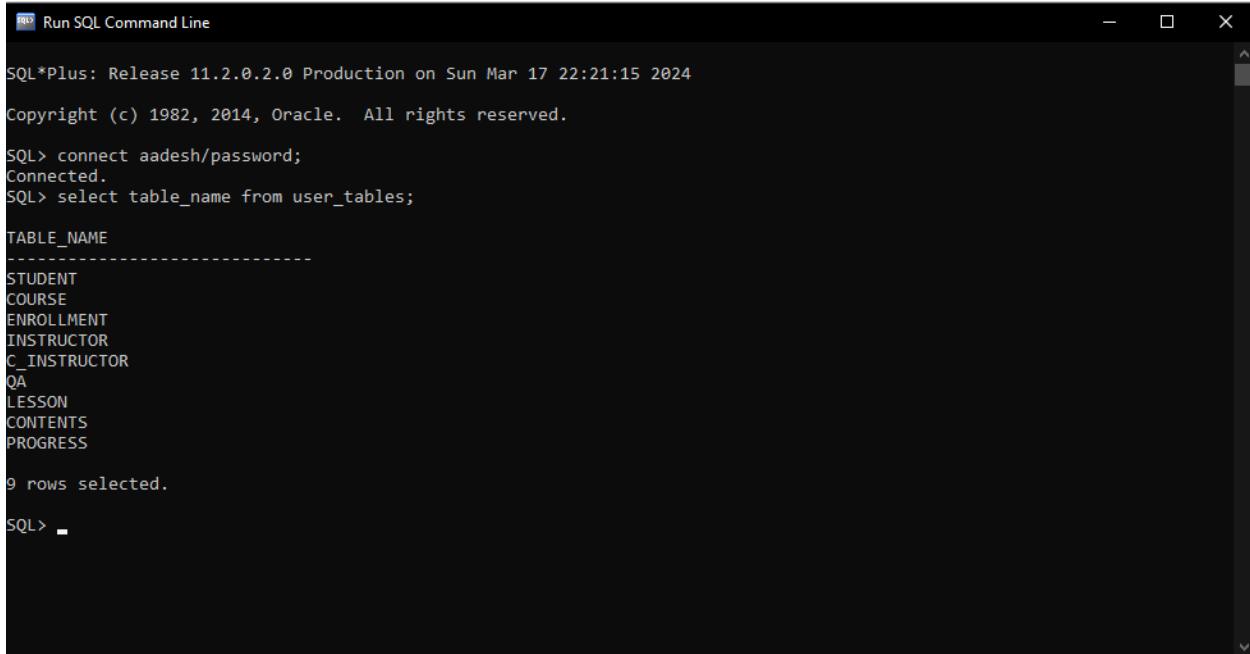
```
last Accessed Date  
);
```

```
ALTER TABLE progress ADD CONSTRAINT progress_pk PRIMARY KEY ( lessonno,  
studentid );
```

```
ALTER TABLE progress  
ADD CONSTRAINT progress_lesson_fk FOREIGN KEY ( lessonno,  
courseid )
```

```
REFERENCES lesson ( lessonno,  
courseid );
```

```
ALTER TABLE progress  
ADD CONSTRAINT progress_student_fk FOREIGN KEY ( studentid )  
REFERENCES student ( studentid );
```



Run SQL Command Line

SQL*Plus: Release 11.2.0.2.0 Production on Sun Mar 17 22:21:15 2024

Copyright (c) 1982, 2014, Oracle. All rights reserved.

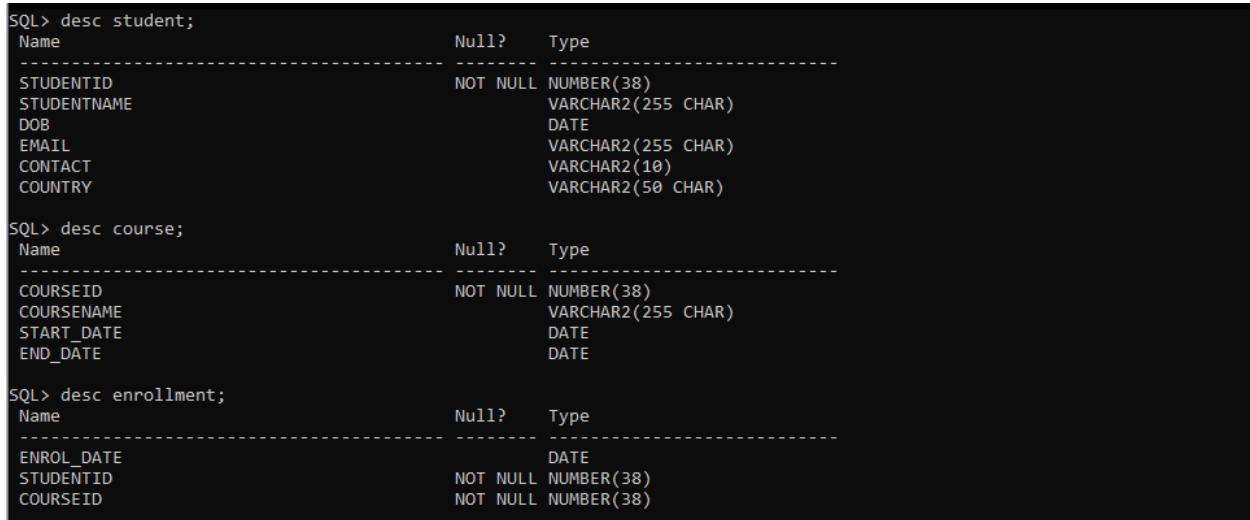
SQL> connect aadesh/password;
Connected.
SQL> select table_name from user_tables;

TABLE_NAME
STUDENT
COURSE
ENROLLMENT
INSTRUCTOR
C_INSTRUCTOR
QA
LESSON
CONTENTS
PROGRESS

9 rows selected.

SQL> -

Figure 12. Connecting User



SQL> desc student;

Name	Null?	Type
STUDENTID	NOT NULL	NUMBER(38)
STUDENTNAME		VARCHAR2(255 CHAR)
DOB		DATE
EMAIL		VARCHAR2(255 CHAR)
CONTACT		VARCHAR2(10)
COUNTRY		VARCHAR2(50 CHAR)

SQL> desc course;

Name	Null?	Type
COURSEID	NOT NULL	NUMBER(38)
COURSENAMES		VARCHAR2(255 CHAR)
START_DATE		DATE
END_DATE		DATE

SQL> desc enrollment;

Name	Null?	Type
ENROL_DATE		DATE
STUDENTID	NOT NULL	NUMBER(38)
COURSEID	NOT NULL	NUMBER(38)

Figure 13. Table Descriptions(1)

```

SQL> desc instructor;
Name          Null?    Type
-----
INSTRUCTORID      NOT NULL NUMBER(38)
INSTRUCTORMNAME   VARCHAR2(255 CHAR)
INSTRUCTORCONTACT VARCHAR2(10 CHAR)
INSTRUCTOREMAIL    VARCHAR2(255)

SQL> desc c_instructor;
Name          Null?    Type
-----
COURSEID      NOT NULL NUMBER(38)
INSTRUCTORID  NOT NULL NUMBER(38)

SQL> desc qa;
Name          Null?    Type
-----
QAID          NOT NULL NUMBER(38)
TTITLE        VARCHAR2(255)
QUERY         VARCHAR2(4000)
STUDENTID     NOT NULL NUMBER(38)
COURSEID      NOT NULL NUMBER(38)
INSTRUCTOR_ID  NOT NULL NUMBER(38)
FEEDBACK      VARCHAR2(4000)

```

Figure 14. Table Descriptions(2)

```

SQL> desc lesson;
Name          Null?    Type
-----
LESSONNO      NOT NULL NUMBER(38)
TITLE         VARCHAR2(255 CHAR)
COURSEID      NOT NULL NUMBER(38)

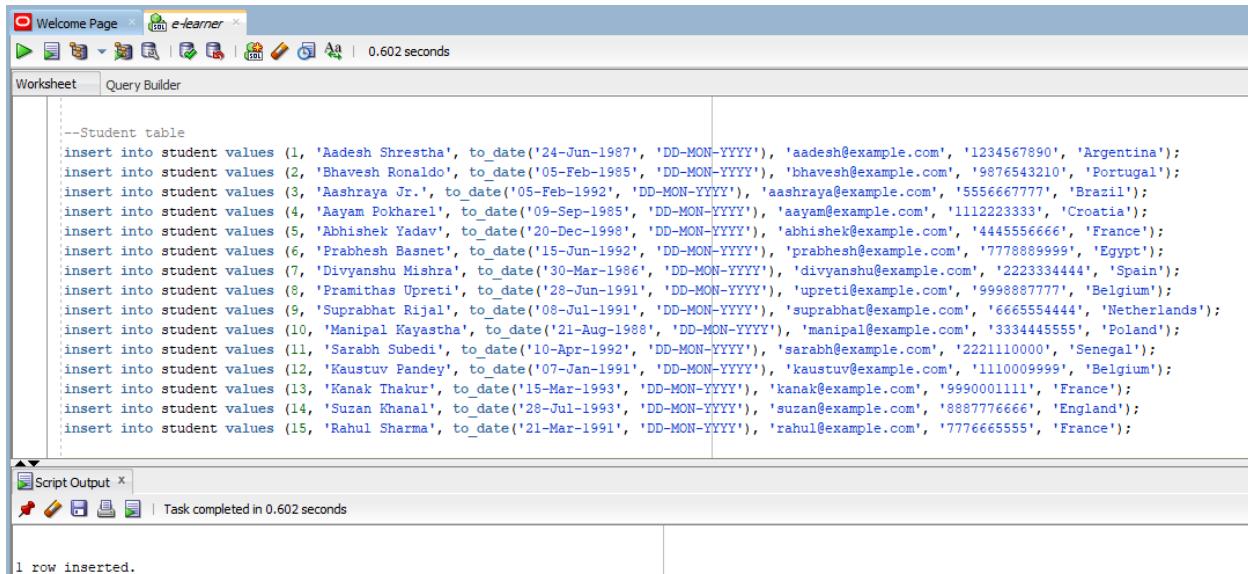
SQL> desc contents;
Name          Null?    Type
-----
CONTENTID    NOT NULL NUMBER(38)
NAME          VARCHAR2(255 CHAR)
TYPE          VARCHAR2(100 CHAR)
LESSONNO     NOT NULL NUMBER(38)
LESSON_COURSEID NOT NULL NUMBER(38)

SQL> desc progress;
Name          Null?    Type
-----
STUDENTID    NOT NULL NUMBER(38)
COURSEID     NOT NULL NUMBER(38)
LESSONNO     NOT NULL NUMBER(38)
STATUS        VARCHAR2(100 CHAR)
LAST_ACCESSED_DATE DATE

```

Figure 15. Table Descriptions(3)

9 Insert Statement

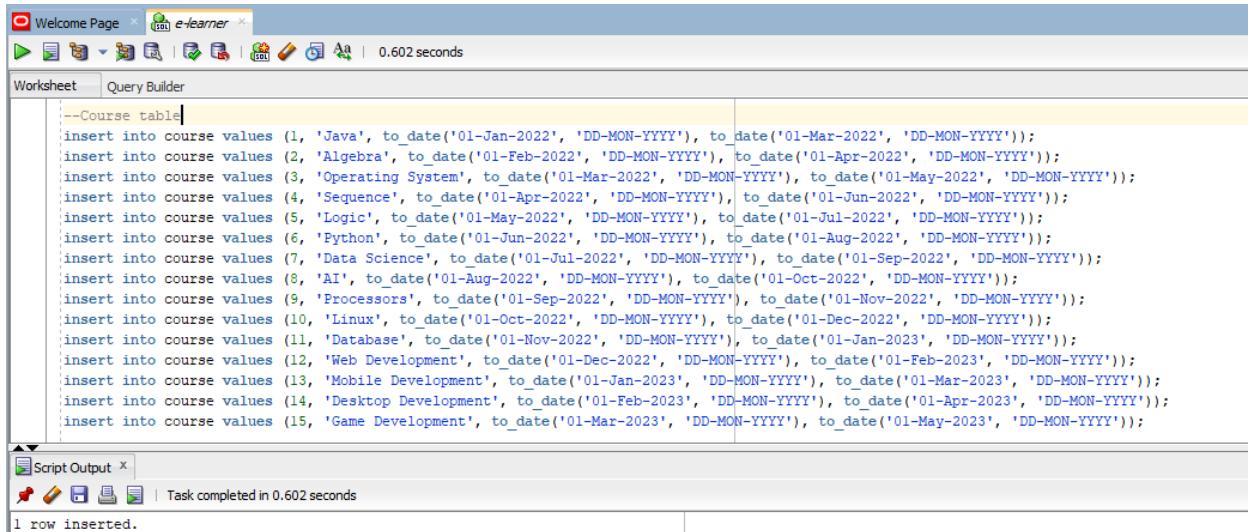


The screenshot shows the e-learner software interface with a query worksheet containing the following SQL code:

```
--Student table
insert into student values (1, 'Aadesh Shrestha', to_date('24-Jun-1987', 'DD-MON-YYYY'), 'aadesh@example.com', '1234567890', 'Argentina');
insert into student values (2, 'Bhavesh Ronaldo', to_date('05-Feb-1985', 'DD-MON-YYYY'), 'bhavesh@example.com', '9876543210', 'Portugal');
insert into student values (3, 'Aashraya Jr.', to_date('05-Feb-1992', 'DD-MON-YYYY'), 'aashraya@example.com', '5556667777', 'Brazil');
insert into student values (4, 'Aayam Pokharel', to_date('09-Sep-1985', 'DD-MON-YYYY'), 'aayam@example.com', '1112223333', 'Croatia');
insert into student values (5, 'Abhishek Yadav', to_date('20-Dec-1998', 'DD-MON-YYYY'), 'abhishek@example.com', '4445556666', 'France');
insert into student values (6, 'Prabhesh Basnet', to_date('15-Jun-1992', 'DD-MON-YYYY'), 'prabhesh@example.com', '7778889999', 'Egypt');
insert into student values (7, 'Divyanshu Mishra', to_date('30-Mar-1986', 'DD-MON-YYYY'), 'divyanshu@example.com', '2223334444', 'Spain');
insert into student values (8, 'Pramithas Upreti', to_date('28-Jun-1991', 'DD-MON-YYYY'), 'upreti@example.com', '9998887777', 'Belgium');
insert into student values (9, 'Suprabhat Rijal', to_date('08-Jul-1991', 'DD-MON-YYYY'), 'suprabhat@example.com', '6665554444', 'Netherlands');
insert into student values (10, 'Manipal Kayaatha', to_date('21-Aug-1988', 'DD-MON-YYYY'), 'manipal@example.com', '3334445555', 'Poland');
insert into student values (11, 'Sarahb Subedi', to_date('10-Apr-1992', 'DD-MON-YYYY'), 'sarabh@example.com', '2221110000', 'Senegal');
insert into student values (12, 'Kaustuv Pandey', to_date('07-Jan-1991', 'DD-MON-YYYY'), 'kaustuv@example.com', '1110009999', 'Belgium');
insert into student values (13, 'Kanak Thakur', to_date('15-Mar-1993', 'DD-MON-YYYY'), 'kanak@example.com', '9990001111', 'France');
insert into student values (14, 'Suzan Khanal', to_date('28-Jul-1993', 'DD-MON-YYYY'), 'suzan@example.com', '8887776666', 'England');
insert into student values (15, 'Rahul Sharma', to_date('21-Mar-1991', 'DD-MON-YYYY'), 'rahul@example.com', '7776665555', 'France');
```

The script output shows "1 row inserted." and the task completed in 0.602 seconds.

Figure 16. Student table insertion



The screenshot shows the e-learner software interface with a query worksheet containing the following SQL code:

```
--Course table
insert into course values (1, 'Java', to_date('01-Jan-2022', 'DD-MON-YYYY'), to_date('01-Mar-2022', 'DD-MON-YYYY'));
insert into course values (2, 'Algebra', to_date('01-Feb-2022', 'DD-MON-YYYY'), to_date('01-Apr-2022', 'DD-MON-YYYY'));
insert into course values (3, 'Operating System', to_date('01-Mar-2022', 'DD-MON-YYYY'), to_date('01-May-2022', 'DD-MON-YYYY'));
insert into course values (4, 'Sequence', to_date('01-Apr-2022', 'DD-MON-YYYY'), to_date('01-Jun-2022', 'DD-MON-YYYY'));
insert into course values (5, 'Logic', to_date('01-May-2022', 'DD-MON-YYYY'), to_date('01-Jul-2022', 'DD-MON-YYYY'));
insert into course values (6, 'Python', to_date('01-Jun-2022', 'DD-MON-YYYY'), to_date('01-Aug-2022', 'DD-MON-YYYY'));
insert into course values (7, 'Data Science', to_date('01-Jul-2022', 'DD-MON-YYYY'), to_date('01-Sep-2022', 'DD-MON-YYYY'));
insert into course values (8, 'AI', to_date('01-Aug-2022', 'DD-MON-YYYY'), to_date('01-Oct-2022', 'DD-MON-YYYY'));
insert into course values (9, 'Processors', to_date('01-Sep-2022', 'DD-MON-YYYY'), to_date('01-Nov-2022', 'DD-MON-YYYY'));
insert into course values (10, 'Linux', to_date('01-Oct-2022', 'DD-MON-YYYY'), to_date('01-Dec-2022', 'DD-MON-YYYY'));
insert into course values (11, 'Database', to_date('01-Nov-2022', 'DD-MON-YYYY'), to_date('01-Jan-2023', 'DD-MON-YYYY'));
insert into course values (12, 'Web Development', to_date('01-Dec-2022', 'DD-MON-YYYY'), to_date('01-Feb-2023', 'DD-MON-YYYY'));
insert into course values (13, 'Mobile Development', to_date('01-Jan-2023', 'DD-MON-YYYY'), to_date('01-Mar-2023', 'DD-MON-YYYY'));
insert into course values (14, 'Desktop Development', to_date('01-Feb-2023', 'DD-MON-YYYY'), to_date('01-Apr-2023', 'DD-MON-YYYY'));
insert into course values (15, 'Game Development', to_date('01-Mar-2023', 'DD-MON-YYYY'), to_date('01-May-2023', 'DD-MON-YYYY'));
```

The script output shows "1 row inserted." and the task completed in 0.602 seconds.

Figure 17. Course table insertion

The screenshot shows the e-learner software interface. At the top, there are tabs for 'Welcome Page' and 'e-learner'. Below the tabs is a toolbar with various icons. The main area is divided into two panes: 'Worksheet' and 'Query Builder'. The 'Worksheet' pane contains an SQL script for inserting data into an 'instructor' table. The 'Query Output' pane at the bottom shows the result of the insertion task.

```
--Instructor table
insert into instructor values (101, 'Madhav Kumar', '1234567890', 'madhav@example.com');
insert into instructor values (102, 'Purna Singh', '2345678901', 'purna@example.com');
insert into instructor values (103, 'Jyotita Magar', '3456789012', 'jyotita@example.com');
insert into instructor values (104, 'Kamala Harris', '4567890123', 'kamala@example.com');
insert into instructor values (105, 'Susma Adhikari', '5678901234', 'susma@example.com');
insert into instructor values (106, 'Anjali Adhikari', '6789012345', 'anjali@example.com');
insert into instructor values (107, 'Krishna Thaku', '7890123456', 'krishna@example.com');
insert into instructor values (108, 'Anju Shrestha', '8901234567', 'anju@example.com');
insert into instructor values (109, 'Man Bahadur', '9012345678', 'man@example.com');
insert into instructor values (110, 'Keshab Yadav', '0123456789', 'keshab@example.com');
insert into instructor values (111, 'Kailash Raut', '1111111111', 'kailash@example.com');
insert into instructor values (112, 'Tilak Raj', '2222222222', 'tilak@example.com');
insert into instructor values (113, 'Arjun Pandey', '3333333333', 'arjun@example.com');
insert into instructor values (114, 'Badal Neupane', '4444444444', 'badal@example.com');
insert into instructor values (115, 'Uzzal Phuyal', '5555555555', 'uzzal@example.com');
```

Script Output | Task completed in 0.602 seconds

1 row inserted.

Figure 18. Instructor table insertion

The screenshot shows the MySQL Workbench interface. The main window is titled 'e-learner' and contains a 'Worksheet' tab with the following SQL code:

```
--Course_Instructor table
insert into c_instructor values (1, 101);
insert into c_instructor values (2, 102);
insert into c_instructor values (3, 103);
insert into c_instructor values (4, 104);
insert into c_instructor values (5, 105);
insert into c_instructor values (6, 106);
insert into c_instructor values (7, 107);
insert into c_instructor values (8, 108);
insert into c_instructor values (9, 109);
insert into c_instructor values (10, 110);
insert into c_instructor values (11, 111);
insert into c_instructor values (12, 112);
insert into c_instructor values (13, 113);
insert into c_instructor values (14, 114);
insert into c_instructor values (15, 115);
insert into c_instructor values (1, 107);
insert into c_instructor values (2, 101);
insert into c_instructor values (3, 101);
insert into c_instructor values (4, 106);
insert into c_instructor values (5, 108);
insert into c_instructor values (6, 107);
insert into c_instructor values (7, 109);
insert into c_instructor values (8, 101);
insert into c_instructor values (9, 102);
insert into c_instructor values (10, 111);
insert into c_instructor values (11, 101);
insert into c_instructor values (12, 104);
insert into c_instructor values (13, 103);
insert into c_instructor values (14, 107);
insert into c_instructor values (15, 106);
```

The bottom panel is titled 'Script Output' and displays the message: 'Task completed in 0.602 seconds'. Below this message, it says '1 row inserted.'

Figure 19. Course_Instructor table insertion

The screenshot shows the MySQL Workbench interface. The top bar displays 'Welcome Page' and 'e-learner'. The main area has tabs for 'Worksheet' and 'Query Builder', with 'Worksheet' selected. The code in the worksheet is:

```
--Enrollment table
insert into enrollment values (to_date('2022-01-01', 'YYYY-MM-DD'), 1, 1);
insert into enrollment values (to_date('2022-01-01', 'YYYY-MM-DD'), 1, 2);
insert into enrollment values (to_date('2022-02-01', 'YYYY-MM-DD'), 2, 1);
insert into enrollment values (to_date('2022-02-01', 'YYYY-MM-DD'), 2, 2);
insert into enrollment values (to_date('2022-03-01', 'YYYY-MM-DD'), 3, 1);
insert into enrollment values (to_date('2022-03-01', 'YYYY-MM-DD'), 3, 2);
insert into enrollment values (to_date('2022-04-01', 'YYYY-MM-DD'), 4, 1);
insert into enrollment values (to_date('2022-04-01', 'YYYY-MM-DD'), 4, 2);
insert into enrollment values (to_date('2022-05-01', 'YYYY-MM-DD'), 5, 1);
insert into enrollment values (to_date('2022-05-01', 'YYYY-MM-DD'), 5, 2);
insert into enrollment values (to_date('2022-06-01', 'YYYY-MM-DD'), 6, 1);
insert into enrollment values (to_date('2022-06-01', 'YYYY-MM-DD'), 6, 2);
insert into enrollment values (to_date('2022-07-01', 'YYYY-MM-DD'), 7, 1);
insert into enrollment values (to_date('2022-07-01', 'YYYY-MM-DD'), 7, 2);
insert into enrollment values (to_date('2022-08-01', 'YYYY-MM-DD'), 8, 1);
insert into enrollment values (to_date('2022-08-01', 'YYYY-MM-DD'), 8, 2);
insert into enrollment values (to_date('2022-05-01', 'YYYY-MM-DD'), 5, 3);
insert into enrollment values (to_date('2022-05-01', 'YYYY-MM-DD'), 5, 5);
insert into enrollment values (to_date('2022-06-01', 'YYYY-MM-DD'), 6, 3);
insert into enrollment values (to_date('2022-06-01', 'YYYY-MM-DD'), 6, 5);
insert into enrollment values (to_date('2022-07-01', 'YYYY-MM-DD'), 7, 7);
insert into enrollment values (to_date('2022-07-01', 'YYYY-MM-DD'), 7, 3);
insert into enrollment values (to_date('2022-08-01', 'YYYY-MM-DD'), 8, 9);
insert into enrollment values (to_date('2022-08-01', 'YYYY-MM-DD'), 8, 5);
insert into enrollment values (to_date('2022-09-01', 'YYYY-MM-DD'), 9, 1);
insert into enrollment values (to_date('2022-09-01', 'YYYY-MM-DD'), 9, 2);
insert into enrollment values (to_date('2022-10-01', 'YYYY-MM-DD'), 10, 1);
insert into enrollment values (to_date('2022-10-01', 'YYYY-MM-DD'), 10, 2);
insert into enrollment values (to_date('2022-11-01', 'YYYY-MM-DD'), 11, 1);
insert into enrollment values (to_date('2022-11-01', 'YYYY-MM-DD'), 11, 2);
insert into enrollment values (to_date('2022-12-01', 'YYYY-MM-DD'), 14, 1);
```

The bottom panel shows the 'Script Output' tab with the message: 'Task completed in 0.602 seconds'. Below it, the output shows: '1 row inserted.'

Figure 20. Enrollment Table insertion

```
--Lesson table
insert into lesson values (1, 'Introduction to Java Programming', 1);
insert into lesson values (2, 'Java Advanced Concepts', 1);
insert into lesson values (3, 'Basics of Algebra', 2);
insert into lesson values (4, 'Advanced Algebra Techniques', 2);
insert into lesson values (5, 'Operating System Fundamentals', 3);
insert into lesson values (6, 'Advanced Operating System Concepts', 3);
insert into lesson values (7, 'Arithmetic Sequences', 4);
insert into lesson values (8, 'Geometric Sequences', 4);
insert into lesson values (9, 'Introduction to Logic', 5);
insert into lesson values (10, 'Advanced Logic and Reasoning', 5);
insert into lesson values (11, 'Python Basics', 6);
insert into lesson values (12, 'Python Advanced Concepts', 6);
insert into lesson values (13, 'Introduction to Data Science', 7);
insert into lesson values (14, 'Data Science Tools and Techniques', 7);
insert into lesson values (15, 'Introduction to Artificial Intelligence', 8);
insert into lesson values (16, 'Advanced AI Applications', 8);
insert into lesson values (17, 'Processor Architecture Fundamentals', 9);
insert into lesson values (18, 'Advanced Processor Concepts', 9);
insert into lesson values (19, 'Linux Basics', 10);
insert into lesson values (20, 'Linux System Administration', 10);
insert into lesson values (21, 'Database Design Principles', 11);
insert into lesson values (22, 'Advanced Database Management', 11);
insert into lesson values (23, 'Web Development Basics', 12);
insert into lesson values (24, 'Advanced Web Development Concepts', 12);
insert into lesson values (25, 'Introduction to Mobile Development', 13);
insert into lesson values (26, 'Mobile App Design and Development', 13);
insert into lesson values (27, 'Desktop Development Fundamentals', 14);
insert into lesson values (28, 'Advanced Desktop Application Development', 14);
insert into lesson values (29, 'Introduction to Game Development', 15);
insert into lesson values (30, 'Game Design and Development Techniques', 15);
```

Script Output | Task completed in 2.419 seconds

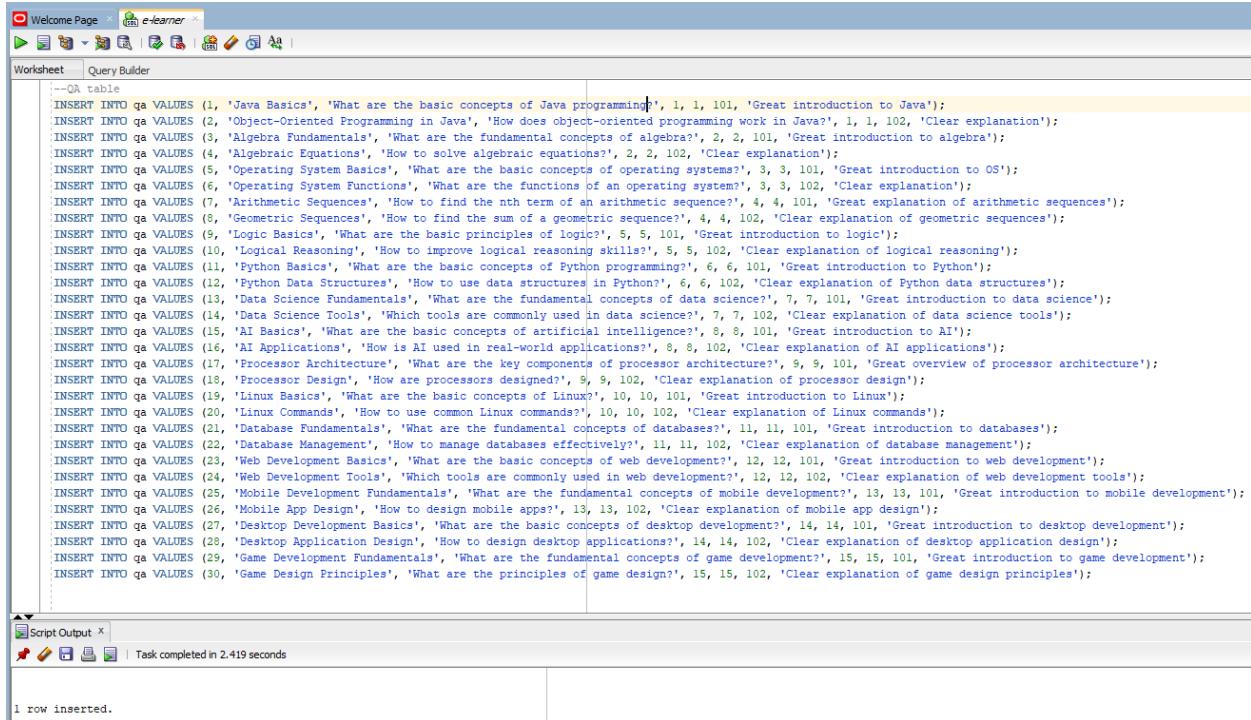
1 row inserted.

Figure 21. Lesson Table insertion

The screenshot shows the MySQL Workbench application. The main window has tabs for 'Worksheet' and 'Query Builder'. The 'Worksheet' tab is active, displaying a SQL script for inserting data into a 'contents' table. The script includes a comment '--Contents table' and 30 insert statements. The last insert statement, which inserts row 30, is highlighted with a yellow background. Below the worksheet is a 'Script Output' window showing the result of the execution: '1 row inserted.' and a note 'Task completed in 2.419 seconds'.

```
--Contents table
insert into contents values (1, 'Introduction to Java Programming Video', 'video', 1, 1);
insert into contents values (2, 'Java Advanced Concepts Text', 'text', 2, 1);
insert into contents values (3, 'Basics of Algebra Text', 'text', 3, 2);
insert into contents values (4, 'Advanced Algebra Techniques Image', 'image', 4, 2);
insert into contents values (5, 'Operating System Fundamentals Video', 'video', 5, 3);
insert into contents values (6, 'Advanced Operating System Concepts Text', 'text', 6, 3);
insert into contents values (7, 'Arithmetic Sequences Text', 'text', 7, 4);
insert into contents values (8, 'Geometric Sequences Video', 'video', 8, 4);
insert into contents values (9, 'Introduction to Logic Video', 'video', 9, 5);
insert into contents values (10, 'Advanced Logic and Reasoning Text', 'text', 10, 5);
insert into contents values (11, 'Python Basics Text', 'text', 11, 6);
insert into contents values (12, 'Python Advanced Concepts Image', 'image', 12, 6);
insert into contents values (13, 'Introduction to Data Science Video', 'video', 13, 7);
insert into contents values (14, 'Data Science Tools and Techniques Text', 'text', 14, 7);
insert into contents values (15, 'Introduction to Artificial Intelligence Video', 'video', 15, 8);
insert into contents values (16, 'Advanced AI Applications Text', 'text', 16, 8);
insert into contents values (17, 'Processor Architecture Fundamentals Video', 'video', 17, 9);
insert into contents values (18, 'Advanced Processor Concepts Image', 'image', 18, 9);
insert into contents values (19, 'Linux Basics Text', 'text', 19, 10);
insert into contents values (20, 'Linux System Administration Video', 'video', 20, 10);
insert into contents values (21, 'Database Design Principles Video', 'video', 21, 11);
insert into contents values (22, 'Advanced Database Management Text', 'text', 22, 11);
insert into contents values (23, 'Web Development Basics Text', 'text', 23, 12);
insert into contents values (24, 'Advanced Web Development Concepts Image', 'image', 24, 12);
insert into contents values (25, 'Introduction to Mobile Development Video', 'video', 25, 13);
insert into contents values (26, 'Mobile App Design and Development Text', 'text', 26, 13);
insert into contents values (27, 'Desktop Development Fundamentals Video', 'video', 27, 14);
insert into contents values (28, 'Advanced Desktop Application Development Image', 'image', 28, 14);
insert into contents values (29, 'Introduction to Game Development Video', 'video', 29, 15);
insert into contents values (30, 'Game Design and Development Techniques Text', 'text', 30, 15);
```

Figure 22. Content Table insertion

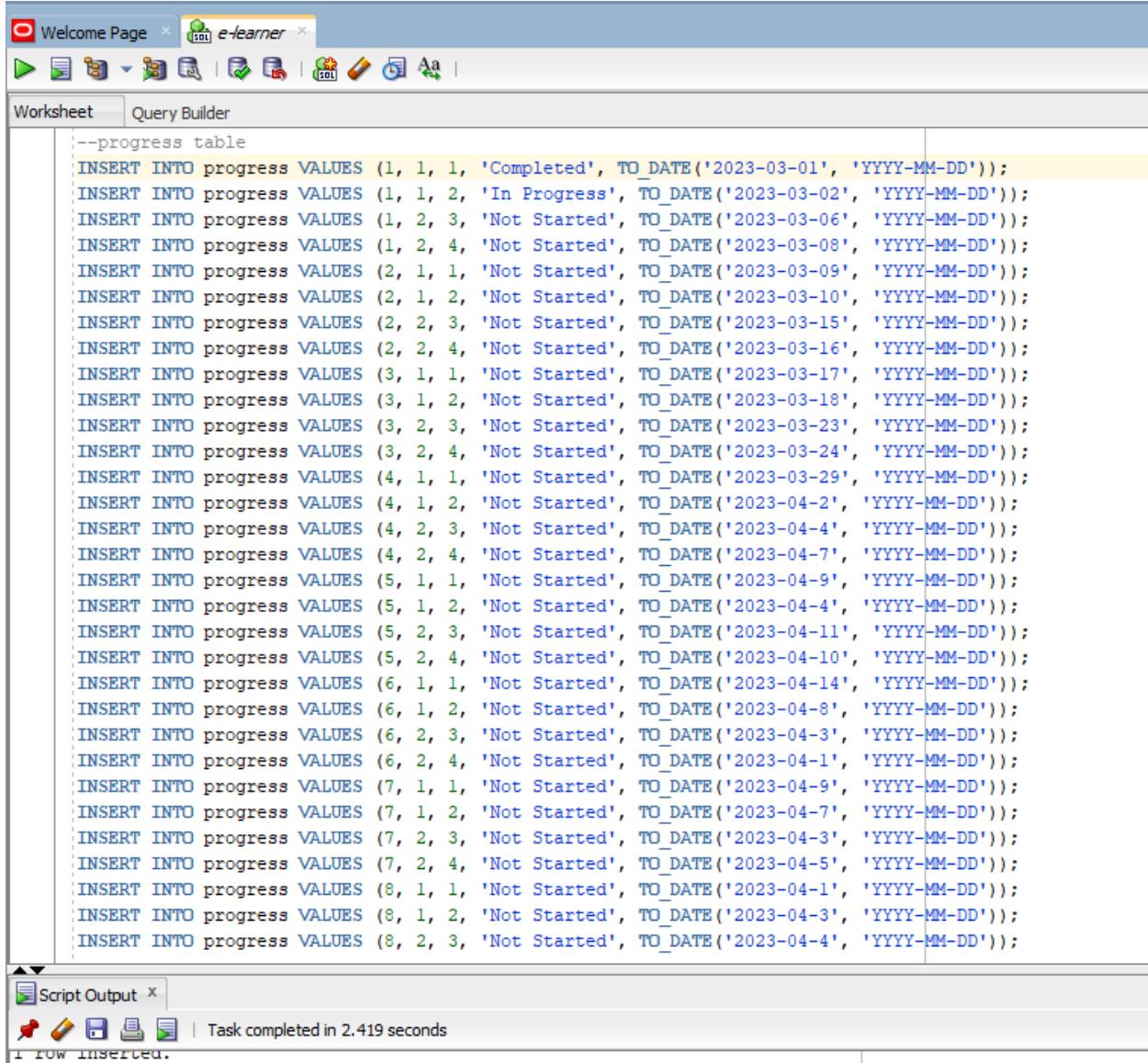


The screenshot shows the MySQL Workbench interface with the 'Query Builder' tab selected. The query editor contains the following SQL script:

```
--QA table
INSERT INTO qa VALUES (1, 'Java Basics', 'What are the basic concepts of Java programming?', 1, 1, 101, 'Great introduction to Java');
INSERT INTO qa VALUES (2, 'Object-Oriented Programming in Java', 'How does object-oriented programming work in Java?', 1, 1, 102, 'Clear explanation');
INSERT INTO qa VALUES (3, 'Algebra Fundamentals', 'What are the fundamental concepts of algebra?', 2, 2, 101, 'Great introduction to algebra');
INSERT INTO qa VALUES (4, 'Algebraic Equations', 'How to solve algebraic equations?', 2, 2, 102, 'Clear explanation');
INSERT INTO qa VALUES (5, 'Operating System Basics', 'What are the basic concepts of operating systems?', 3, 3, 101, 'Great introduction to OS');
INSERT INTO qa VALUES (6, 'Operating System Functions', 'What are the functions of an operating system?', 3, 3, 102, 'Clear explanation');
INSERT INTO qa VALUES (7, 'Arithmetic Sequences', 'How to find the nth term of an arithmetic sequence?', 4, 4, 101, 'Great explanation of arithmetic sequences');
INSERT INTO qa VALUES (8, 'Geometric Sequences', 'How to find the sum of a geometric sequence?', 4, 4, 102, 'Great explanation of geometric sequences');
INSERT INTO qa VALUES (9, 'Logic Basics', 'What are the basic principles of logic?', 5, 5, 101, 'Great introduction to logic');
INSERT INTO qa VALUES (10, 'Logical Reasoning', 'How to improve logical reasoning skills?', 5, 5, 102, 'Clear explanation of logical reasoning');
INSERT INTO qa VALUES (11, 'Python Basics', 'What are the basic concepts of Python programming?', 6, 6, 101, 'Great introduction to Python');
INSERT INTO qa VALUES (12, 'Python Data Structures', 'How to use data structures in Python?', 6, 6, 102, 'Clear explanation of Python data structures');
INSERT INTO qa VALUES (13, 'Data Science Fundamentals', 'What are the fundamental concepts of data science?', 7, 7, 101, 'Great introduction to data science');
INSERT INTO qa VALUES (14, 'Data Science Tools', 'Which tools are commonly used in data science?', 7, 7, 102, 'Clear explanation of data science tools');
INSERT INTO qa VALUES (15, 'AI Basics', 'What are the basic concepts of artificial intelligence?', 8, 8, 101, 'Great introduction to AI');
INSERT INTO qa VALUES (16, 'AI Applications', 'How is AI used in real-world applications?', 8, 8, 102, 'Clear explanation of AI applications');
INSERT INTO qa VALUES (17, 'Processor Architecture', 'What are the key components of processor architecture?', 9, 9, 101, 'Great overview of processor architecture');
INSERT INTO qa VALUES (18, 'Processor Design', 'How are processors designed?', 9, 9, 102, 'Clear explanation of processor design');
INSERT INTO qa VALUES (19, 'Linux Basics', 'What are the basic concepts of Linux?', 10, 10, 101, 'Great introduction to Linux');
INSERT INTO qa VALUES (20, 'Linux Commands', 'How to use common Linux commands?', 10, 10, 102, 'Clear explanation of Linux commands');
INSERT INTO qa VALUES (21, 'Database Fundamentals', 'What are the fundamental concepts of databases?', 11, 11, 101, 'Great introduction to databases');
INSERT INTO qa VALUES (22, 'Database Management', 'How to manage databases effectively?', 11, 11, 102, 'Clear explanation of database management');
INSERT INTO qa VALUES (23, 'Web Development Basics', 'What are the basic concepts of web development?', 12, 12, 101, 'Great introduction to web development');
INSERT INTO qa VALUES (24, 'Web Development Tools', 'Which tools are commonly used in web development?', 12, 12, 102, 'Clear explanation of web development tools');
INSERT INTO qa VALUES (25, 'Mobile Development Fundamentals', 'What are the fundamental concepts of mobile development?', 13, 13, 101, 'Great introduction to mobile development');
INSERT INTO qa VALUES (26, 'Mobile App Design', 'How to design mobile apps?', 13, 13, 102, 'Clear explanation of mobile app design');
INSERT INTO qa VALUES (27, 'Desktop Development Basics', 'What are the basic concepts of desktop development?', 14, 14, 101, 'Great introduction to desktop development');
INSERT INTO qa VALUES (28, 'Desktop Application Design', 'How to design desktop applications?', 14, 14, 102, 'Clear explanation of desktop application design');
INSERT INTO qa VALUES (29, 'Game Development Fundamentals', 'What are the fundamental concepts of game development?', 15, 15, 101, 'Great introduction to game development');
INSERT INTO qa VALUES (30, 'Game Design Principles', 'What are the principles of game design?', 15, 15, 102, 'Clear explanation of game design principles');
```

The 'Script Output' pane at the bottom shows the message: 'Task completed in 2.419 seconds'. A status bar at the bottom left indicates '1 row inserted.'

Figure 23. QA table insertion



```
--progress table
INSERT INTO progress VALUES (1, 1, 1, 'Completed', TO_DATE('2023-03-01', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (1, 1, 2, 'In Progress', TO_DATE('2023-03-02', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (1, 2, 3, 'Not Started', TO_DATE('2023-03-06', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (1, 2, 4, 'Not Started', TO_DATE('2023-03-08', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (2, 1, 1, 'Not Started', TO_DATE('2023-03-09', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (2, 1, 2, 'Not Started', TO_DATE('2023-03-10', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (2, 2, 3, 'Not Started', TO_DATE('2023-03-15', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (2, 2, 4, 'Not Started', TO_DATE('2023-03-16', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (3, 1, 1, 'Not Started', TO_DATE('2023-03-17', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (3, 1, 2, 'Not Started', TO_DATE('2023-03-18', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (3, 2, 3, 'Not Started', TO_DATE('2023-03-23', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (3, 2, 4, 'Not Started', TO_DATE('2023-03-24', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (4, 1, 1, 'Not Started', TO_DATE('2023-03-29', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (4, 1, 2, 'Not Started', TO_DATE('2023-04-2', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (4, 2, 3, 'Not Started', TO_DATE('2023-04-4', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (4, 2, 4, 'Not Started', TO_DATE('2023-04-7', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (5, 1, 1, 'Not Started', TO_DATE('2023-04-9', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (5, 1, 2, 'Not Started', TO_DATE('2023-04-4', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (5, 2, 3, 'Not Started', TO_DATE('2023-04-11', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (5, 2, 4, 'Not Started', TO_DATE('2023-04-10', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (6, 1, 1, 'Not Started', TO_DATE('2023-04-14', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (6, 1, 2, 'Not Started', TO_DATE('2023-04-8', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (6, 2, 3, 'Not Started', TO_DATE('2023-04-3', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (6, 2, 4, 'Not Started', TO_DATE('2023-04-1', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (7, 1, 1, 'Not Started', TO_DATE('2023-04-9', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (7, 1, 2, 'Not Started', TO_DATE('2023-04-7', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (7, 2, 3, 'Not Started', TO_DATE('2023-04-3', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (7, 2, 4, 'Not Started', TO_DATE('2023-04-5', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (8, 1, 1, 'Not Started', TO_DATE('2023-04-1', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (8, 1, 2, 'Not Started', TO_DATE('2023-04-3', 'YYYY-MM-DD'));
INSERT INTO progress VALUES (8, 2, 3, 'Not Started', TO_DATE('2023-04-4', 'YYYY-MM-DD'));
```

Script Output | Task completed in 2.419 seconds
1 row inserted.

Figure 24. Progress Table insertion

10 Select Statement

The screenshot shows a SQL worksheet interface with the following details:

- Tab bar: e-learner.sql, Welcome Page, e-learner
- Toolbar: SQL Worksheet, History, various icons for file operations.
- Worksheet tab: Worksheet (selected), Query Builder
- Query Editor: `select * from student;`
- Result tab: Query Result (selected)
- Result pane: Shows a table with 15 rows of student data. The columns are STUDENTID, STUDENTNAME, DOB, EMAIL, CONTACT, and COUNTRY.
- Message bar: All Rows Fetched: 15 in 1.023 seconds

STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY
1	Aadesh Shrestha	24-JUN-87	aadesh@example.com	1234567890	Argentina
2	Bhavesh Ronaldo	05-FEB-85	bhavesh@example.com	9876543210	Portugal
3	Aashraya Jr.	05-FEB-92	aashraya@example.com	5556667777	Brazil
4	Aayam Pokharel	09-SEP-85	aayam@example.com	1112223333	Croatia
5	Abhishek Yadav	20-DEC-98	abhishek@example.com	4445556666	France
6	Prabhesh Basnet	15-JUN-92	prabhesh@example.com	7778889999	Egypt
7	Divyanshu Mishra	30-MAR-86	divyanshu@example.com	2223334444	Spain
8	Pramithas Upreti	28-JUN-91	upreti@example.com	9998887777	Belgium
9	Suprabhat Rijal	08-JUL-91	suprabhat@example.com	6665554444	Netherlands
10	Manipal Kayastha	21-AUG-88	manipal@example.com	3334445555	Poland
11	Sarabh Subedi	10-APR-92	sarabh@example.com	2221110000	Senegal
12	Kaustuv Pandey	07-JAN-91	kaustuv@example.com	1110009999	Belgium
13	Kanak Thakur	15-MAR-93	kanak@example.com	9990001111	France
14	Suzan Khanal	28-JUL-93	suzan@example.com	8887776666	England
15	Rahul Sharma	21-MAR-91	rahul@example.com	7776665555	France

Figure 25. Student data

The screenshot shows a database management interface with a toolbar at the top, followed by tabs for 'SQL Worksheet' and 'History'. The 'SQL Worksheet' tab is active, displaying the query: 'select * from course;'. Below the query is a 'Query Result' window showing a table with 15 rows of course data. The table has columns: COURSEID, COURSENNAME, START_DATE, and END_DATE. The data is as follows:

COURSEID	COURSENNAME	START_DATE	END_DATE
1	1 Java	01-JAN-22	01-MAR-22
2	2 Algebra	01-FEB-22	01-APR-22
3	3 Operating System	01-MAR-22	01-MAY-22
4	4 Sequence	01-APR-22	01-JUN-22
5	5 Logic	01-MAY-22	01-JUL-22
6	6 Python	01-JUN-22	01-AUG-22
7	7 Data Science	01-JUL-22	01-SEP-22
8	8 AI	01-AUG-22	01-OCT-22
9	9 Processors	01-SEP-22	01-NOV-22
10	10 Linux	01-OCT-22	01-DEC-22
11	11 Database	01-NOV-22	01-JAN-23
12	12 Web Development	01-DEC-22	01-FEB-23
13	13 Mobile Development	01-JAN-23	01-MAR-23
14	14 Desktop Development	01-FEB-23	01-APR-23
15	15 Game Development	01-MAR-23	01-MAY-23

Figure 26. Course Data

The screenshot shows a SQL worksheet interface with the following details:

- SQL Worksheet History:** A tab bar at the top with "e-learner.sql", "Welcome Page", and "e-learner".
- Worksheet:** The active tab, showing the query `select * from enrollment;`
- Query Result:** A table displaying the results of the query.
- Table Headers:** ENROL_DATE, STUDENTID, COURSEID
- Data Rows:** 31 rows of enrollment data, each containing a row number (1-31), an enrollment date, a student ID, and a course ID.

	ENROL_DATE	STUDENTID	COURSEID
1	01-JAN-22	1	1
2	01-JAN-22	1	2
3	01-FEB-22	2	1
4	01-FEB-22	2	2
5	01-MAR-22	3	1
6	01-MAR-22	3	2
7	01-APR-22	4	1
8	01-APR-22	4	2
9	01-MAY-22	5	1
10	01-MAY-22	5	2
11	01-JUN-22	6	1
12	01-JUN-22	6	2
13	01-JUL-22	7	1
14	01-JUL-22	7	2
15	01-AUG-22	8	1
16	01-AUG-22	8	2
17	01-MAY-22	5	3
18	01-MAY-22	5	5
19	01-JUN-22	6	3
20	01-JUN-22	6	5
21	01-JUL-22	7	7
22	01-JUL-22	7	3
23	01-AUG-22	8	9
24	01-AUG-22	8	5
25	01-SEP-22	9	1
26	01-SEP-22	9	2
27	01-OCT-22	10	1
28	01-OCT-22	10	2
29	01-NOV-22	11	1
30	01-NOV-22	11	2
31	01-DEC-22	14	1

Figure 27. Enrollment Data

The screenshot shows the SQL Worksheet interface in SQL Server Management Studio. A query is run against the 'e-learner' database to select all columns from the 'instructor' table. The results are displayed in a grid format with 15 rows, each representing an instructor with their ID, name, contact number, and email address.

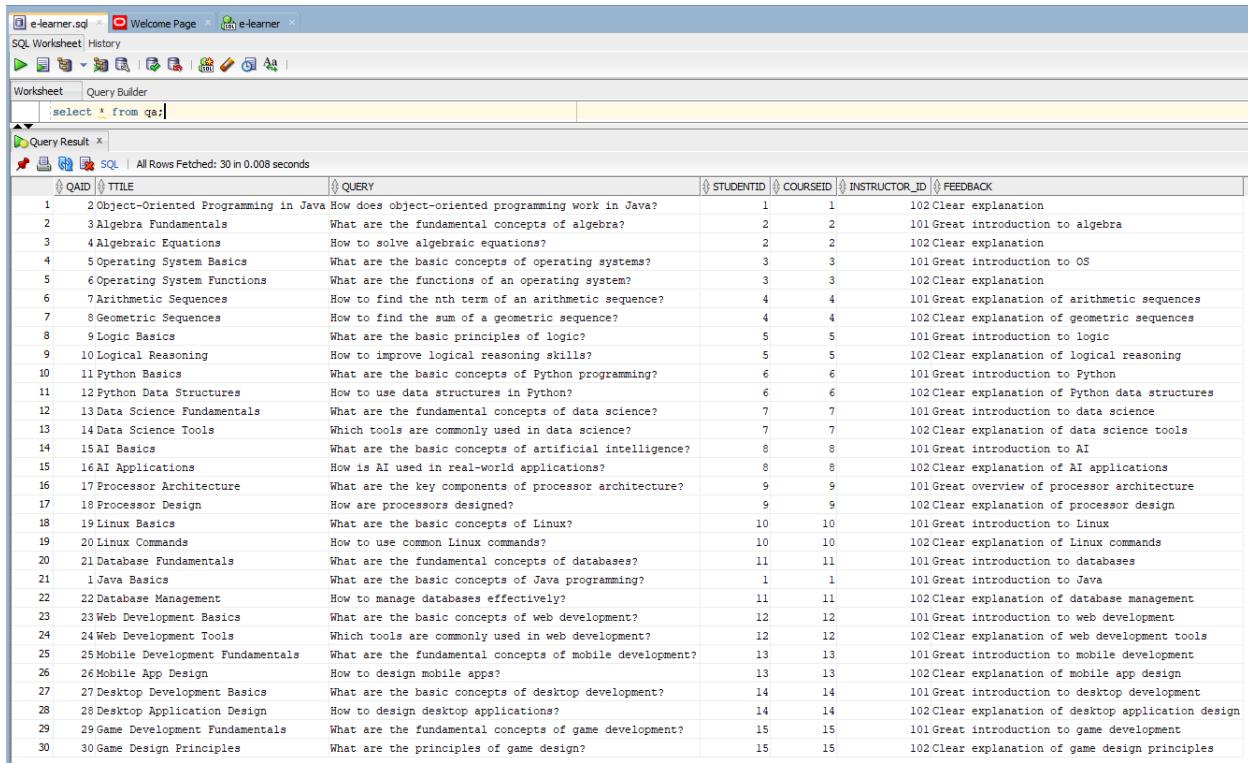
	INSTRUCTORID	INSTRUCTORMNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
1	101	Madhav Kumar	1234567890	madhav@example.com
2	102	Purna Singh	2345678901	purna@example.com
3	103	Jyotita Magar	3456789012	jyotita@example.com
4	104	Kamala Harris	4567890123	kamala@example.com
5	105	Susma Adhikari	5678901234	susma@example.com
6	106	Anjali Adhikari	6789012345	anjali@example.com
7	107	Krishna Thaku	7890123456	krishna@example.com
8	108	Anju Shrestha	8901234567	anju@example.com
9	109	Man Bahadur	9012345678	man@example.com
10	110	Keshab Yadav	0123456789	keshab@example.com
11	111	Kailash Raut	1111111111	kailash@example.com
12	112	Tilak Raj	2222222222	tilak@example.com
13	113	Arjun Pandey	3333333333	arjun@example.com
14	114	Badal Neupane	4444444444	badal@example.com
15	115	Uzzal Phuyal	5555555555	uzzal@example.com

Figure 28. Insturctor Data

The screenshot shows a SQL development environment with three tabs at the top: 'e-learner.sql' (selected), 'Welcome Page', and 'e-learner'. The 'SQL Worksheet' tab is active, displaying the query: 'select * from c_instructor;'. Below the query is a table titled 'Query Result' showing the results of the execution. The table has two columns: 'COURSEID' and 'INSTRUCTORID'. The data consists of 30 rows, each containing a value for 'COURSEID' and 'INSTRUCTORID'. The 'INSTRUCTORID' values are all 101 except for the last row which is 115.

	COURSEID	INSTRUCTORID
1	1	101
2	2	101
3	3	101
4	8	101
5	11	101
6	2	102
7	9	102
8	3	103
9	13	103
10	4	104
11	12	104
12	5	105
13	4	106
14	6	106
15	15	106
16	1	107
17	6	107
18	7	107
19	14	107
20	5	108
21	8	108
22	7	109
23	9	109
24	10	110
25	10	111
26	11	111
27	12	112
28	13	113
29	14	114
30	15	115

Figure 29. Course_Instructor Data



The screenshot shows a SQL worksheet interface with a toolbar at the top and a main area divided into two panes. The left pane is titled 'Worksheet' and contains a query editor with the SQL command: 'select * from qa;'. The right pane is titled 'Query Result' and displays a table with 30 rows of data. The table has columns: QID, TITLE, QUERY, STUDENTID, COURSEID, INSTRUCTOR_ID, and FEEDBACK.

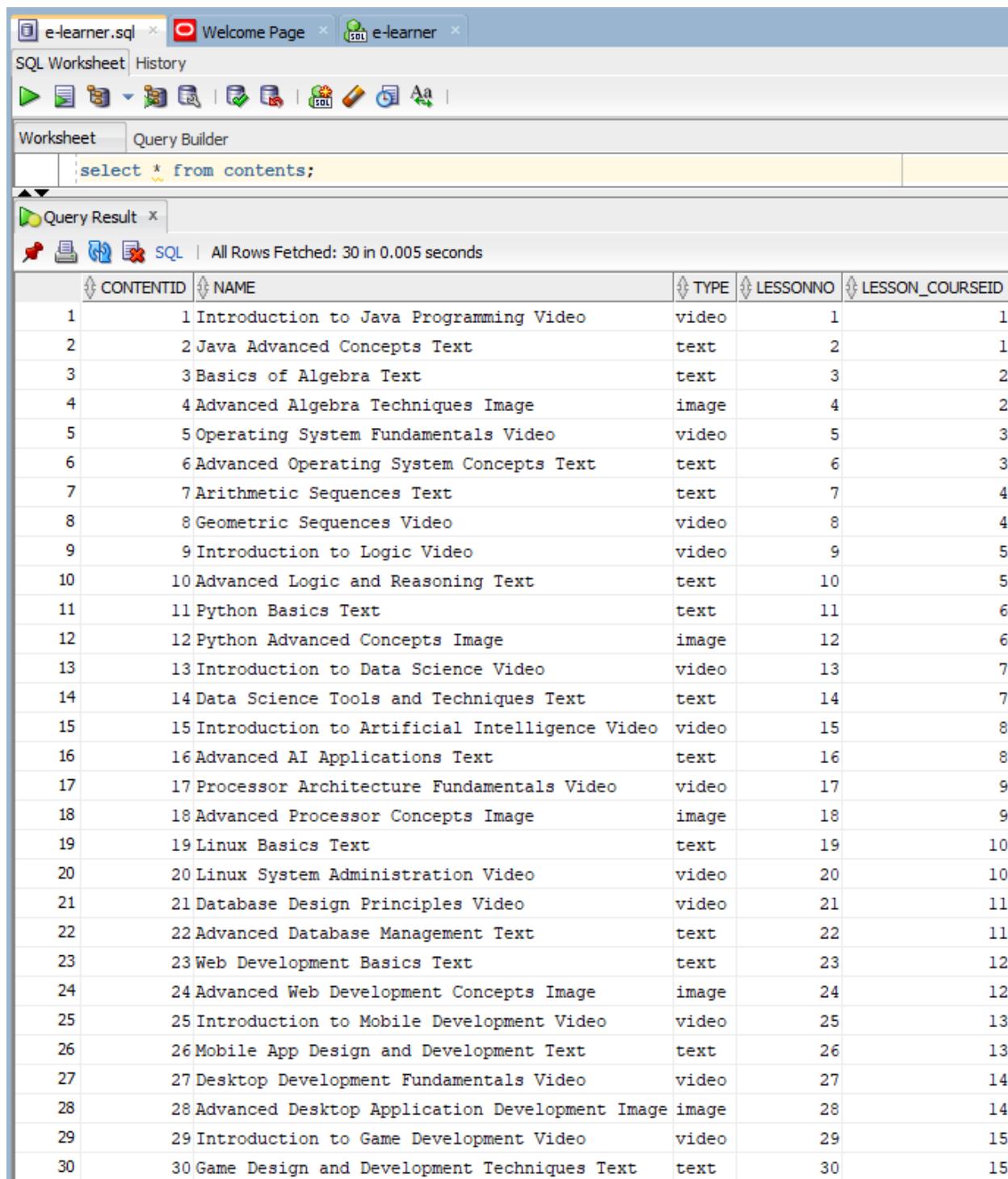
QID	TITLE	QUERY	STUDENTID	COURSEID	INSTRUCTOR_ID	FEEDBACK
1	2 Object-Oriented Programming in Java	How does object-oriented programming work in Java?	1	1	102	Clear explanation
2	3 Algebra Fundamentals	What are the fundamental concepts of algebra?	2	2	101	Great introduction to algebra
3	4 Algebraic Equations	How to solve algebraic equations?	2	2	102	Clear explanation
4	5 Operating System Basics	What are the basic concepts of operating systems?	3	3	101	Great introduction to OS
5	6 Operating System Functions	What are the functions of an operating system?	3	3	102	Clear explanation
6	7 Arithmetic Sequences	How to find the nth term of an arithmetic sequence?	4	4	101	Great explanation of arithmetic sequences
7	8 Geometric Sequences	How to find the sum of a geometric sequence?	4	4	102	Clear explanation of geometric sequences
8	9 Logic Basics	What are the basic principles of logic?	5	5	101	Great introduction to logic
9	10 Logical Reasoning	How to improve logical reasoning skills?	5	5	102	Clear explanation of logical reasoning
10	11 Python Basics	What are the basic concepts of Python programming?	6	6	101	Great introduction to Python
11	12 Python Data Structures	How to use data structures in Python?	6	6	102	Clear explanation of Python data structures
12	13 Data Science Fundamentals	What are the fundamental concepts of data science?	7	7	101	Great introduction to data science
13	14 Data Science Tools	Which tools are commonly used in data science?	7	7	102	Clear explanation of data science tools
14	15 AI Basics	What are the basic concepts of artificial intelligence?	8	8	101	Great introduction to AI
15	16 AI Applications	How is AI used in real-world applications?	8	8	102	Clear explanation of AI applications
16	17 Processor Architecture	What are the key components of processor architecture?	9	9	101	Great overview of processor architecture
17	18 Processor Design	How are processors designed?	9	9	102	Clear explanation of processor design
18	19 Linux Basics	What are the basic concepts of Linux?	10	10	101	Great introduction to Linux
19	20 Linux Commands	How to use common Linux commands?	10	10	102	Clear explanation of Linux commands
20	21 Database Fundamentals	What are the fundamental concepts of databases?	11	11	101	Great introduction to databases
21	22 Java Basics	What are the basic concepts of Java programming?	1	1	101	Great introduction to Java
22	23 Database Management	How to manage databases effectively?	11	11	102	Clear explanation of database management
23	24 Web Development Basics	What are the basic concepts of web development?	12	12	101	Great introduction to web development
24	25 Web Development Tools	Which tools are commonly used in web development?	12	12	102	Clear explanation of web development tools
25	26 Mobile Development Fundamentals	What are the fundamental concepts of mobile development?	13	13	101	Great introduction to mobile development
26	27 Mobile App Design	How to design mobile apps?	13	13	102	Clear explanation of mobile app design
27	28 Desktop Development Basics	What are the basic concepts of desktop development?	14	14	101	Great introduction to desktop development
28	29 Desktop Application Design	How to design desktop applications?	14	14	102	Clear explanation of desktop application design
29	30 Game Development Fundamentals	What are the fundamental concepts of game development?	15	15	101	Great introduction to game development
30	Game Design Principles	What are the principles of game design?	15	15	102	Clear explanation of game design principles

Figure 30. QA data

The screenshot shows the e-learner software interface. At the top, there are tabs for 'Welcome Page' and 'e-learner'. Below the tabs is a toolbar with various icons. The main area has two tabs: 'Worksheet' and 'Query Builder', with 'Worksheet' selected. A SQL query is entered in the worksheet: 'select * from lesson;'. Below the query is a table titled 'Script Output' with a single row showing 'Query Result'. The table displays the results of the query, which is a list of 30 lessons. The columns are 'LESSONNO', 'TITLE', and 'COURSEID'. The data is as follows:

LESSONNO	TITLE	COURSEID
1	1 Introduction to Java Programming	1
2	2 Java Advanced Concepts	1
3	3 Basics of Algebra	2
4	4 Advanced Algebra Techniques	2
5	5 Operating System Fundamentals	3
6	6 Advanced Operating System Concepts	3
7	7 Arithmetic Sequences	4
8	8 Geometric Sequences	4
9	9 Introduction to Logic	5
10	10 Advanced Logic and Reasoning	5
11	11 Python Basics	6
12	12 Python Advanced Concepts	6
13	13 Introduction to Data Science	7
14	14 Data Science Tools and Techniques	7
15	15 Introduction to Artificial Intelligence	8
16	16 Advanced AI Applications	8
17	17 Processor Architecture Fundamentals	9
18	18 Advanced Processor Concepts	9
19	19 Linux Basics	10
20	20 Linux System Administration	10
21	21 Database Design Principles	11
22	22 Advanced Database Management	11
23	23 Web Development Basics	12
24	24 Advanced Web Development Concepts	12
25	25 Introduction to Mobile Development	13
26	26 Mobile App Design and Development	13
27	27 Desktop Development Fundamentals	14
28	28 Advanced Desktop Application Development	14
29	29 Introduction to Game Development	15
30	30 Game Design and Development Techniques	15

Figure 31. Lesson data



The screenshot shows a SQL worksheet interface with three tabs at the top: 'e-learner.sql' (selected), 'Welcome Page', and 'e-learner'. Below the tabs is a toolbar with various icons. The main area has two tabs: 'Worksheet' (selected) and 'Query Builder'. In the 'Worksheet' tab, the query `select * from contents;` is entered. The results are displayed in a table titled 'Query Result' with 30 rows. The table has columns: CONTENTID, NAME, TYPE, LESSONNO, and LESSON_COURSEID. The data includes various types of educational content like videos, texts, and images across different lesson numbers and course IDs.

CONTENTID	NAME	TYPE	LESSONNO	LESSON_COURSEID
1	1 Introduction to Java Programming Video	video	1	1
2	2 Java Advanced Concepts Text	text	2	1
3	3 Basics of Algebra Text	text	3	2
4	4 Advanced Algebra Techniques Image	image	4	2
5	5 Operating System Fundamentals Video	video	5	3
6	6 Advanced Operating System Concepts Text	text	6	3
7	7 Arithmetic Sequences Text	text	7	4
8	8 Geometric Sequences Video	video	8	4
9	9 Introduction to Logic Video	video	9	5
10	10 Advanced Logic and Reasoning Text	text	10	5
11	11 Python Basics Text	text	11	6
12	12 Python Advanced Concepts Image	image	12	6
13	13 Introduction to Data Science Video	video	13	7
14	14 Data Science Tools and Techniques Text	text	14	7
15	15 Introduction to Artificial Intelligence Video	video	15	8
16	16 Advanced AI Applications Text	text	16	8
17	17 Processor Architecture Fundamentals Video	video	17	9
18	18 Advanced Processor Concepts Image	image	18	9
19	19 Linux Basics Text	text	19	10
20	20 Linux System Administration Video	video	20	10
21	21 Database Design Principles Video	video	21	11
22	22 Advanced Database Management Text	text	22	11
23	23 Web Development Basics Text	text	23	12
24	24 Advanced Web Development Concepts Image	image	24	12
25	25 Introduction to Mobile Development Video	video	25	13
26	26 Mobile App Design and Development Text	text	26	13
27	27 Desktop Development Fundamentals Video	video	27	14
28	28 Advanced Desktop Application Development Image	image	28	14
29	29 Introduction to Game Development Video	video	29	15
30	30 Game Design and Development Techniques Text	text	30	15

Figure 32. Content Data

The screenshot shows a SQL worksheet interface with the following details:

- SQL Worksheet History:** A tab bar at the top includes "e-learner.sql", "Welcome Page", and "e-learner".
- Worksheet:** The main area displays the query: `select * from progress;`
- Query Result:** A table titled "Query Result" shows the results of the query. The table has the following columns:
 - STUDENTID
 - COURSEID
 - LESSONNO
 - STATUS
 - LAST_ACCESSED_DATE
- Data:** The table contains 32 rows of data, each representing a student's progress. The data is as follows:

STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSED_DATE
1	1	1	1 Completed	01-MAR-23
2	1	1	2 In Progress	02-MAR-23
3	1	2	3 Not Started	07-MAR-23
4	1	2	4 Not Started	08-MAR-23
5	2	1	1 Not Started	09-MAR-23
6	2	1	2 Not Started	10-MAR-23
7	2	2	3 Not Started	15-MAR-23
8	2	2	4 Not Started	16-MAR-23
9	3	1	1 Not Started	17-MAR-23
10	3	1	2 Not Started	18-MAR-23
11	3	2	3 Not Started	23-MAR-23
12	3	2	4 Not Started	24-MAR-23
13	4	1	1 Not Started	29-MAR-23
14	4	1	2 Not Started	02-APR-23
15	4	2	3 Not Started	04-APR-23
16	4	2	4 Not Started	07-APR-23
17	5	1	1 Not Started	09-APR-23
18	5	1	2 Not Started	04-APR-23
19	5	2	3 Not Started	11-APR-23
20	5	2	4 Not Started	10-APR-23
21	6	1	1 Not Started	14-APR-23
22	6	1	2 Not Started	08-APR-23
23	6	2	3 Not Started	03-APR-23
24	6	2	4 Not Started	01-APR-23
25	7	1	1 Not Started	09-APR-23
26	7	1	2 Not Started	07-APR-23
27	7	2	3 Not Started	03-APR-23
28	7	2	4 Not Started	05-APR-23
29	8	1	1 Not Started	01-APR-23
30	8	1	2 Not Started	03-APR-23
31	8	2	3 Not Started	04-APR-23
32	9	1	1 Not Started	01-APR-23

Figure 33. Progress Data

11 Forms

11.1 Dashboard

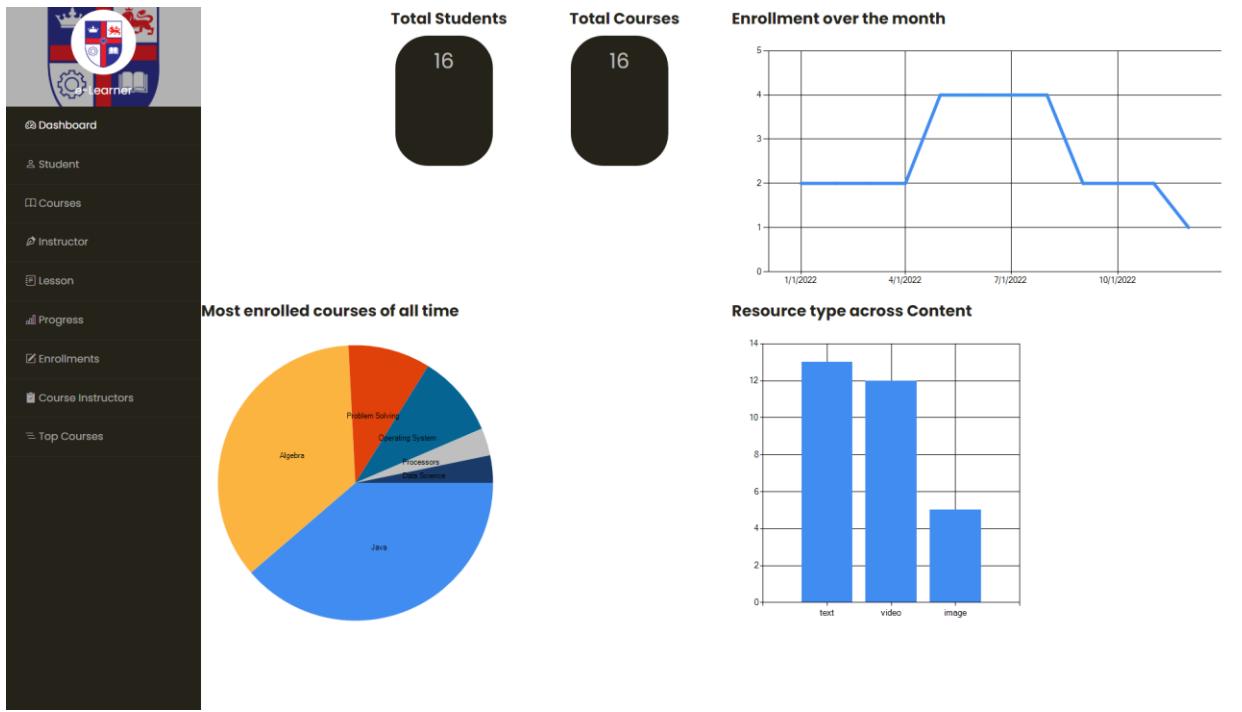


Figure 34. Home Page

11.2 Complex Form and Queries

11.2.1 SQL Queries

I. Student Enrollment

Student DropDown

```
SELECT "STUDENTNAME", "STUDENTID" FROM "STUDENT"
```

GridView

```
SELECT      e.ENROL_DATE,      s.STUDENTID,      s.STUDENTNAME,      c.COURSEID,
c.COURSENAME, c.START_DATE, c.END_DATE
FROM enrollment e
JOIN student s ON e.STUDENTID = s.STUDENTID
JOIN course c ON e.COURSEID = c.COURSEID
WHERE s.STUDENTID = :student
```

II. Course Instructors

Course DropDown

```
SELECT "COURSEID", "COURSENAME" FROM "COURSE"
```

GridView

```
SELECT      c.COURSEID,      c.COURSENAME,      c.START_DATE,      c.END_DATE,
i.INSTRUCTORID,          i.INSTRUCTORTNAME,          i.INSTRUCTORCONTACT,
i.INSTRUCTOREMAIL  FROM course c JOIN c_instructor ci ON c.COURSEID =
```

```
ci.COURSEID JOIN instructor i ON ci.INSTRUCTORID = i.INSTRUCTORID WHERE  
c.courseid = :course
```

III. Top Courses for given month

Month DropDown

```
SELECT DISTINCT  
    TO_CHAR(enrol_date, 'Mon') AS EnrollMonth  
FROM  
    ENROLLMENT  
ORDER BY EnrollMonth
```

Year DropDown

```
SELECT DISTINCT  
    EXTRACT(YEAR FROM Enrol_Date) AS EnrollYear  
FROM  
    ENROLLMENT  
ORDER BY  
    EnrollYear DESC
```

GridView

```
SELECT * FROM (  
    SELECT  
        Course.COURSEID,
```

```
Course.COURSENAME,  
COUNT(*) AS NumberOfStudents  
FROM  
ENROLLMENT  
INNER JOIN  
Course ON Enrollment.COURSEID = Course.COURSEID  
WHERE  
TO_CHAR(enrol_date, 'Mon') = :SelectedMonth  
AND  
EXTRACT(YEAR FROM ENROL_DATE) = :SelectedYear  
GROUP BY  
Course.COURSEID,  
Course.COURSENAME  
ORDER BY  
NumberOfStudents DESC  
)  
WHERE ROWNUM <= 3
```

11.2.2 Dashboard Queries

Line Chart

```
SELECT enrol_date, COUNT(*) AS enrollment_count  
FROM enrollment  
GROUP BY enrol_date  
ORDER BY enrol_date
```

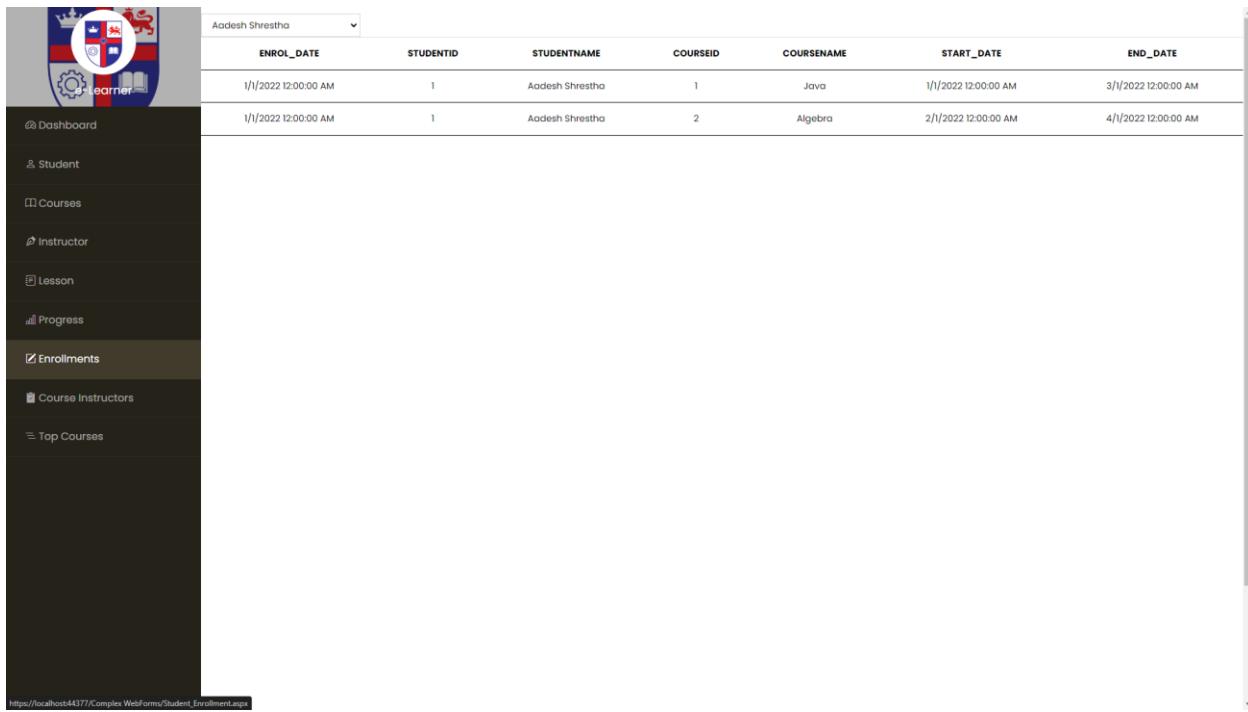
Pie Chart

```
SELECT C.COURSENAME, COUNT(E.COURSEID) AS ENROLLMENT_COUNT  
FROM ENROLLMENT E  
JOIN COURSE C ON E.COURSEID = C.COURSEID  
GROUP BY C.COURSENAME  
ORDER BY ENROLLMENT_COUNT DESC
```

Bar Graph

```
SELECT TYPE, COUNT(TYPE) AS TP  
FROM CONTENTS  
GROUP BY TYPE
```

11.2.3 Complex Forms

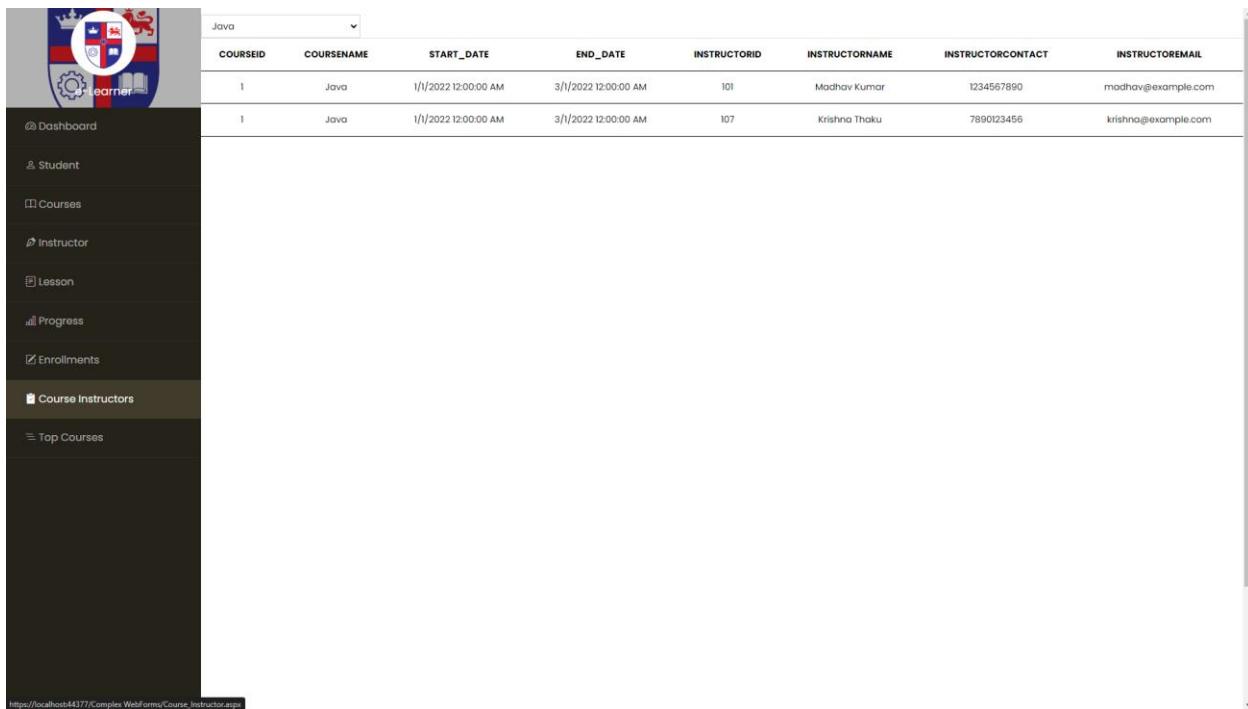


The screenshot shows a web application interface for managing student enrollments. On the left, a sidebar navigation menu includes options like Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments (which is selected), Course Instructors, and Top Courses. The main content area displays a dropdown menu for 'Aadesh Shrestha' and a table of enrollment records:

ENROL_DATE	STUDENTID	STUDENTNAME	COURSEID	COURSENAME	START_DATE	END_DATE
1/1/2022 12:00:00 AM	1	Aadesh Shrestha	1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM
1/1/2022 12:00:00 AM	1	Aadesh Shrestha	2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM

https://localhost:44377/Complex_WebForms/Student_Enrollment.aspx

Figure 35. Student Enrollment Page

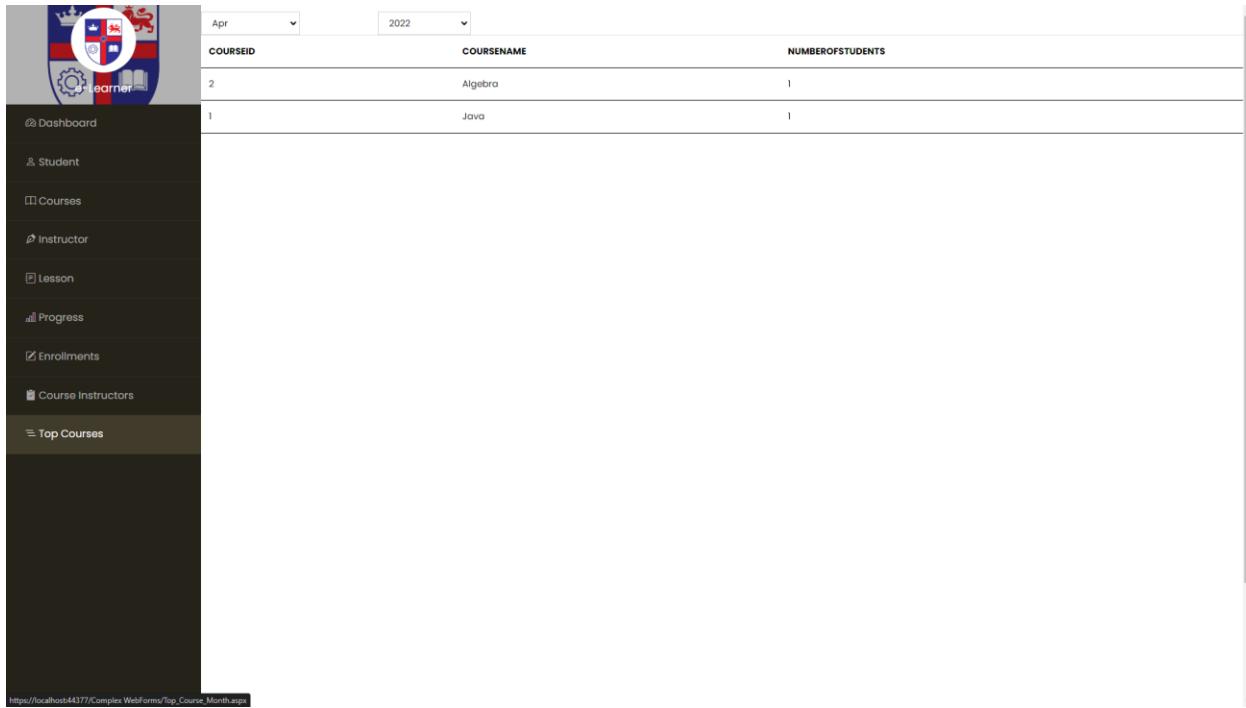


The screenshot shows a web application interface for managing course instructors. On the left, a sidebar navigation menu includes options like Dashboard, Student, Courses, Instructor (which is selected), Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main content area displays a dropdown menu for 'Java' and a table of instructor records:

COURSEID	COURSENAME	START_DATE	END_DATE	INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	101	Madhav Kumar	1234567890	madhav@example.com
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	107	Krishna Thaku	7890123456	krishna@example.com

https://localhost:44377/Complex_WebForms/Course_Instructor.aspx

Figure 36. Course Instructor Page

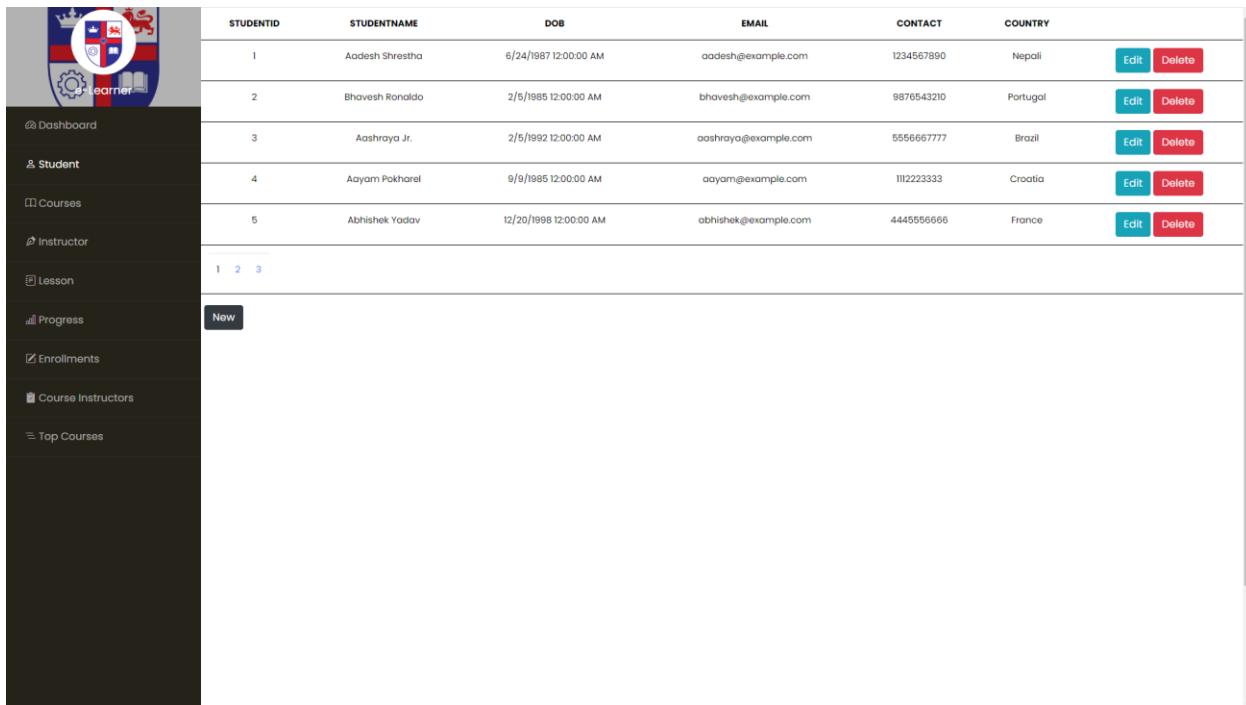


The screenshot shows a web application interface for managing courses. On the left is a dark sidebar menu with various navigation options. The main content area displays a table titled "Top Courses" with two rows. The columns are labeled "COURSEID", "COURSENAME", and "NUMBEROFSTUDENTS". The first row shows "Algebra" with "1" student, and the second row shows "Java" with "1" student. At the top of the main area, there are dropdown menus for "Month" (set to "Apr") and "Year" (set to "2022").

COURSEID	COURSENAME	NUMBEROFSTUDENTS
2	Algebra	1
1	Java	1

Figure 37. Top Course Enrolled Page

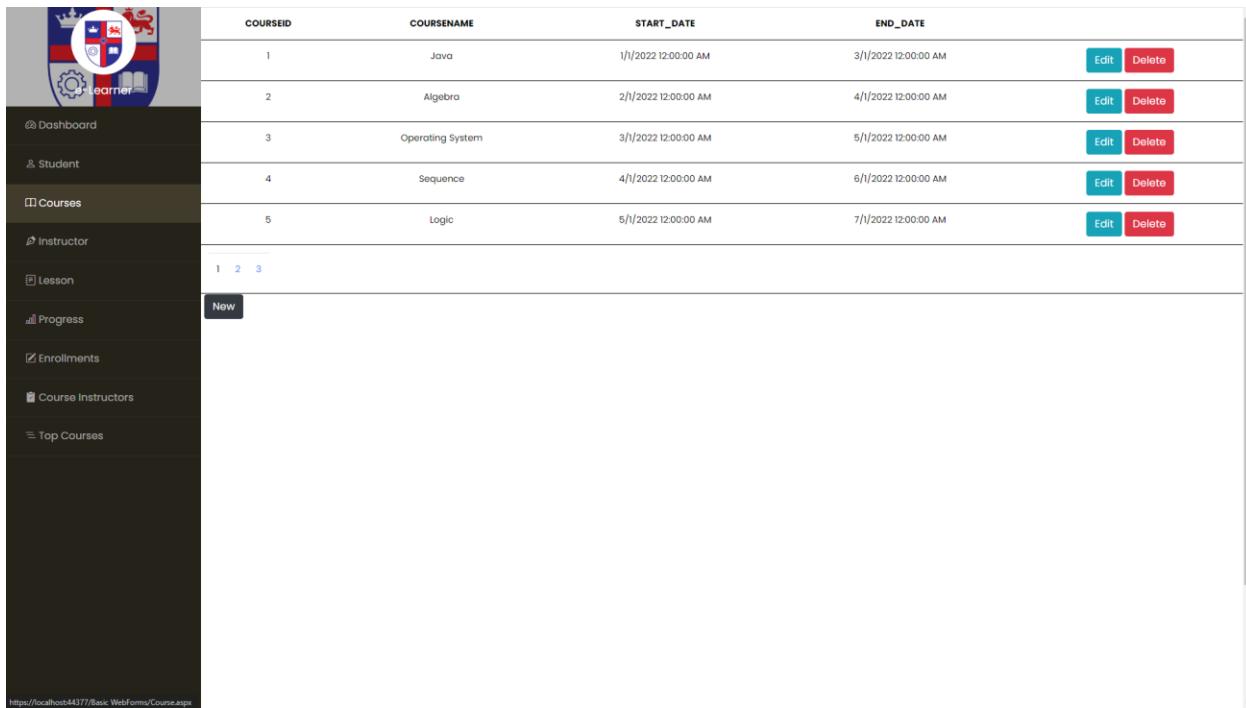
11.3 Simple Forms



The screenshot shows a web application interface for managing student records. On the left is a dark sidebar menu with various navigation options. The main content area displays a table titled "Students" with five rows of data. The columns are labeled "STUDENTID", "STUDENTNAME", "DOB", "EMAIL", "CONTACT", and "COUNTRY". Each row includes "Edit" and "Delete" buttons. At the bottom of the table, there are navigation links "1 2 3" and a "New" button.

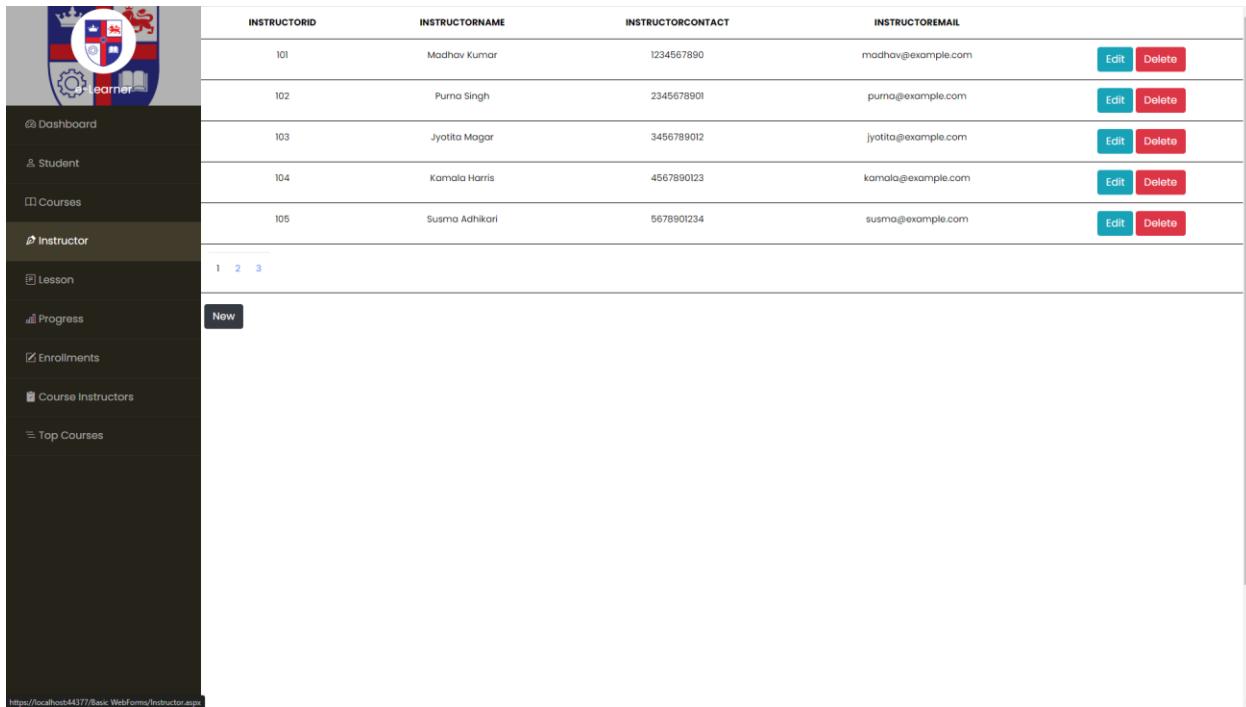
STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	Nepali
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France

Figure 38. Student Page



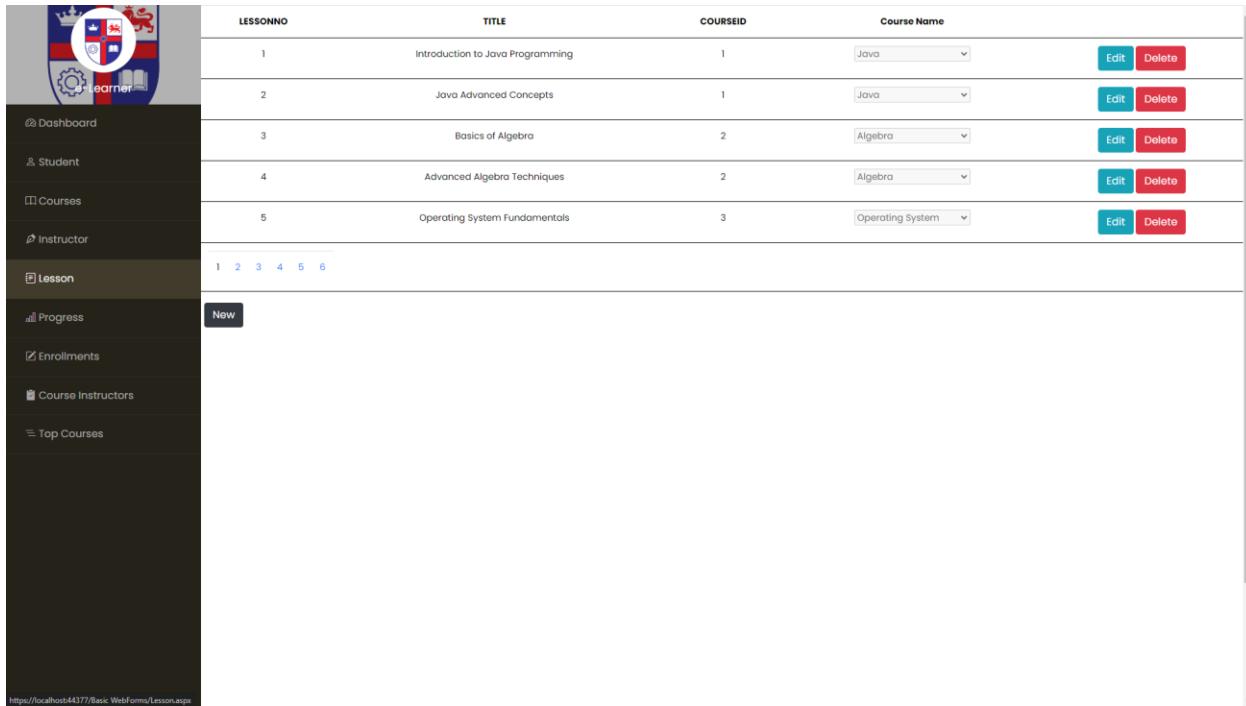
COURSEID	COURSENAME	START_DATE	END_DATE	
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
3	Operating System	3/1/2022 12:00:00 AM	5/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
4	Sequence	4/1/2022 12:00:00 AM	6/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
5	Logic	5/1/2022 12:00:00 AM	7/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>

Figure 39. Courses Page



INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL	
101	Madhav Kumar	1234567890	madhav@example.com	<button>Edit</button> <button>Delete</button>
102	Purna Singh	2345678901	purna@example.com	<button>Edit</button> <button>Delete</button>
103	Jyotita Magor	3456789012	jyotita@example.com	<button>Edit</button> <button>Delete</button>
104	Kamala Harris	4567890123	kamala@example.com	<button>Edit</button> <button>Delete</button>
105	Susma Adhikari	5678901234	susma@example.com	<button>Edit</button> <button>Delete</button>

Figure 40. Instructors Page

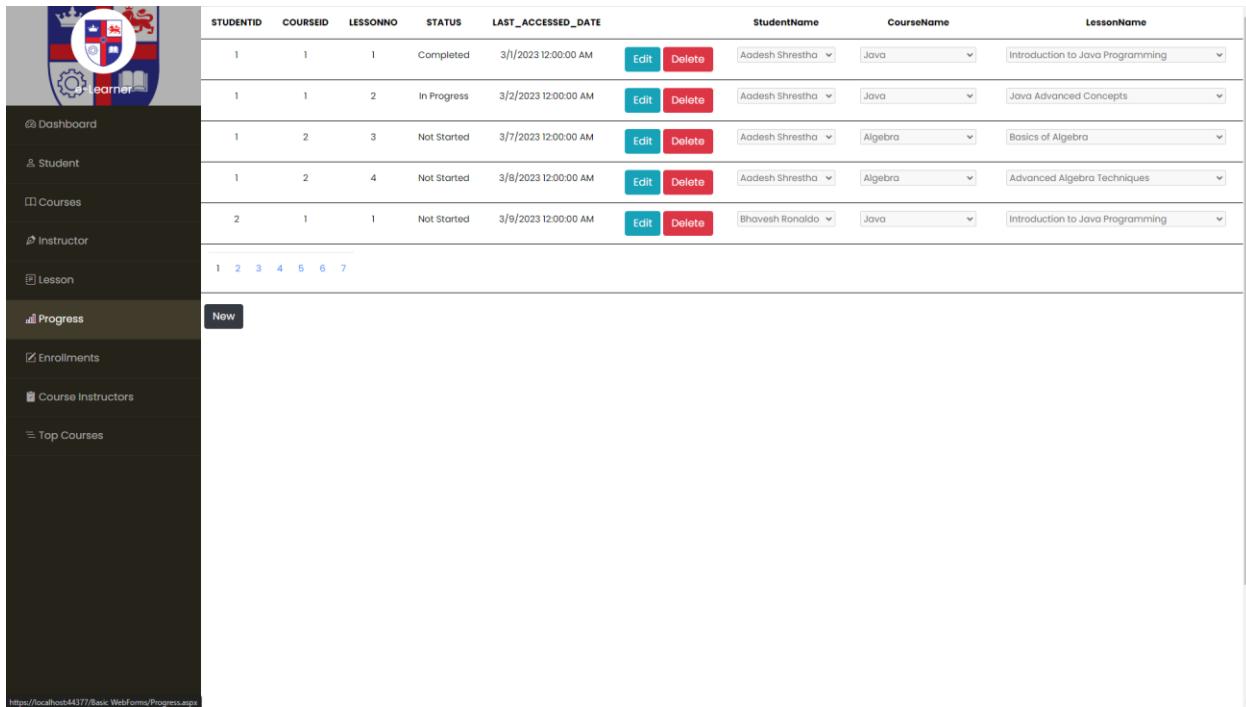


The screenshot shows a web-based application interface for managing lessons. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson (selected), Progress, Enrollments, Course Instructors, and Top Courses. Below the sidebar is a URL bar with the address <https://localhost:44377/basic.WebForms/Lesson.aspx>. The main content area displays a table titled 'Lesson' with the following data:

LESSONNO	TITLE	COURSEID	Course Name
1	Introduction to Java Programming	1	Java
2	Java Advanced Concepts	1	Java
3	Basics of Algebra	2	Algebra
4	Advanced Algebra Techniques	2	Algebra
5	Operating System Fundamentals	3	Operating System

Below the table are page navigation numbers 1, 2, 3, 4, 5, 6 and a 'New' button.

Figure 41. Lesson Page



The screenshot shows a web-based application interface for tracking student progress. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson (selected), Progress, Enrollments, Course Instructors, and Top Courses. Below the sidebar is a URL bar with the address <https://localhost:44377/basic.WebForms/Progress.aspx>. The main content area displays a table titled 'Progress' with the following data:

STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSED_DATE	StudentName	CourseName	LessonName
1	1	1	Completed	3/1/2023 12:00:00 AM	Edit Delete	Aadesh Shrestha	Java
1	1	2	In Progress	3/2/2023 12:00:00 AM	Edit Delete	Aadesh Shrestha	Java Advanced Concepts
1	2	3	Not Started	3/7/2023 12:00:00 AM	Edit Delete	Aadesh Shrestha	Basics of Algebra
1	2	4	Not Started	3/8/2023 12:00:00 AM	Edit Delete	Aadesh Shrestha	Advanced Algebra Techniques
2	1	1	Not Started	3/9/2023 12:00:00 AM	Edit Delete	Bhavesh Ronaldo	Introduction to Java Programming

Below the table are page navigation numbers 1, 2, 3, 4, 5, 6, 7 and a 'New' button.

Figure 42. Progress Page

12 User Manual

Introduction

e-learner is an online learning platform built for the sole purpose of providing online learning courses and resources to a wide range of students. Any student can enrol in multiple courses, taught by world class instructors. The students are provided with wide range of resources/content for better understanding of every details. The enrolled students have the opportunity to learn from the best and also track their progress on enrolled courses.

Getting Started

To start learning from our platform, any user can run the program using any edition of Visual Studio 2022. Open the Solution file and directly run the folder from the Interface provided by Visual Studio 2022.

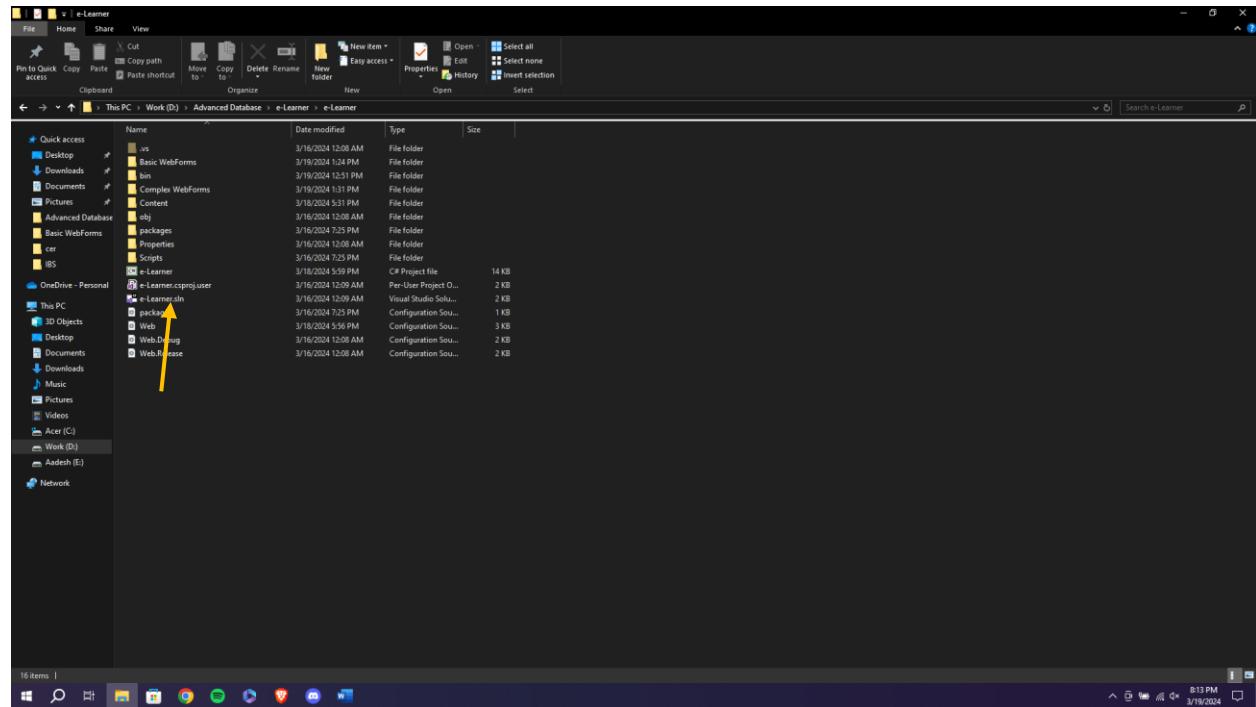


Figure 43. Folder with .sln file

After running the sln file, directly press the IIS Express Button on top of Visual Studio 2022

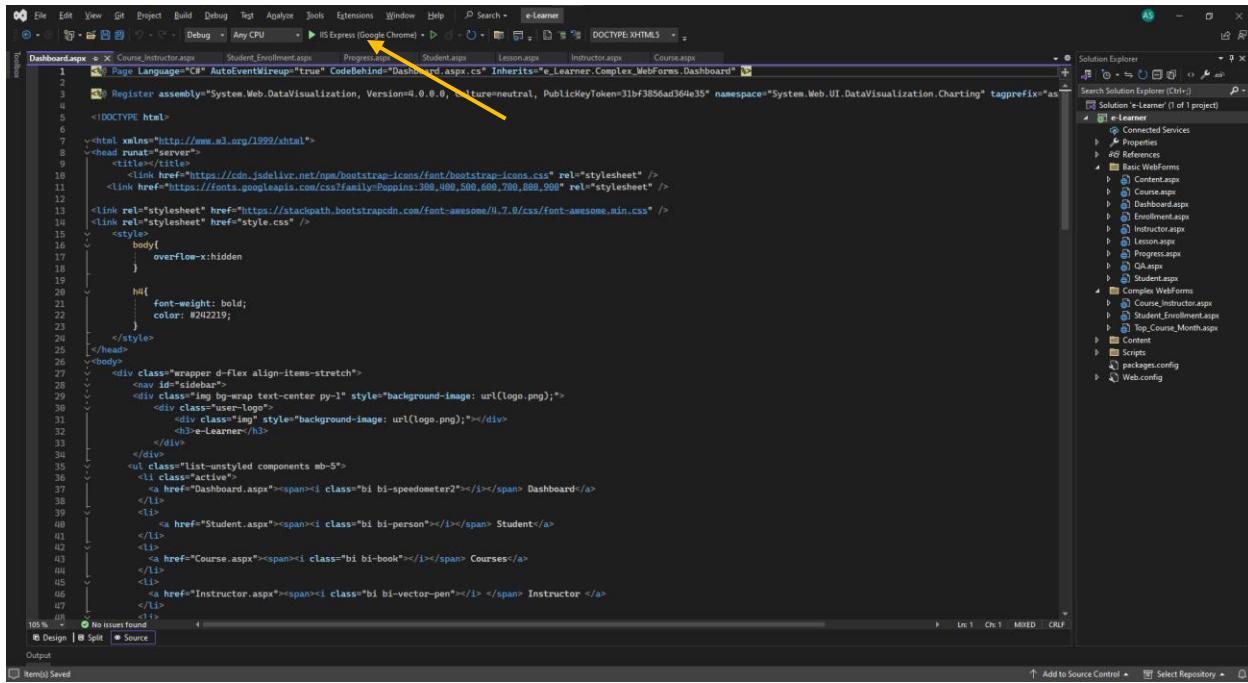


Figure 44. Website Server Launcher

The button triggers localhost, and the website is up and running on you google chrome web browser

UI Overview

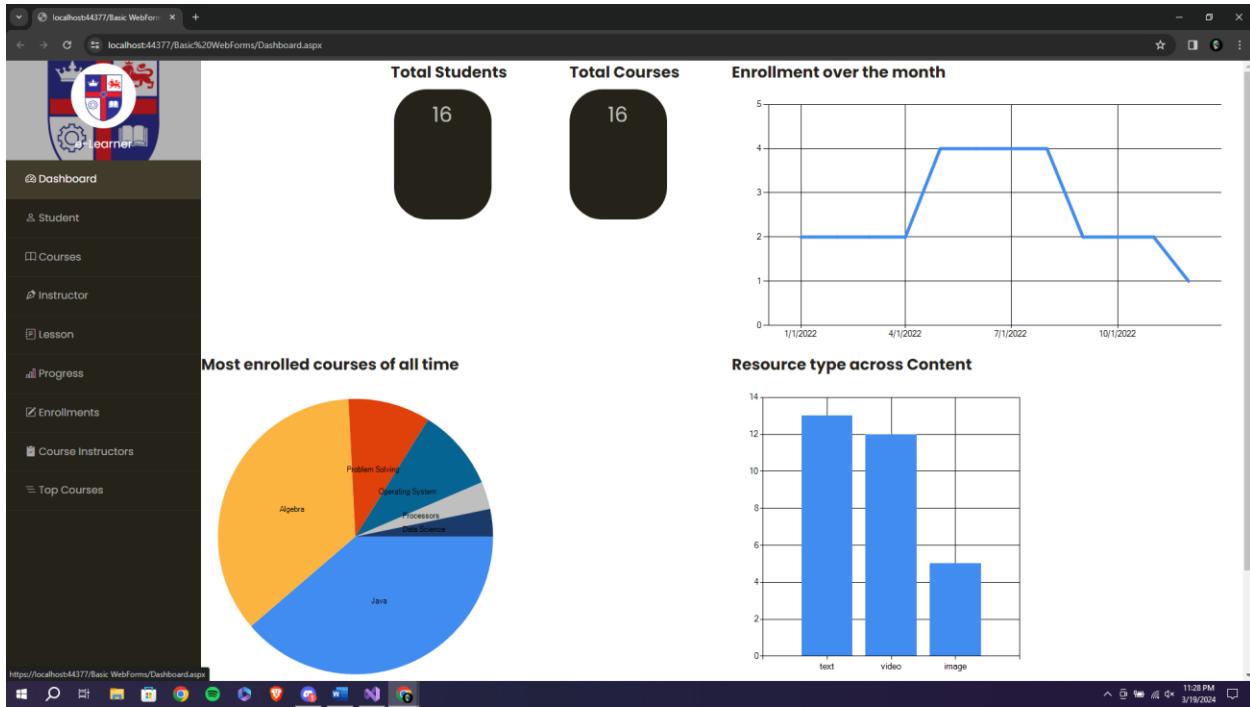


Figure 45. Home Page

The first page displayed is the main page/dashboard with some meaningful insights about e-Learners.

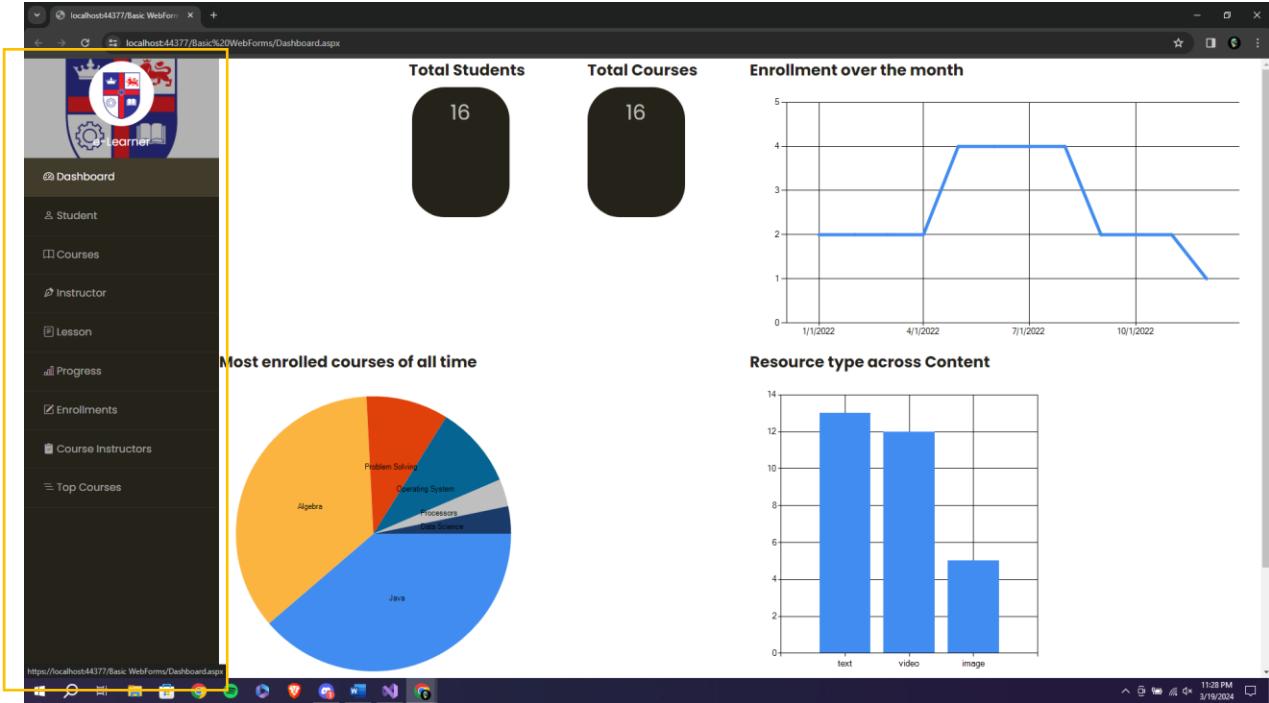
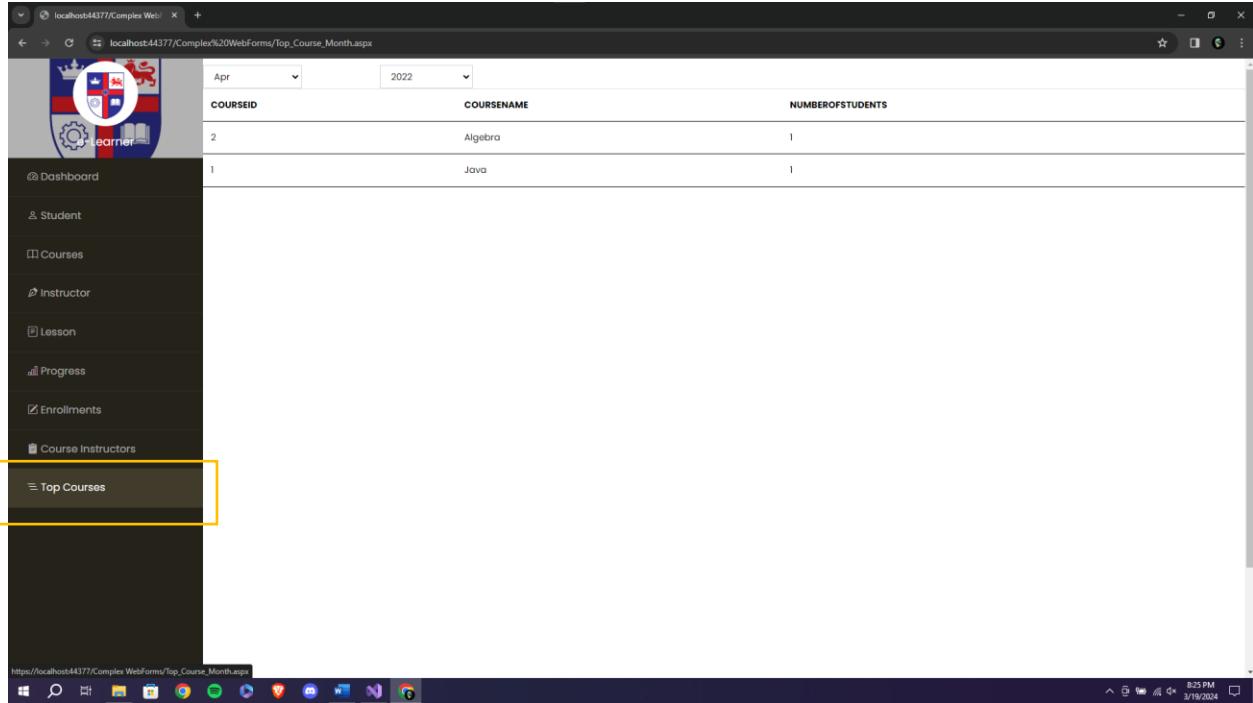


Figure 46. Sidebar for navigation

The highlighted section is the navigation bar for the website with a beautiful logo of the e-Learner online platform. Clicking any one of the names on the sidebar directs the student to a different page as mentioned.

Given is an example of directing from dashboard to Top Courses Page



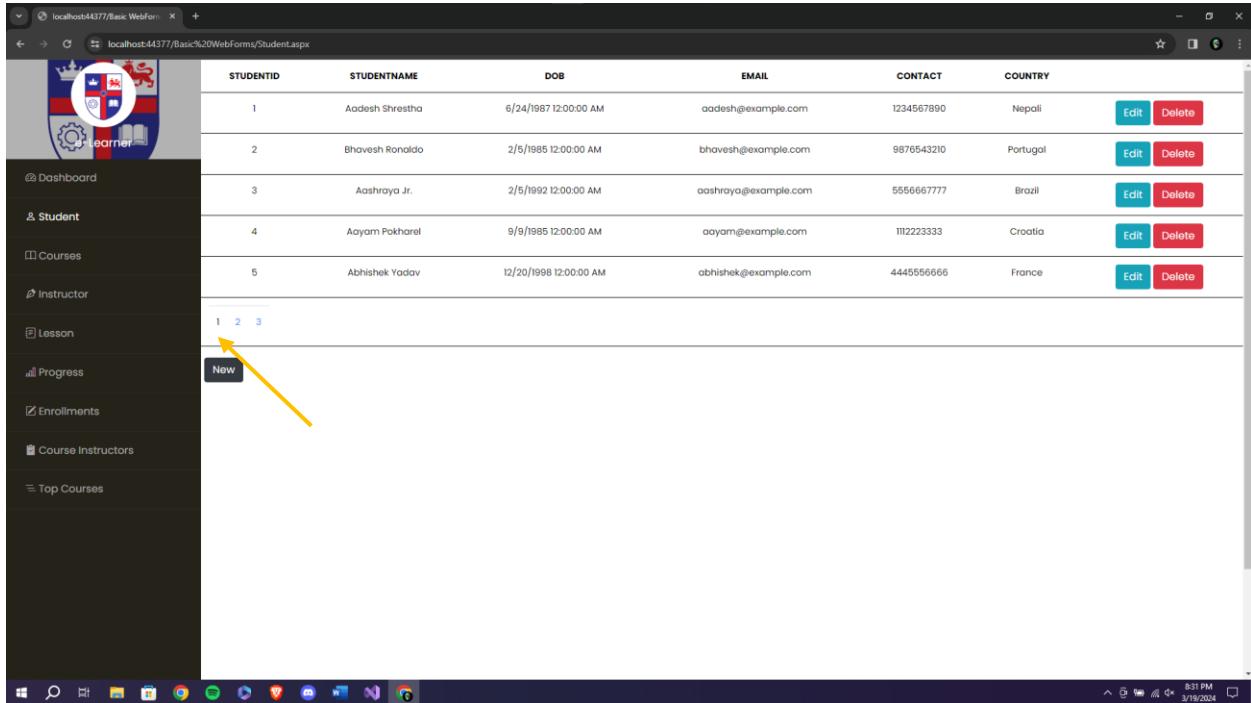
The screenshot shows a web application interface. On the left is a dark sidebar with various navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The 'Top Courses' link is highlighted with a yellow box. The main content area displays a table titled 'Top_Course_Month.aspx' for April 2022. The table has columns for COURSEID, COURSENAME, and NUMBEROFSTUDENTS. It shows two rows: one for Algebra with 1 student and another for Java with 1 student.

COURSEID	COURSENAME	NUMBEROFSTUDENTS
2	Algebra	1
1	Java	1

Figure 47. Navigated to Top Courses Page

As the figure suggests, the user is now redirected to the Top courses page where for given month and given year the top enrolled courses can be viewed. Pages can be smoothly navigated back and forth with minimum effort.

Using the Form



The screenshot shows a web-based application interface for managing student data. On the left, there is a vertical sidebar with a logo at the top and several navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. Below these links is a Windows taskbar with various icons.

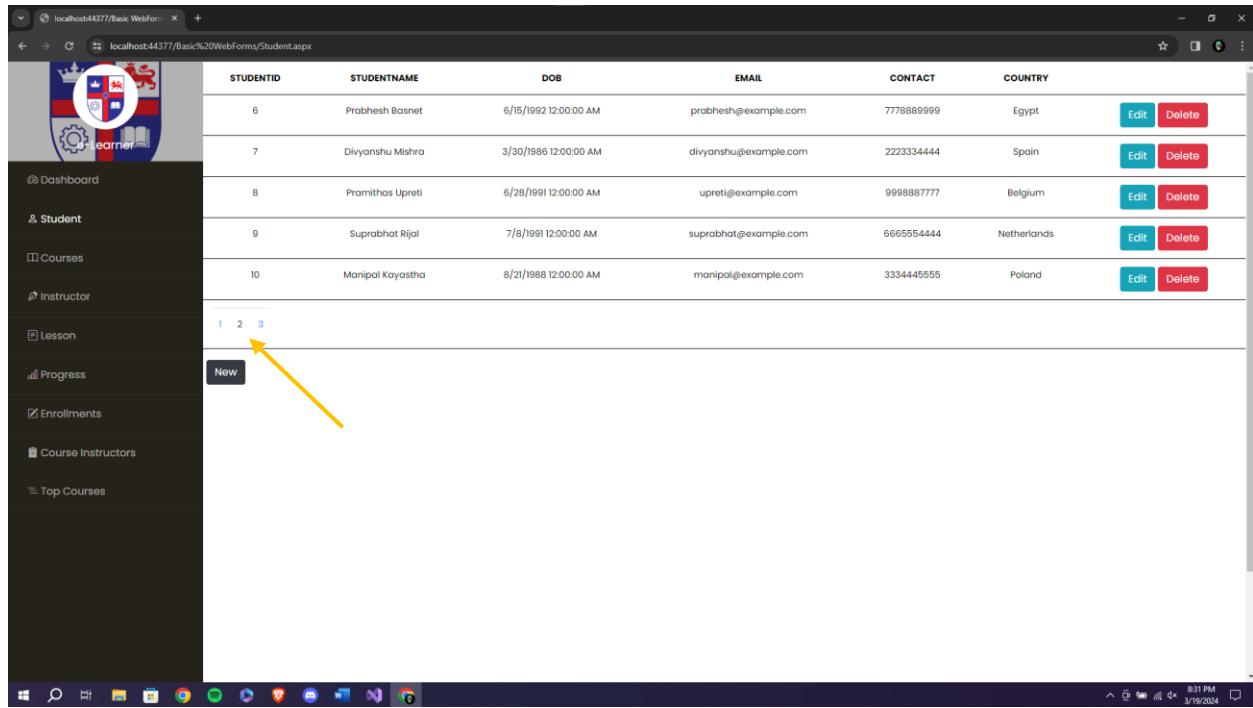
The main content area displays a table titled "StudentList" with the following columns: STUDENTID, STUDENTNAME, DOB, EMAIL, CONTACT, and COUNTRY. The table contains five rows of data:

STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	Nepali
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France

Below the table, there are three small numbered buttons (1, 2, 3) and a "New" button. A yellow arrow points from the text "student in the current table page." to the number "2".

Figure 48. Table Page 1 of view

Viewing data is very easy thanks to the use of available technology. In the given screen, the student is displayed with student information inside the student page. Currently the user is accessing data from the first table view that is “1” and highlighted in black. The page displays 5 values each and to view next 5 data, user can press the button ‘2’ right below the table. The arrow shows the student in the current table page.



The screenshot shows a web-based application interface. On the left is a dark sidebar with various navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main content area displays a table titled "StudentList" with columns: STUDENTID, STUDENTNAME, DOB, EMAIL, CONTACT, and COUNTRY. The table contains 10 rows of student data. At the bottom of the table, there are three small blue buttons labeled 1, 2, and 3. A yellow arrow points from the text "After clicking the '2' button, it turns black and indicates the user has successfully navigated to the second table." to the number 2. Below the table is a "New" button.

STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY
6	Probhesh Basnet	6/15/1992 12:00:00 AM	probhesh@example.com	7778889999	Egypt
7	Divyanshu Mishra	3/30/1986 12:00:00 AM	divyanshu@example.com	2223334444	Spain
8	Pramithas Upreti	6/28/1991 12:00:00 AM	upreti@example.com	9998887777	Belgium
9	Suprabhat Rijal	7/8/1991 12:00:00 AM	suprabhat@example.com	6665554444	Netherlands
10	Manipal Kayastha	8/21/1988 12:00:00 AM	manipal@example.com	3334445555	Poland
1 2 3					
New					

Figure 49. Table page 2

After clicking the '2' button, it turns black and indicates the user has successfully navigated to the second table. Similarly, for every pages on the website, data can be accessed and viewed easily without any hassle.

Some pages have values that can be filtered for easy viewing of the user. In the screen below, the user is accessing Course Instructors page, where the user can filter Courses and the table displays every Instructors for the selected Course.

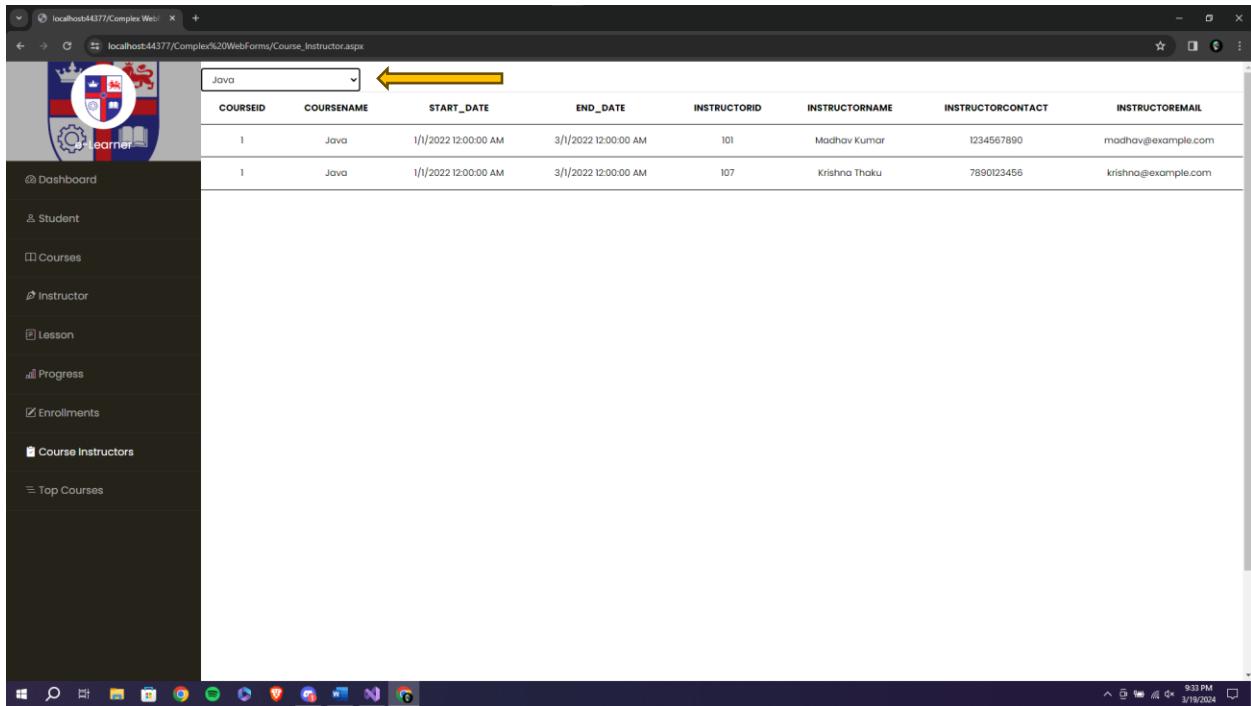


Figure 50. Dropdown menu for Course Instructors

Clicking on “Java” opens up a dropdown menu where all the courses name can be accessed and instructors for specific courses can be displayed.

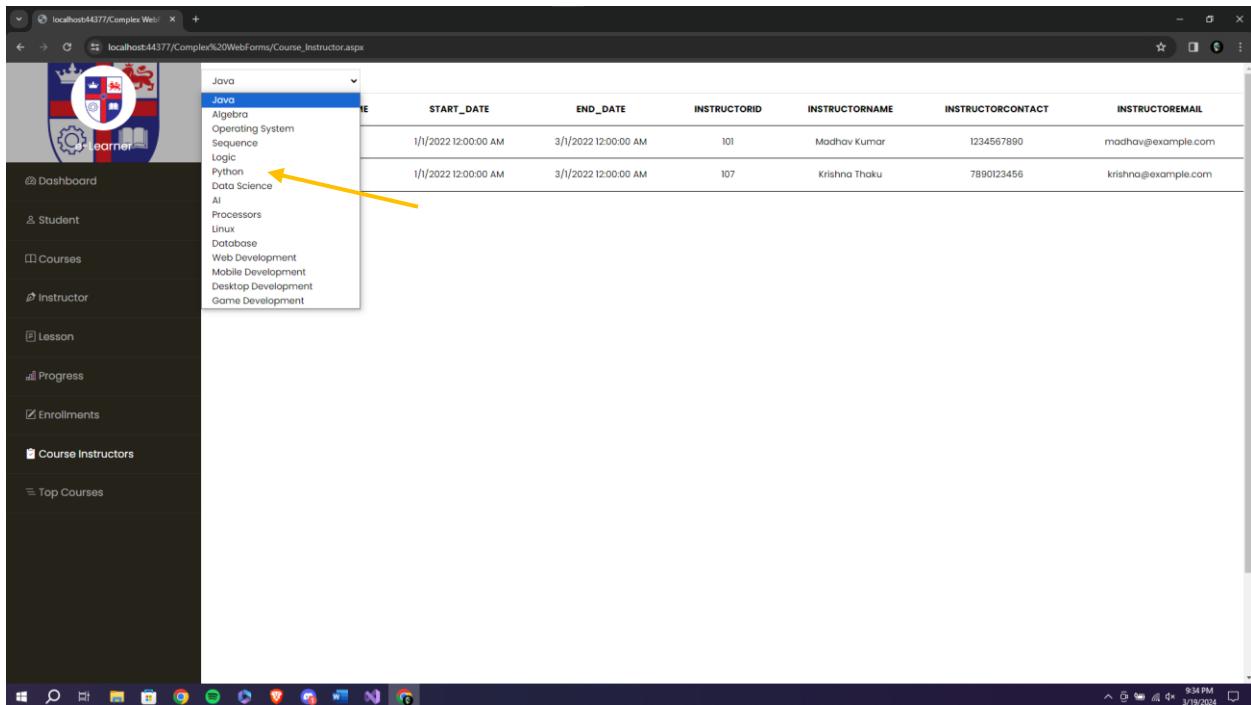
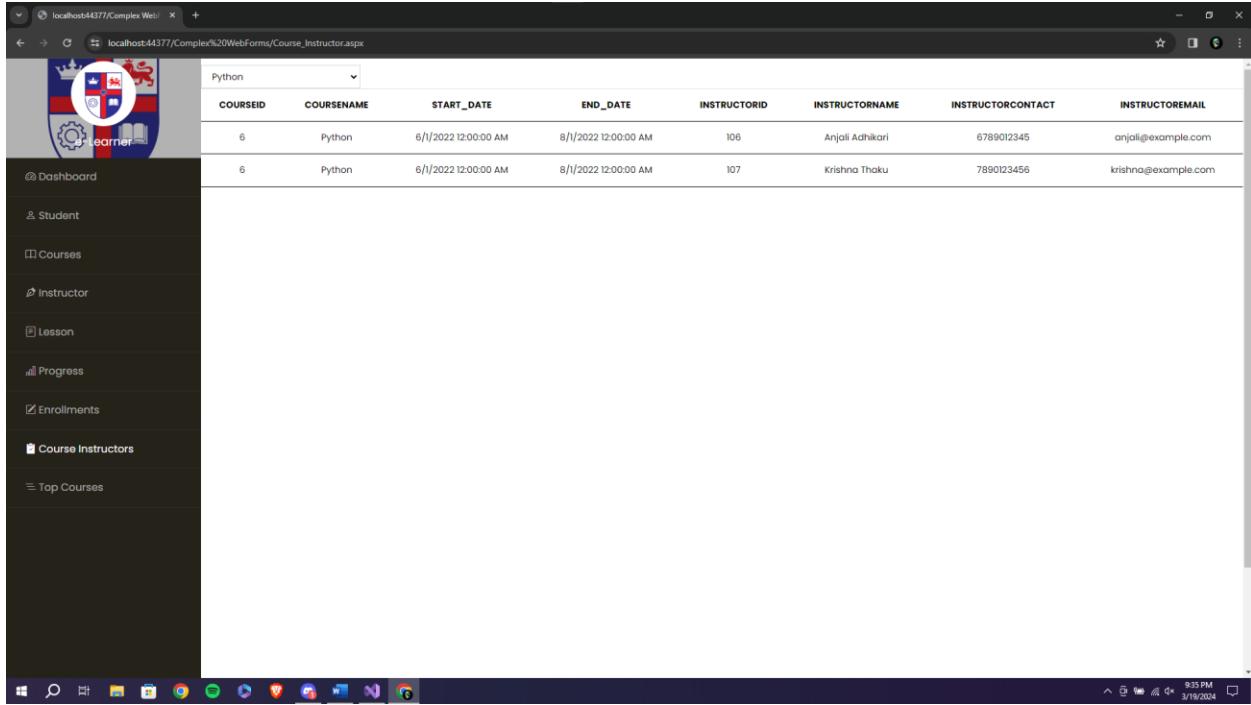


Figure 51. Dropdown after click



The screenshot shows a web application interface for managing courses. On the left, there is a vertical sidebar with a logo at the top and several navigation items: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors (which is currently selected), and Top Courses. The main content area displays a table titled "Python" with the following data:

COURSEID	COURSENAME	START_DATE	END_DATE	INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
6	Python	6/1/2022 12:00:00 AM	8/1/2022 12:00:00 AM	106	Anjali Adhikari	6789012345	anjali@example.com
6	Python	6/1/2022 12:00:00 AM	8/1/2022 12:00:00 AM	107	Krishna Thaku	7890123456	krishna@example.com

Figure 52. New table according to Course

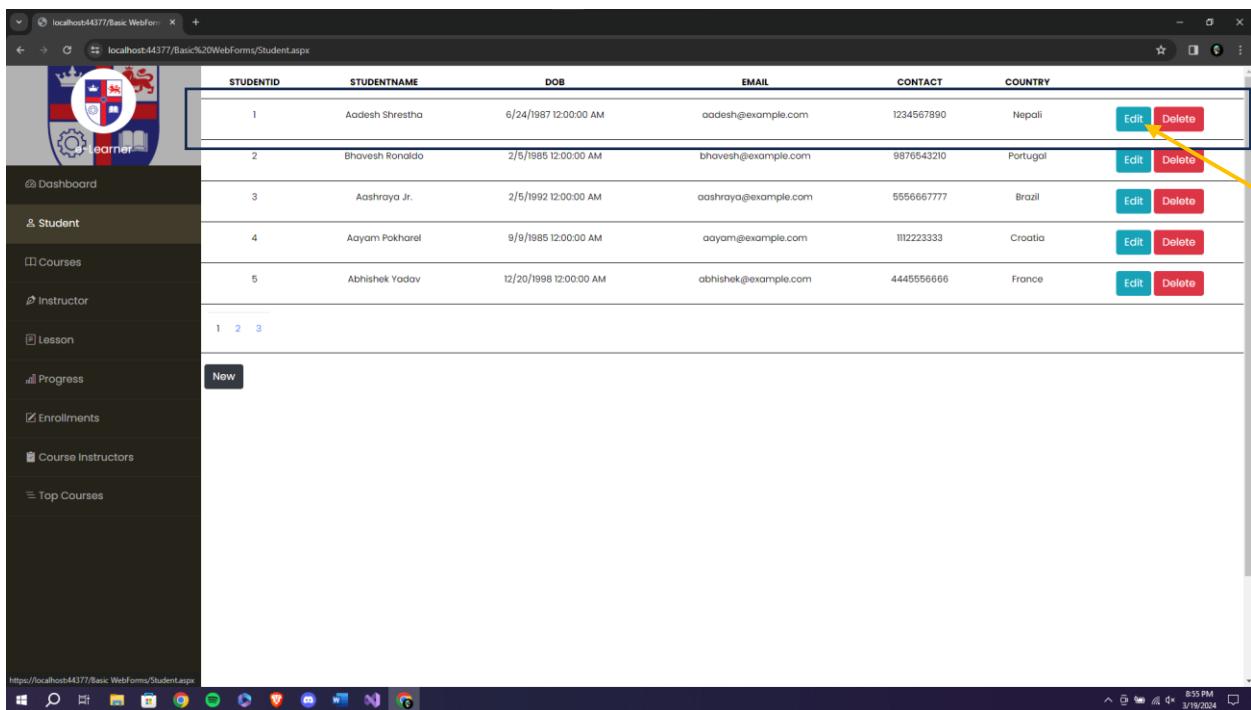
The user has accessed the course “Python” in the screen given above.

Similarly, the pages Enrollments and Top Courses are included with the same features. Clicking on the dropdown menu easily displays appropriate data to display filtered table.

Editing, Deleting and Inserting new Values

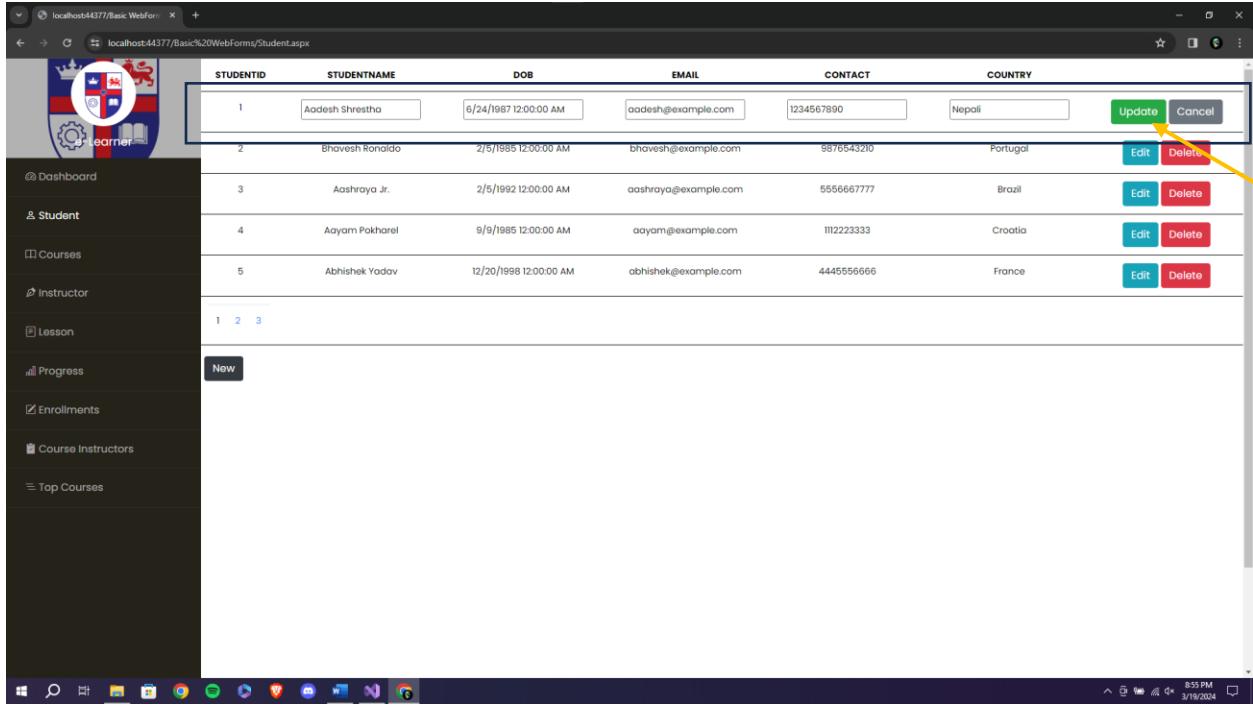
Editing the value can be necessary due to multiple reasons, if the user chooses to edit value they can simply go to the page where they want to edit the table and click on edit button of the preferred column.

Here, in this example; the Country for the student Aadesh Shrestha is set to Nepali which doesn't exists. To edit the country name to Nepal; the user can simply click on the edit button



STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY	
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	Nepali	<button>Edit</button> <button>Delete</button>
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal	<button>Edit</button> <button>Delete</button>
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil	<button>Edit</button> <button>Delete</button>
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia	<button>Edit</button> <button>Delete</button>
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France	<button>Edit</button> <button>Delete</button>

Figure 53. Edit Button



STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY	
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	Nepali	<button>Update</button> <button>Cancel</button>
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal	<button>Edit</button> <button>Delete</button>
3	Aashraya Jr.	2/9/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil	<button>Edit</button> <button>Delete</button>
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia	<button>Edit</button> <button>Delete</button>
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France	<button>Edit</button> <button>Delete</button>

1 2 3

New

Figure 54. Updating Values

The user is presented with editable textfields where the values can be edited. Now the user can simply rename the country to correct name. The user can also cancel the edit process anytime by pressing the “Cancel” button right next to the “Edit” button.

STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY	
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	Nepal	<button>Edit</button> <button>Delete</button>
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal	<button>Edit</button> <button>Delete</button>
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil	<button>Edit</button> <button>Delete</button>
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia	<button>Edit</button> <button>Delete</button>
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France	<button>Edit</button> <button>Delete</button>

Figure 55. Updated Value

Congratulations! You have successfully edited a value in the table. To change values in any table if editable, you can simply follow the same steps.

Deleting a value is even easier than editing an already existing value.

In the example below, the new student has registered with meaningless credentials. To delete the user, one can simple click the delete button(in red) to delete the student. An alert box appears asking the user to ensure that the intended student is being deleted.

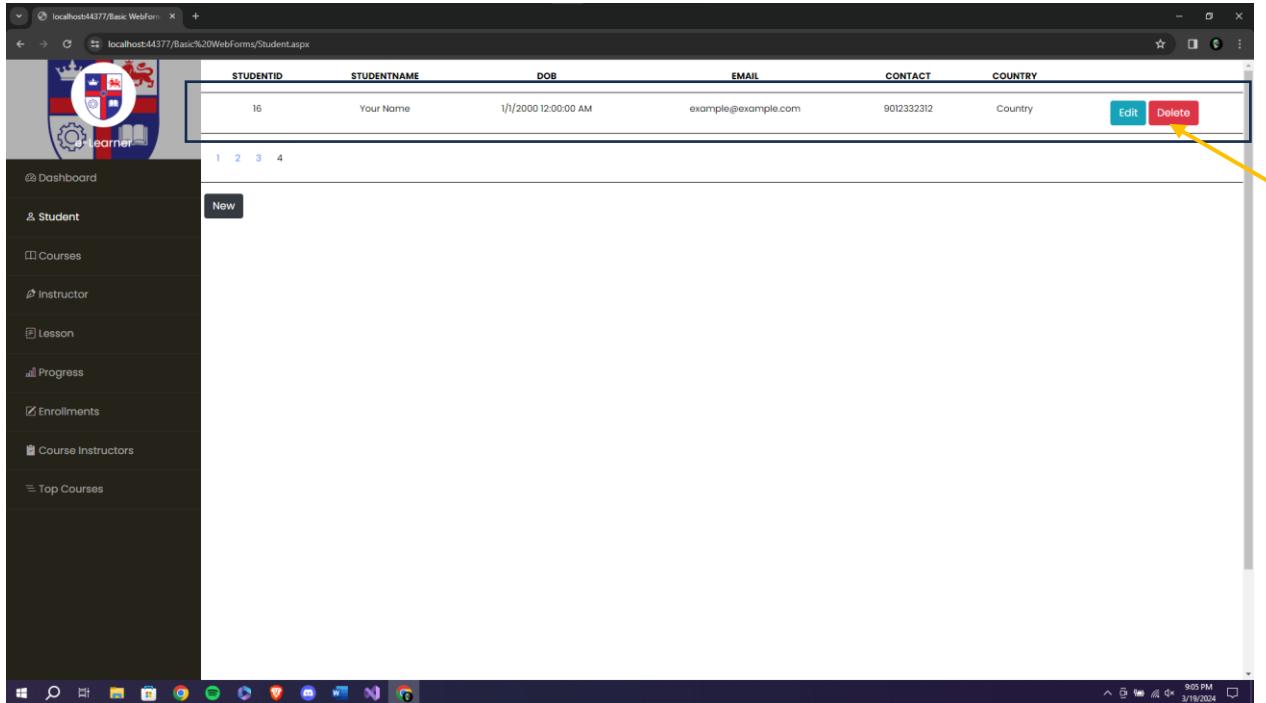


Figure 56. Deleting Data

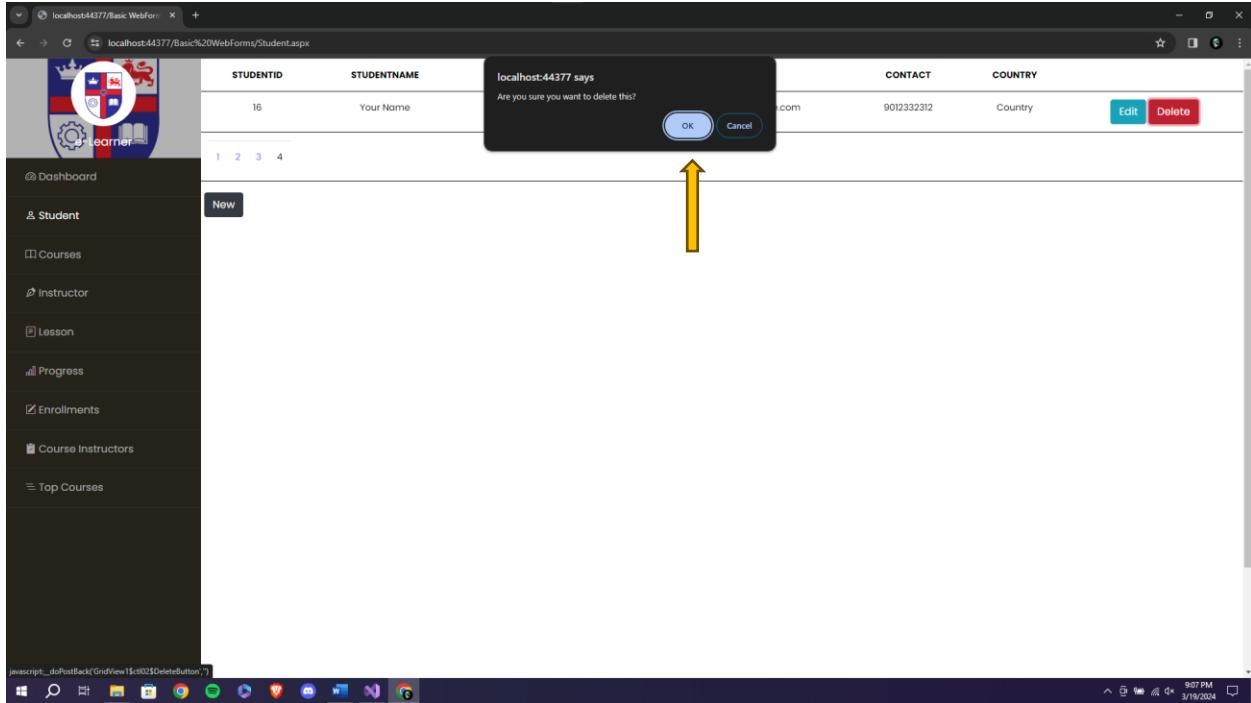


Figure 57. Alert on Delete

Pressing the OK button permanently deletes the student from the database. Here, no one with the StudentId 116 appears to be displayed.

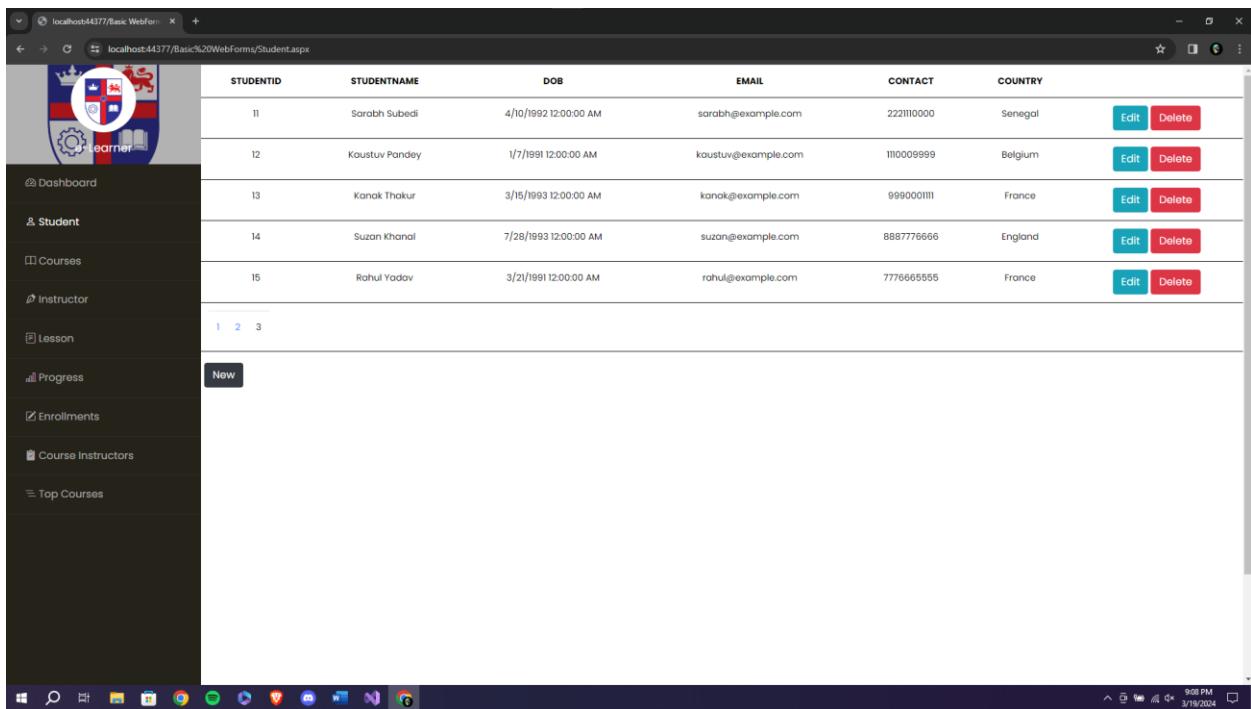
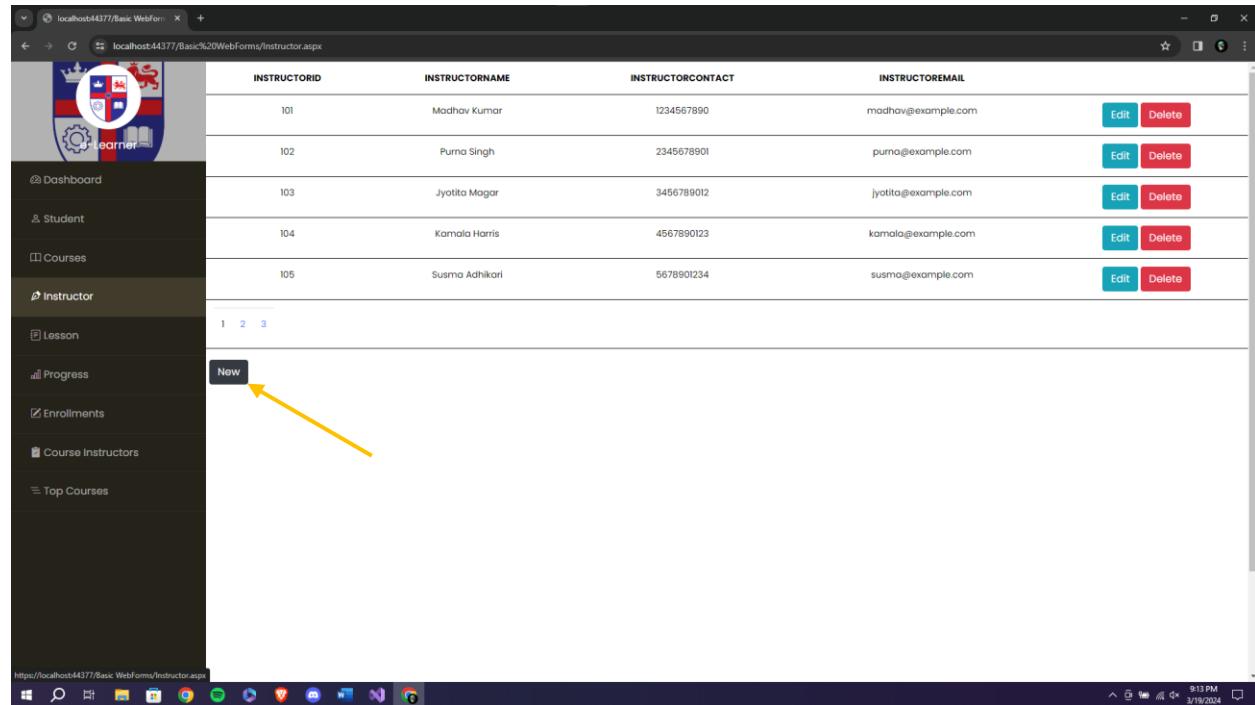


Figure 58. Deleted Value

The same process can be followed to delete data in tables which can be altered.

Adding a data to the table can be achieved by clicking the “New” button.

Below, the user is accessing Instructor page to view all Instructors working in e-Learner online platform. To add a new instructor, one can simply click the “New” button right below the table.



A screenshot of a web browser displaying the 'Instructor.aspx' page from 'localhost:44377'. The page shows a table of five instructors with columns: INSTRUCTORID, INSTRUCTORNAME, INSTRUCTORCONTACT, and INSTRUCTOREMAIL. Each row has 'Edit' and 'Delete' buttons. Below the table is a navigation bar with links 1, 2, and 3. At the bottom left, there is a 'New' button with a yellow arrow pointing to it. On the left side, there is a sidebar with various navigation links: Dashboard, Student, Courses, Instructor (which is selected), Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The URL in the address bar is 'https://localhost:44377/Basic Webforms/Instructor.aspx'.

INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL	
101	Madhav Kumar	1234567890	madhav@example.com	<button>Edit</button> <button>Delete</button>
102	Purna Singh	2345678901	purna@example.com	<button>Edit</button> <button>Delete</button>
103	Jyotita Magar	3456789012	jyotita@example.com	<button>Edit</button> <button>Delete</button>
104	Kamala Harris	4567890123	kamala@example.com	<button>Edit</button> <button>Delete</button>
105	Susma Adhikari	5678901234	susma@example.com	<button>Edit</button> <button>Delete</button>

Figure 59. Button to add new data

Clicking the “New” button opens up a form view where the user can simply input desired values to the respective columns.

The screenshot shows a web application interface. On the left is a sidebar with navigation links: Dashboard, Student, Courses, Instructor (selected), Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main content area displays a table of instructors with columns: INSTRUCTORID, INSTRUCTORMNAME, INSTRUCTORCONTACT, and INSTRUCTOREMAIL. Each row has 'Edit' and 'Delete' buttons. Below the table is a navigation bar with links 1, 2, and 3. A modal window is open in the center, titled 'INSTRUCTORID:' with a text input field containing '106'. It also contains fields for 'INSTRUCTORMNAME:' (empty), 'INSTRUCTORCONTACT:' (empty), and 'INSTRUCTOREMAIL:' (empty). At the bottom of the modal are 'Insert' and 'Cancel' buttons.

Figure 60. Form for data insertion

Now lets give a dummy data to add a new Instructor to e-Learners online platform

This screenshot shows the same web application after a new instructor has been added. The table now includes a new row with INSTRUCTORID 116, INSTRUCTORMNAME 'Lekhnath Kotuwal', INSTRUCTORCONTACT '9123234567', and INSTRUCTOREMAIL 'kotuwal@example.com'. A yellow arrow points from the bottom-left towards the 'Insert' button in the modal window, indicating the action taken.

Figure 61. Inserting the data

Confirm the addition of the data by clicking on “Insert” button. The process can be cancelled anytime by pressing the cancel button.

The screenshot shows a web browser window with the URL `localhost:4377/Basic%20Webforms/Instructor.aspx`. The main content area displays a table with four columns: INSTRUCTORID, INSTRUCTORMNAME, INSTRUCTORCONTACT, and INSTRUCTOREMAIL. A single row is present with values: 116, Lekhnath Katuwal, 9123234567, and `katuwal@example.com`. To the right of the table are two buttons: a blue "Edit" button and a red "Delete" button. Below the table, there is a navigation menu on the left with items like Dashboard, Student, Courses, Instructor (which is selected and highlighted in blue), Lesson, Progress, Enrollments, Course Instructors, and Top Courses. A "New" button is located at the bottom of the main content area. The status bar at the bottom right shows the date and time as 3/19/2024 8:19 PM.

Figure 62. New Instructor Created

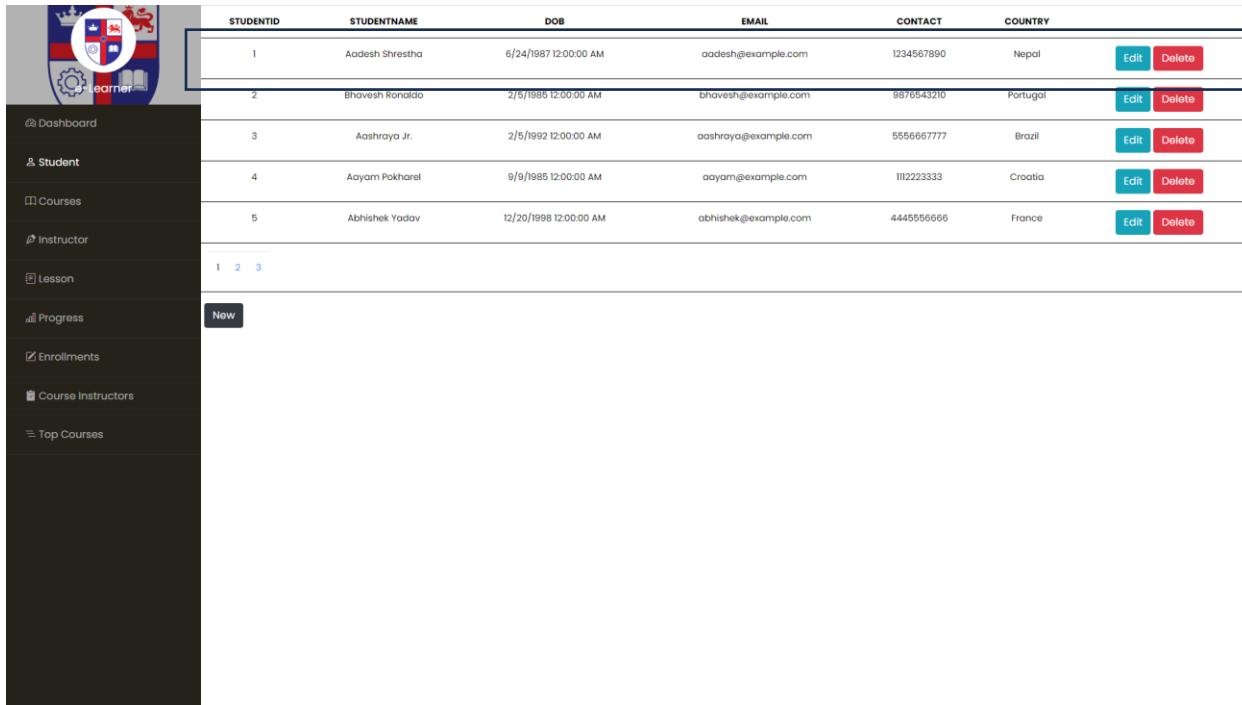
Cheers! A new Instructor has been successfully added to the table.

The user can add new data to tables by simply clicking on the “New” button displayed below the tables.

13 Testing

13.1 Basic Forms

13.1.1 Student Page



STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY	Actions
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	Nepal	<button>Edit</button> <button>Delete</button>
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	8876543210	Portugal	<button>Edit</button> <button>Delete</button>
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil	<button>Edit</button> <button>Delete</button>
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia	<button>Edit</button> <button>Delete</button>
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France	<button>Edit</button> <button>Delete</button>

Figure 63. Student Edit

STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY		
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	United Kingdom	<button>Update</button>	<button>Cancel</button>
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal	<button>Edit</button>	<button>Delete</button>
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil	<button>Edit</button>	<button>Delete</button>
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia	<button>Edit</button>	<button>Delete</button>
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France	<button>Edit</button>	<button>Delete</button>

1 2 3

New

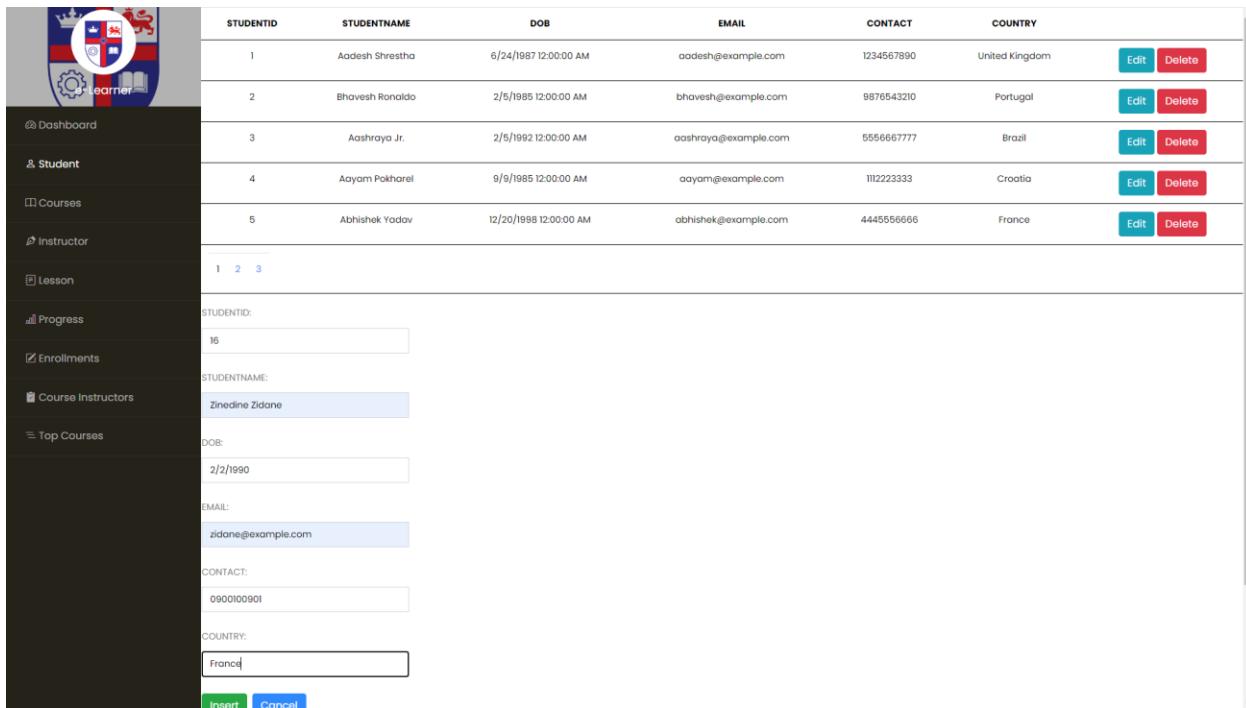
Figure 64. Student Update

STUDENTID	STUDENTNAME	DOB	EMAIL	CONTACT	COUNTRY		
1	Aadesh Shrestha	6/24/1987 12:00:00 AM	aadesh@example.com	1234567890	United Kingdom	<button>Edit</button>	<button>Delete</button>
2	Bhavesh Ronaldo	2/5/1985 12:00:00 AM	bhavesh@example.com	9876543210	Portugal	<button>Edit</button>	<button>Delete</button>
3	Aashraya Jr.	2/5/1992 12:00:00 AM	aashraya@example.com	5556667777	Brazil	<button>Edit</button>	<button>Delete</button>
4	Aayam Pokharel	9/9/1985 12:00:00 AM	aayam@example.com	1112223333	Croatia	<button>Edit</button>	<button>Delete</button>
5	Abhishek Yadav	12/20/1998 12:00:00 AM	abhishek@example.com	4445556666	France	<button>Edit</button>	<button>Delete</button>

1 2 3

New

Figure 65. Updated Student

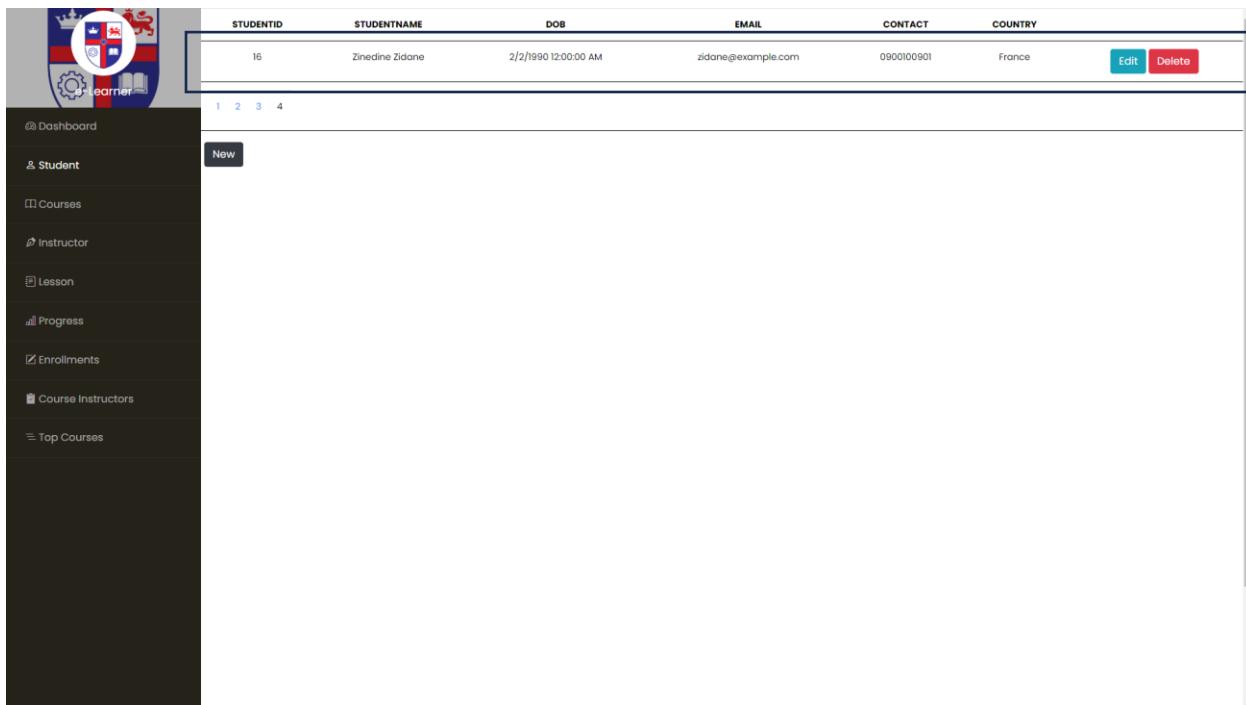


The screenshot shows a web-based application for managing student data. On the left is a sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main area displays a table of students with columns: STUDENTID, STUDENTNAME, DOB, EMAIL, CONTACT, and COUNTRY. Each row has 'Edit' and 'Delete' buttons. Below the table is a form for adding a new student:

STUDENTID:	<input type="text" value="16"/>
STUDENTNAME:	<input type="text" value="Zinedine Zidane"/>
DOB:	<input type="text" value="2/2/1990"/>
EMAIL:	<input type="text" value="zidane@example.com"/>
CONTACT:	<input type="text" value="0900100901"/>
COUNTRY:	<input type="text" value="France"/>

At the bottom of the form are 'Insert' and 'Cancel' buttons.

Figure 66. Inserting Student



The screenshot shows the same application interface after a new student has been added. The table now includes a new row for Zinedine Zidane with STUDENTID 16, DOB 2/2/1990, EMAIL zidane@example.com, CONTACT 0900100901, and COUNTRY France. The 'Edit' and 'Delete' buttons are present for this new entry.

Figure 67. Inserted New Student

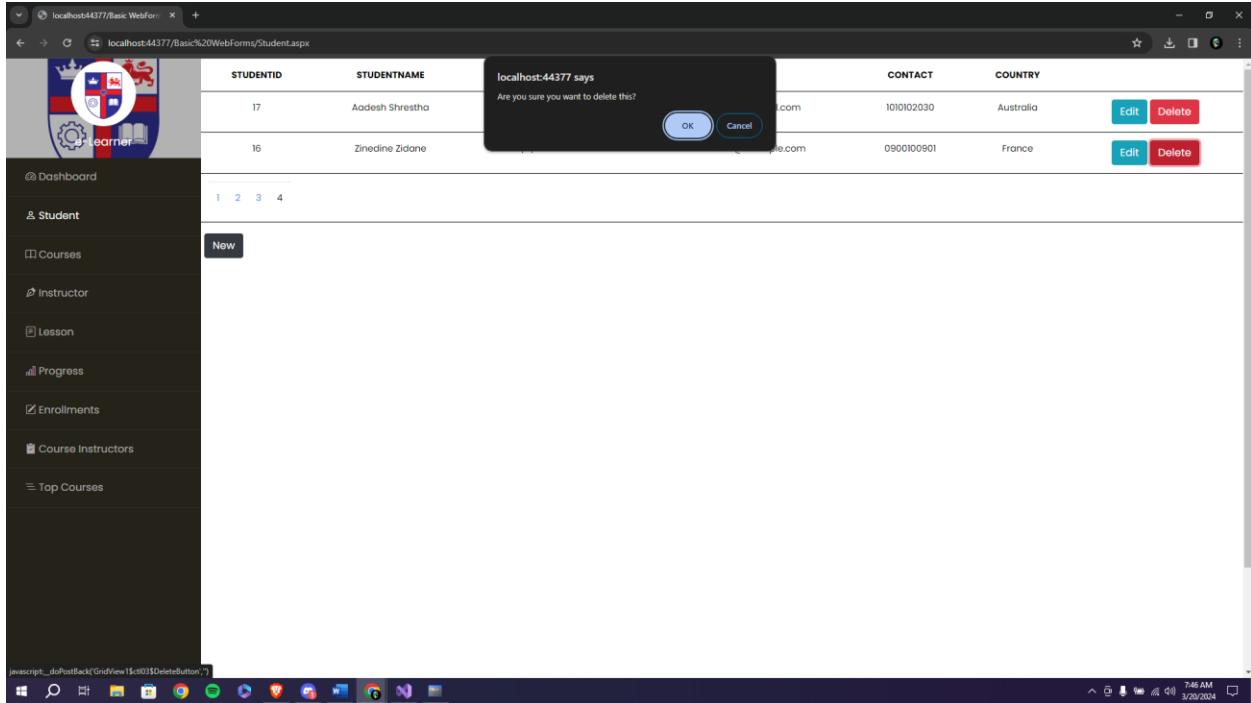


Figure 68. Deleting Student data

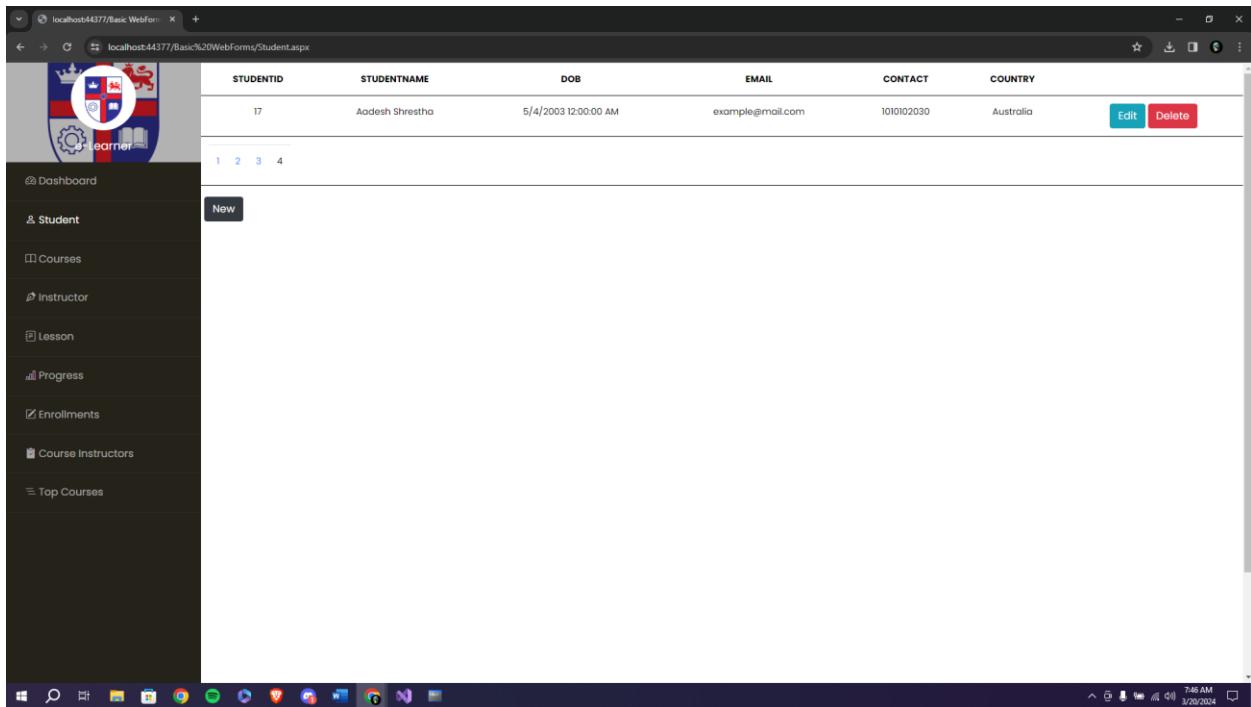


Figure 69. Student Data Deleted

13.1.2 Courses Page

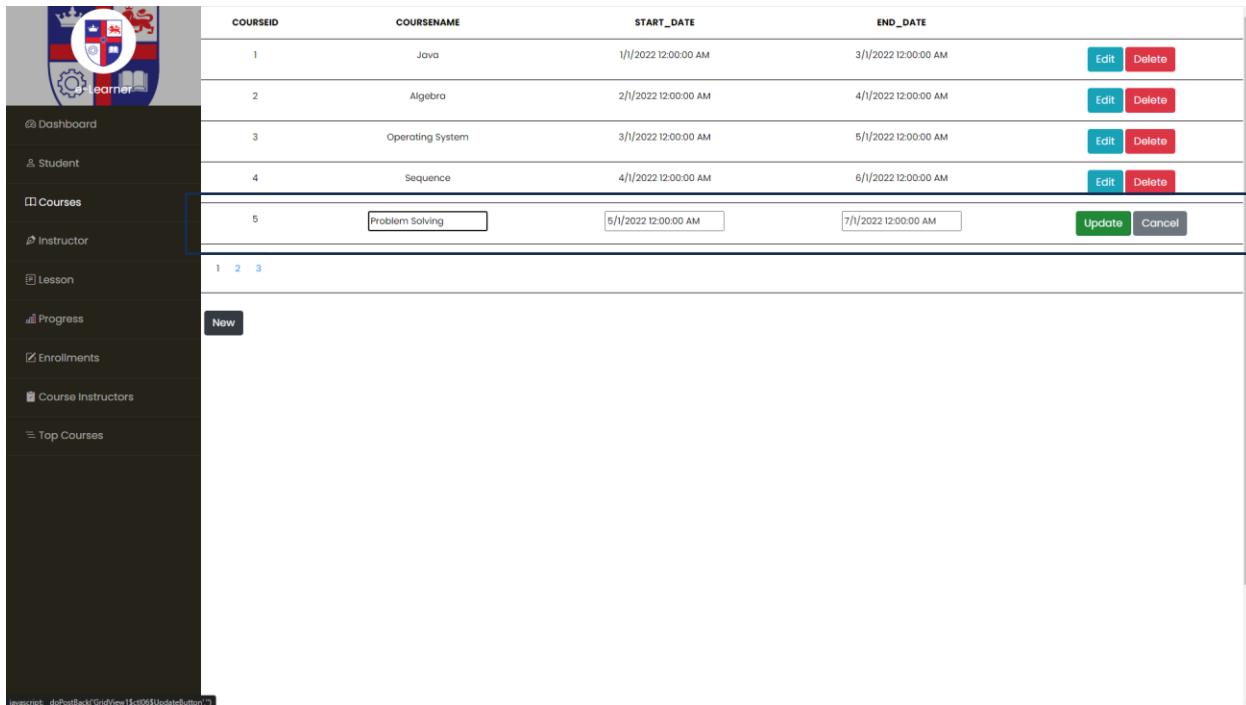


The screenshot shows a web-based application interface for managing courses. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses (selected), Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main content area displays a grid of course information:

COURSEID	COURSENAME	START_DATE	END_DATE	
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
3	Operating System	3/1/2022 12:00:00 AM	5/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
4	Sequence	4/1/2022 12:00:00 AM	6/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
5	Logic	5/1/2022 12:00:00 AM	7/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>

Below the grid, there are navigation links (1, 2, 3) and a "New" button.

Figure 70. Course Edit



This screenshot shows the same application interface after a course has been edited. The course entry for ID 5 has been updated to "Problem Solving". The "Update" and "Cancel" buttons are visible at the bottom right of the grid row.

COURSEID	COURSENAME	START_DATE	END_DATE	
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
3	Operating System	3/1/2022 12:00:00 AM	5/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
4	Sequence	4/1/2022 12:00:00 AM	6/1/2022 12:00:00 AM	<button>Edit</button> <button>Delete</button>
5	Problem Solving	5/1/2022 12:00:00 AM	7/1/2022 12:00:00 AM	<button>Update</button> <button>Cancel</button>

Figure 71. Course Update

COURSEID	COURSENAME	START_DATE	END_DATE		
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
3	Operating System	3/1/2022 12:00:00 AM	5/1/2022 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
4	Sequence	4/1/2022 12:00:00 AM	6/1/2022 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
5	Problem Solving	5/1/2022 12:00:00 AM	7/1/2022 12:00:00 AM	<button>Edit</button>	<button>Delete</button>

Figure 72. Updated Course

COURSEID	COURSENAME	START_DATE	END_DATE		
11	Database	1/1/2022 12:00:00 AM	1/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
12	Web Development	12/1/2022 12:00:00 AM	2/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
13	Mobile Development	1/1/2023 12:00:00 AM	3/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
14	Desktop Development	2/1/2023 12:00:00 AM	4/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
15	Game Development	3/1/2023 12:00:00 AM	5/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>

COURSEID:
16

COURSENAME:
UI/UX

START_DATE:
1/1/2022

END_DATE:
6/7/2023

Insert Cancel

Figure 73. Inserting Course

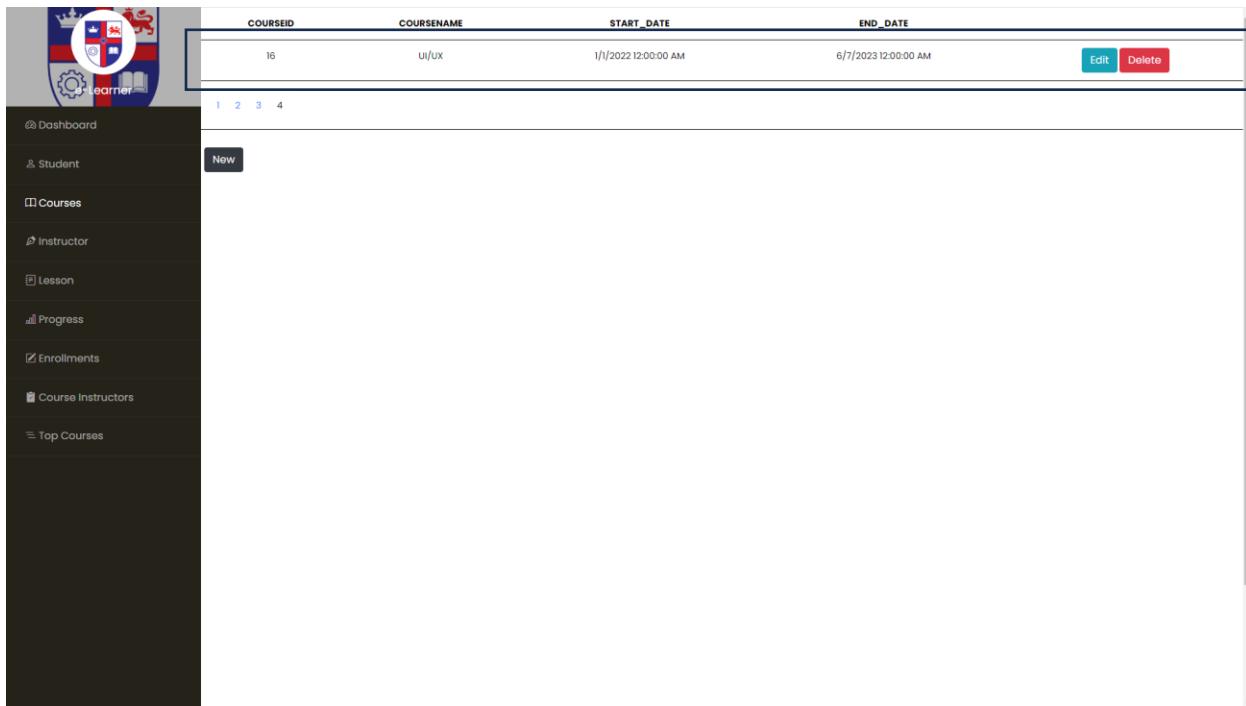


Figure 74. New Course Inserted

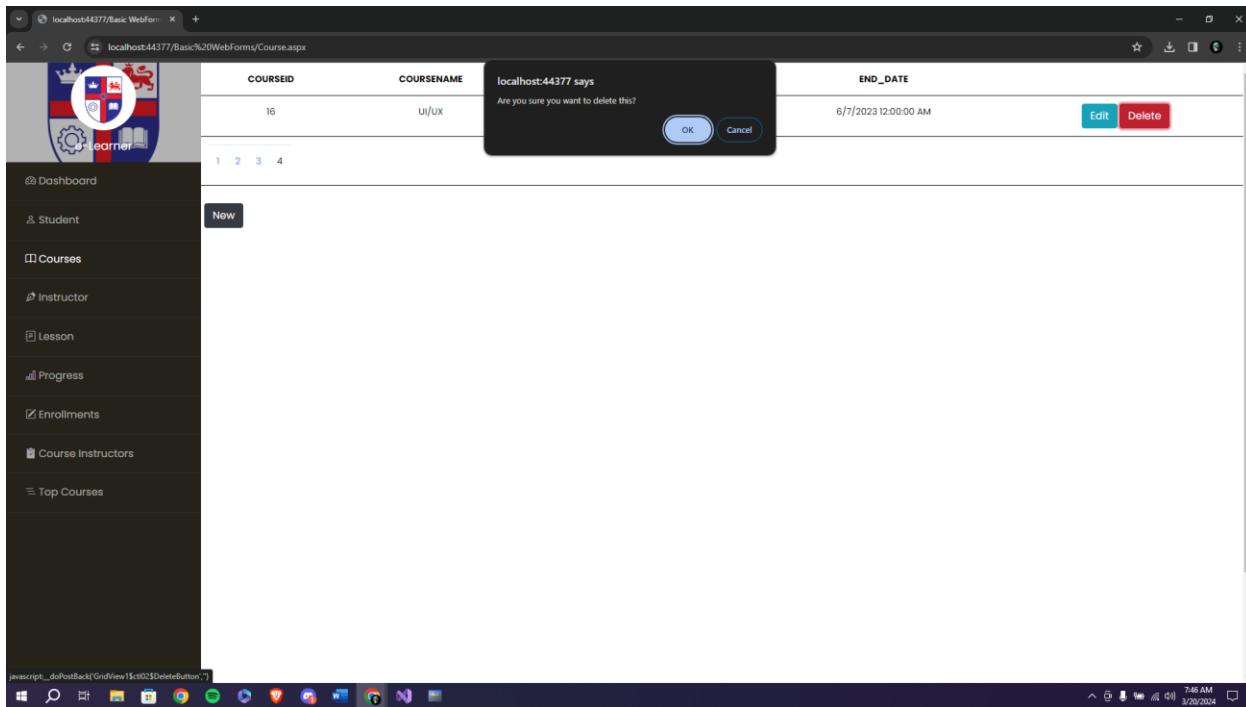


Figure 75. Deleting Course Data

COURSEID	COURSENAME	START_DATE	END_DATE		
11	Database	11/1/2022 12:00:00 AM	1/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
12	Web Development	12/1/2022 12:00:00 AM	2/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
13	Mobile Development	1/1/2023 12:00:00 AM	3/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
14	Desktop Development	2/1/2023 12:00:00 AM	4/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>
15	Game Development	3/1/2023 12:00:00 AM	5/1/2023 12:00:00 AM	<button>Edit</button>	<button>Delete</button>

Figure 76. Deleted Course Data

13.1.3 Instructor Pages

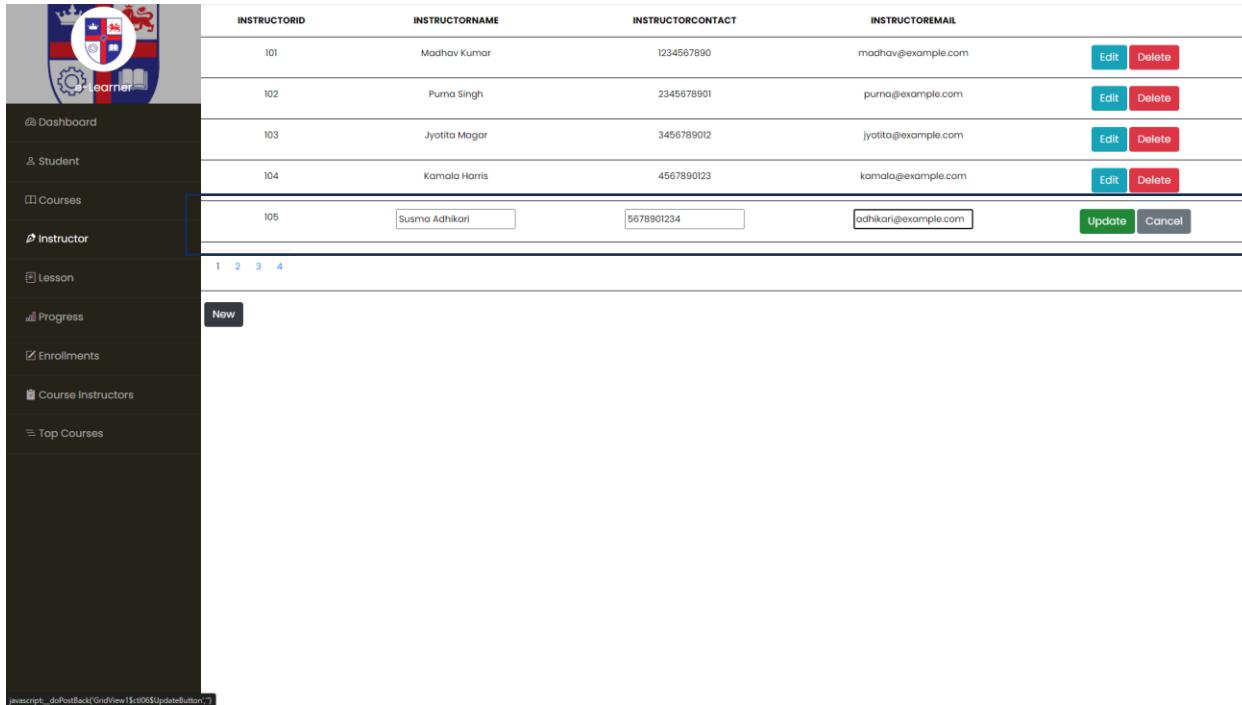


This screenshot shows the 'Instructor Edit' page. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses, Instructor (which is selected), Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main area contains a grid of instructor information:

INSTRUCTORID	INSTRUCTORMNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL	
101	Madhav Kumar	1234567890	madhav@example.com	<button>Edit</button> <button>Delete</button>
102	Purna Singh	2345678901	purna@example.com	<button>Edit</button> <button>Delete</button>
103	Jyotita Magar	3456789012	jyotita@example.com	<button>Edit</button> <button>Delete</button>
104	Kamala Harris	4567890123	kamala@example.com	<button>Edit</button> <button>Delete</button>
105	Susma Adhikari	5678901234	susma@example.com	<button>Edit</button> <button>Delete</button>

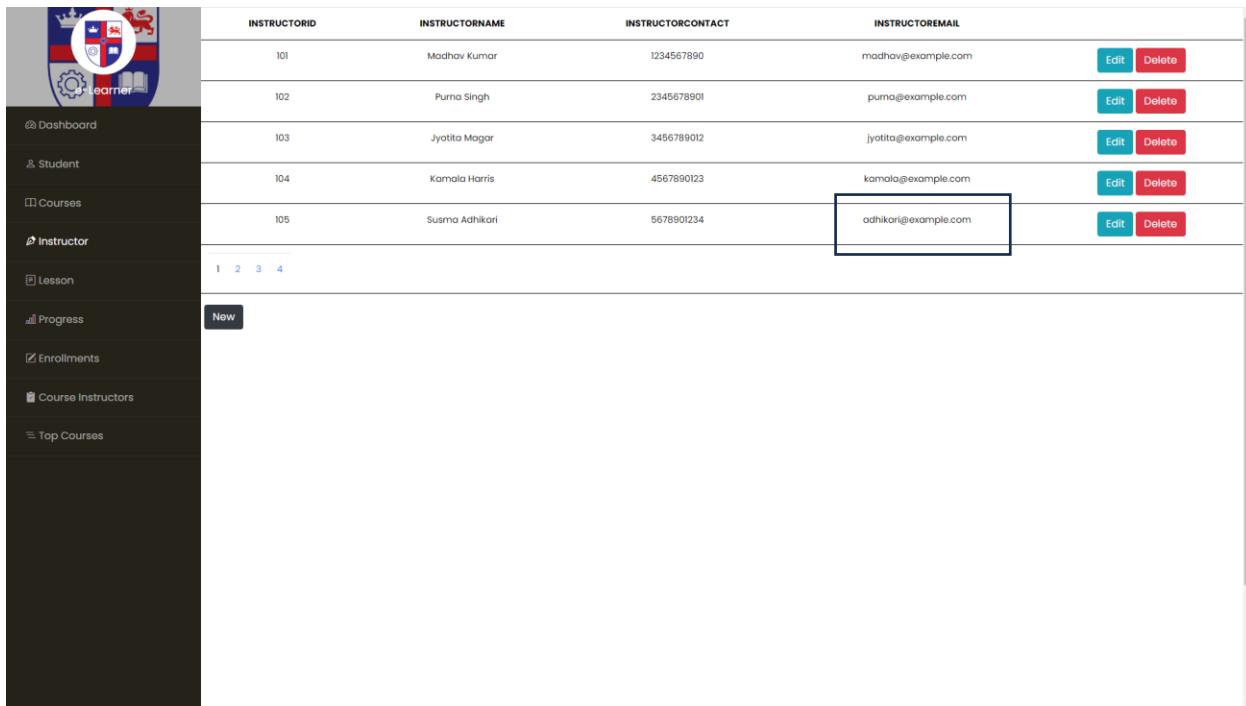
Below the grid is a 'New' button and a footer with page numbers (1, 2, 3, 4). A script tag at the bottom contains the value 'javascript:_doPostBack(Gridview1\$ctl06\$Editbutton,"')'.

Figure 77. Instructor Edit



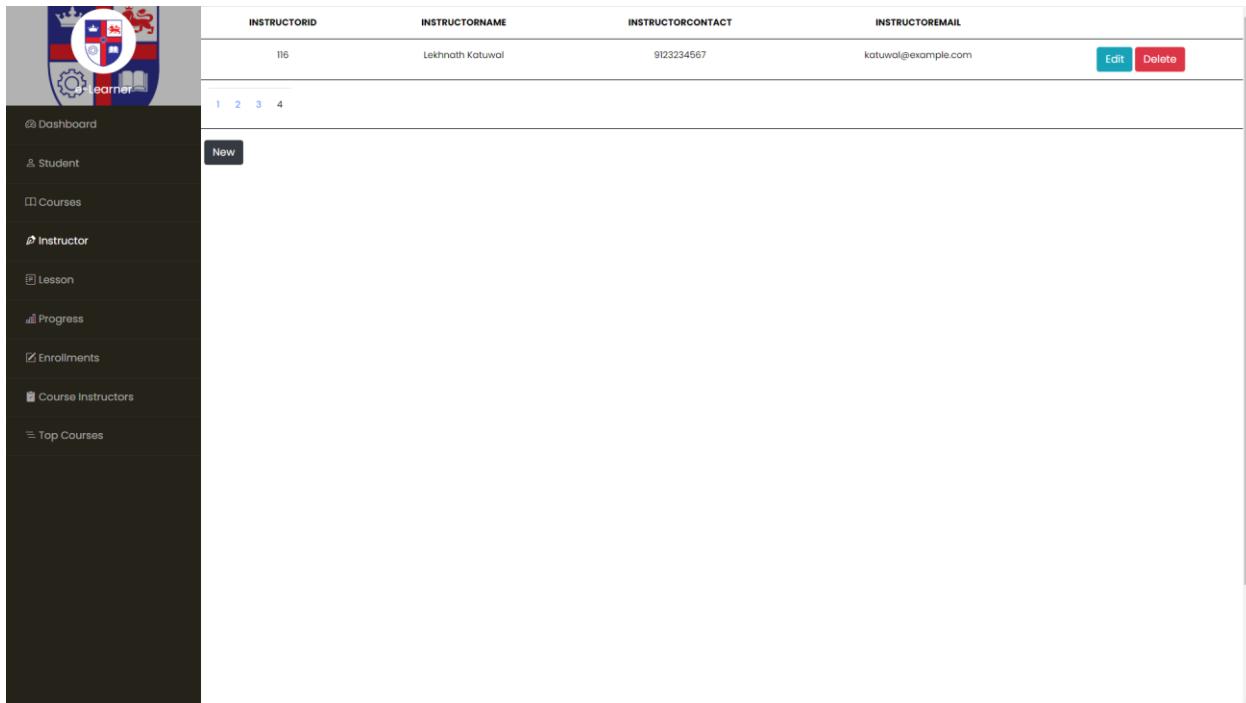
This screenshot shows the 'Instructor Update' page, similar to Figure 77 but with edit fields active. The sidebar and grid structure are identical. The 'INSTRUCTORMNAME' field for row 105 is highlighted with a red border, and the 'INSTRUCTORCONTACT' and 'INSTRUCTOREMAIL' fields are also highlighted. At the bottom right of the grid are 'Update' and 'Cancel' buttons. A script tag at the bottom contains the value 'javascript:_doPostBack(Gridview1\$ctl06\$Updatebutton,"')'.

Figure 78. Instructor Update



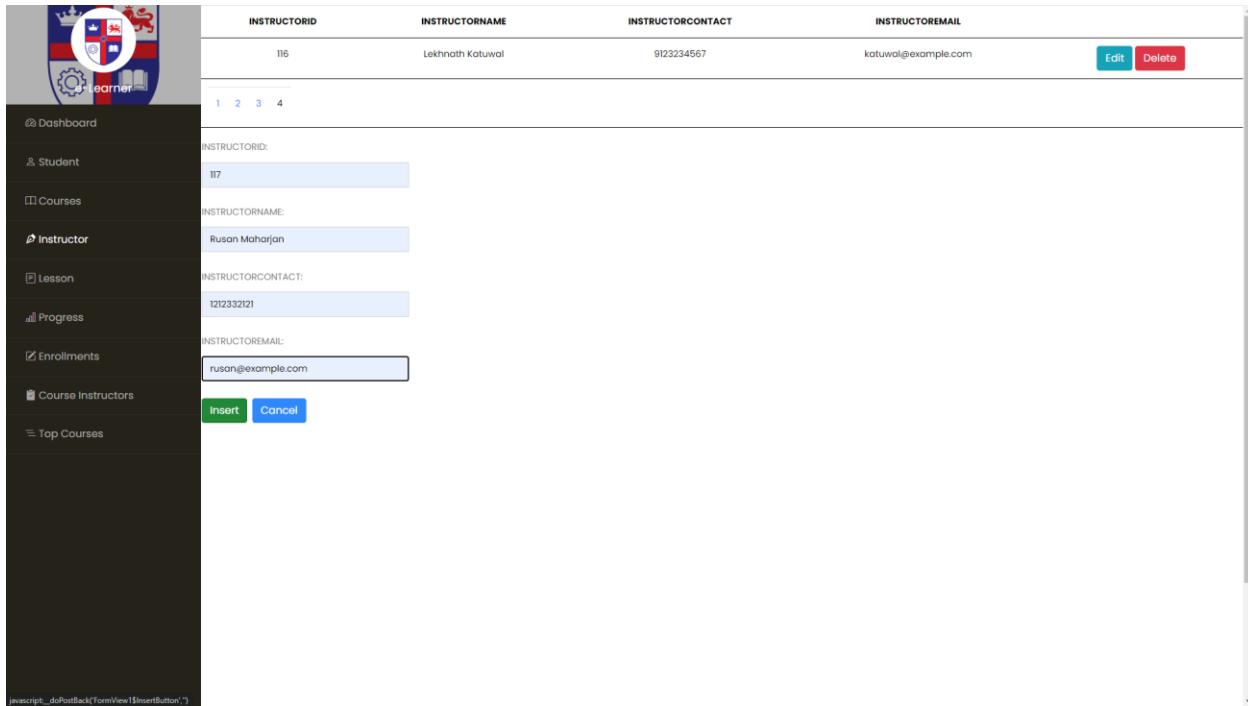
INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL		
101	Madhav Kumar	1234567890	madhav@example.com	<button>Edit</button>	<button>Delete</button>
102	Purna Singh	2345678901	purna@example.com	<button>Edit</button>	<button>Delete</button>
103	Jyotita Magar	3456789012	jyotita@example.com	<button>Edit</button>	<button>Delete</button>
104	Kamala Harris	4567890123	kamala@example.com	<button>Edit</button>	<button>Delete</button>
105	Susma Adhikari	5678901234	odhikari@example.com	<button>Edit</button>	<button>Delete</button>

Figure 79. Updated Instructor



INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL		
116	Lekhnath Katuwal	9123234567	katuwal@example.com	<button>Edit</button>	<button>Delete</button>
1	2	3	4		

Figure 80. Only one value on page 4

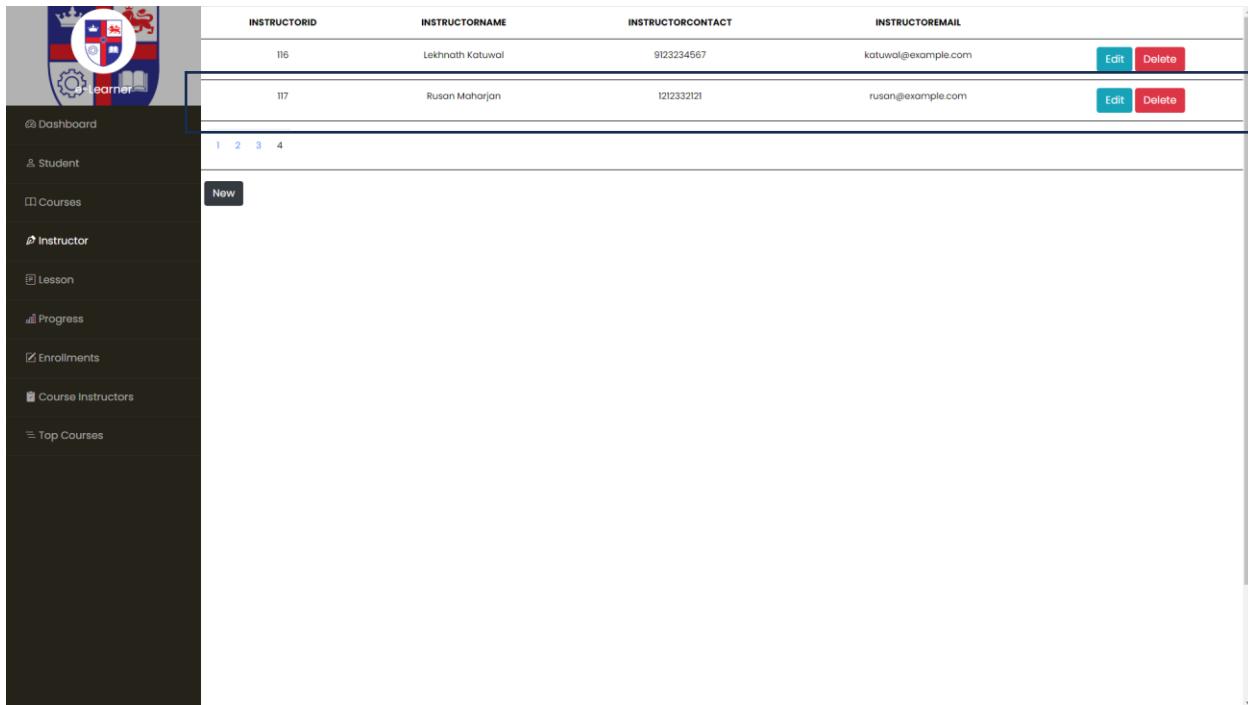


The screenshot shows a web-based application interface for managing instructors. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses, Instructor (selected), Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main area has a header 'INSTRUCTORID' and fields for 'INSTRUCTORMNAME', 'INSTRUCTORCONTACT', and 'INSTRUCTOREMAIL'. Below these are input fields for 'INSTRUCTORID' (117), 'INSTRUCTORMNAME' (Rusan Mahajan), 'INSTRUCTORCONTACT' (1212332121), and 'INSTRUCTOREMAIL' (rusan@example.com). At the bottom are 'Insert' and 'Cancel' buttons.

INSTRUCTORID	INSTRUCTORMNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
116	Lekhnath Katuwal	9123234567	katuwal@example.com
117	Rusan Mahajan	1212332121	rusan@example.com

`script._doPostBack('FormView1$InsertButton','')`

Figure 81. Adding new Instructor



The screenshot shows the same application interface after the new instructor has been added. The 'Instructor' table now contains two rows: one for Lekhnath Katuwal and one for Rusan Mahajan. The 'New' button is visible at the bottom of the table.

INSTRUCTORID	INSTRUCTORMNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
116	Lekhnath Katuwal	9123234567	katuwal@example.com
117	Rusan Mahajan	1212332121	rusan@example.com

Figure 82. Newly Added Instructor

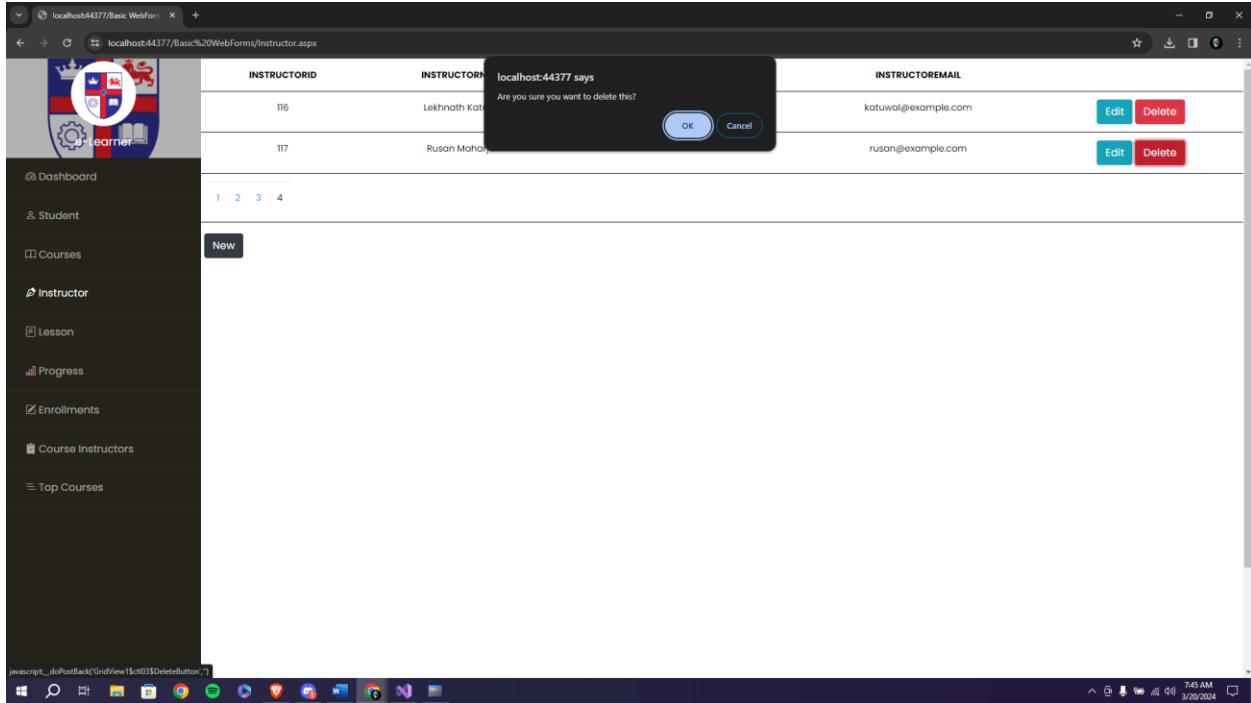


Figure 83. Deleting Instructor Data

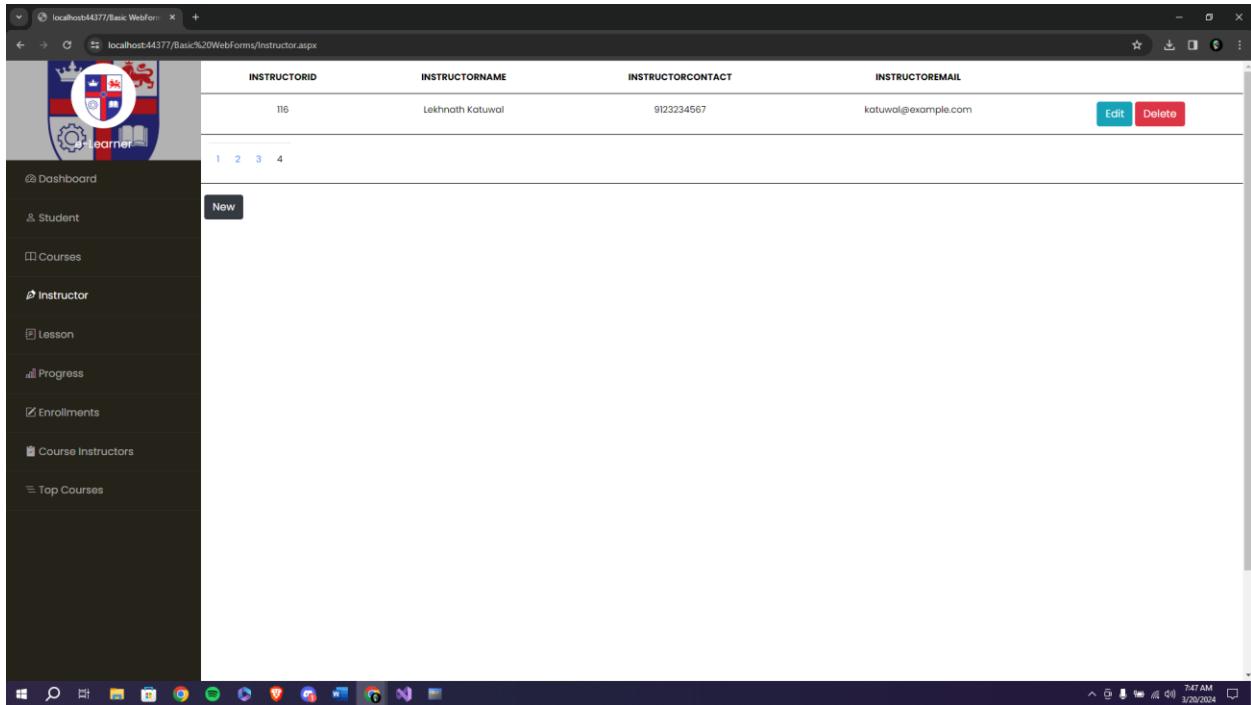
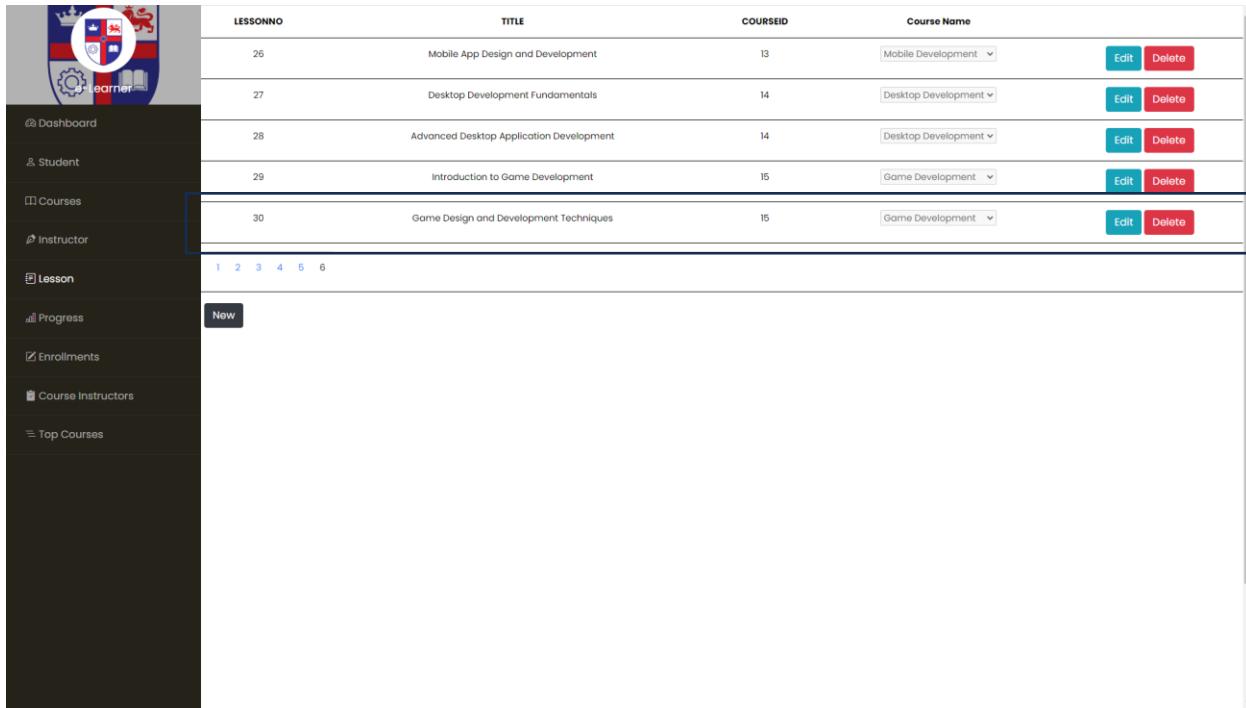


Figure 84. Deleted Instructor Data

13.1.4 Lesson Page

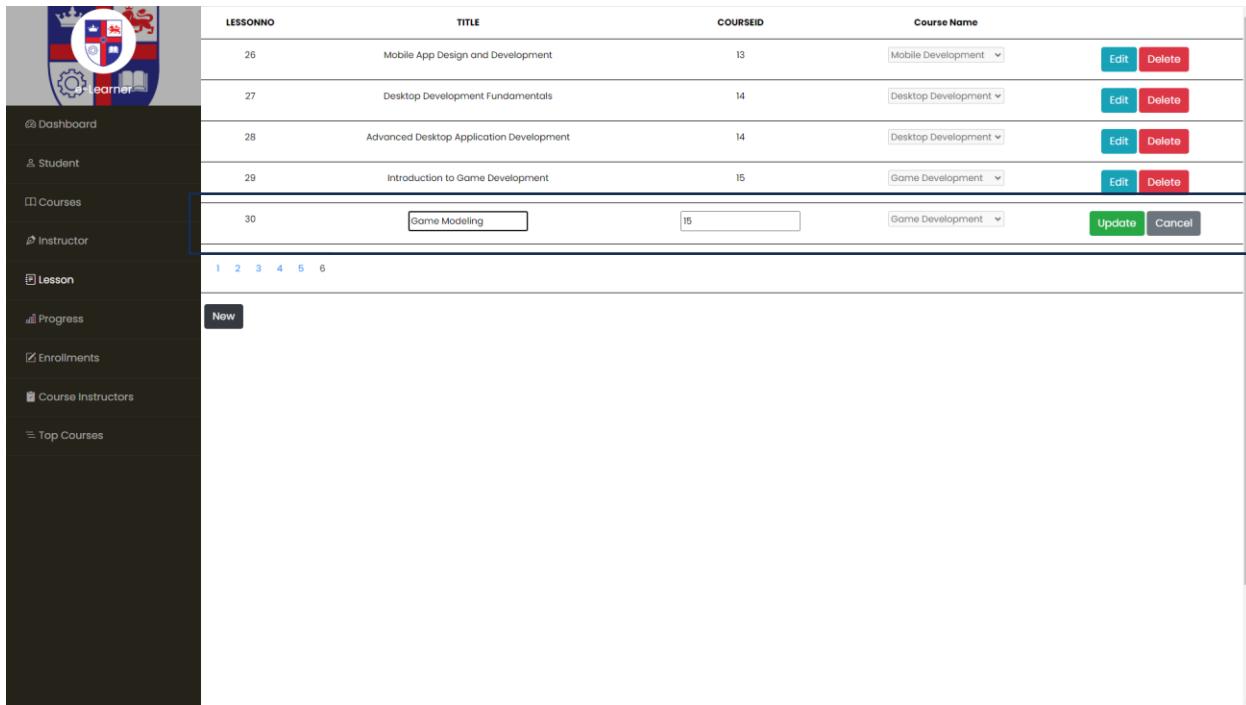


The screenshot shows a user interface for managing lessons. On the left is a sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson (which is selected), Progress, Enrollments, Course Instructors, and Top Courses. The main area displays a table of lessons:

LESSONNO	TITLE	COURSEID	Course Name	
26	Mobile App Design and Development	13	Mobile Development	Edit Delete
27	Desktop Development Fundamentals	14	Desktop Development	Edit Delete
28	Advanced Desktop Application Development	14	Desktop Development	Edit Delete
29	Introduction to Game Development	15	Game Development	Edit Delete
30	Game Design and Development Techniques	15	Game Development	Edit Delete

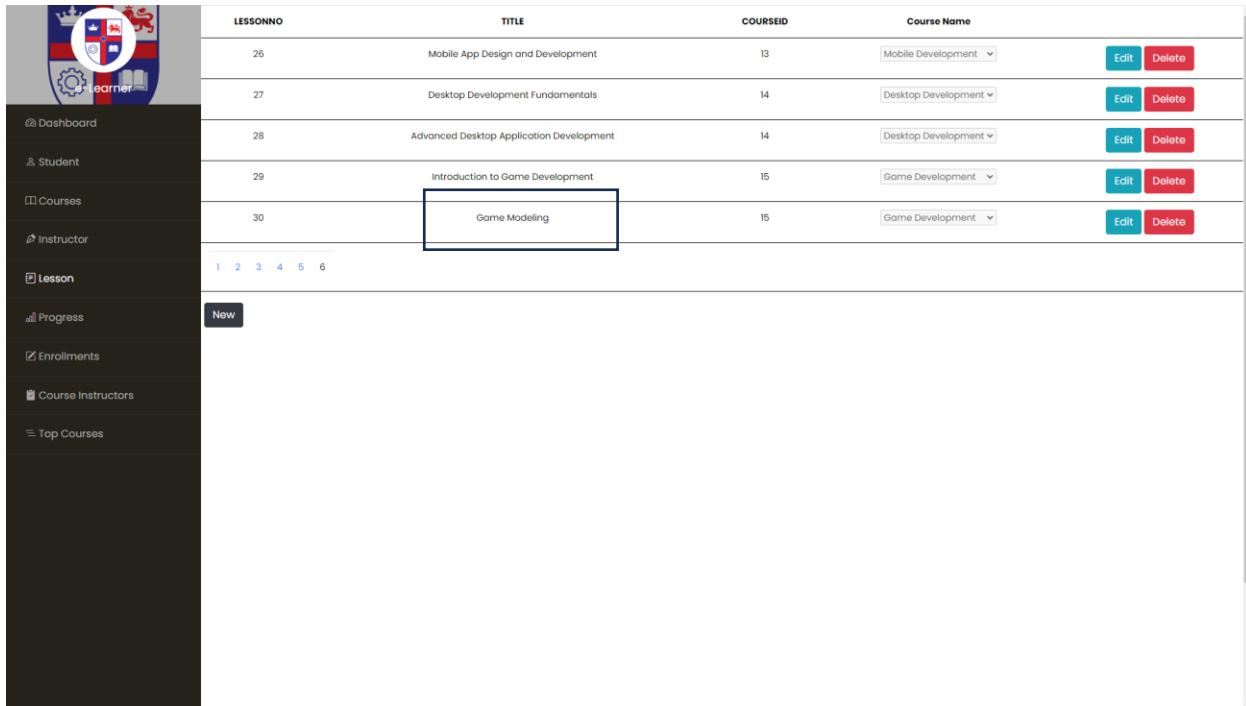
Below the table are page navigation links (1-6) and a "New" button.

Figure 85. Editing Lesson



This screenshot shows the same interface as Figure 85, but with a lesson being edited. The "Lesson No." 30 row is highlighted, and its "Title" field contains "Game Modeling". The "Course ID" field contains "15". At the bottom right of the table, there are "Update" and "Cancel" buttons.

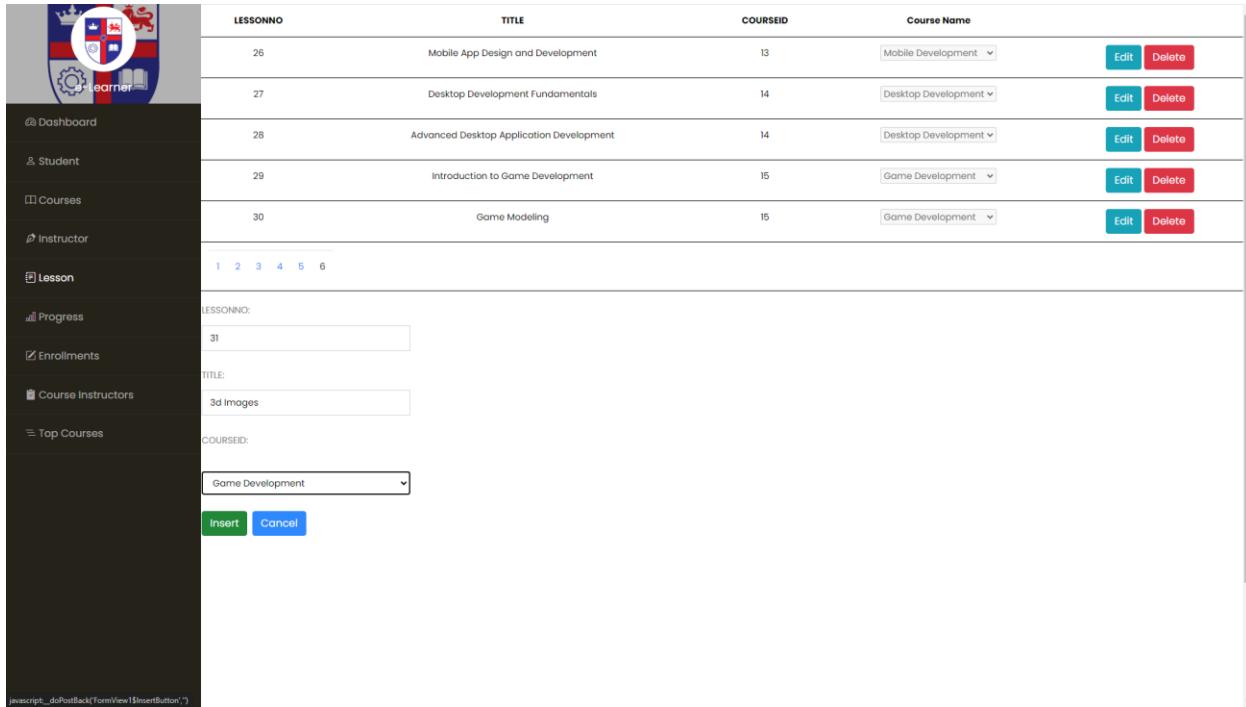
Figure 86. Lesson Update



The screenshot shows a web-based application interface for managing lessons. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson (selected), Progress, Enrollments, Course Instructors, and Top Courses. The main area has a header 'LESSONNO' and a table with columns: LESSONNO, TITLE, COURSEID, and Course Name. The table contains six rows of data. A new row is being added at the bottom, with 'LESSONNO' set to 30, 'TITLE' to 'Game Modeling', and 'COURSEID' to 15. The 'Course Name' dropdown is set to 'Game Development'. There are 'Edit' and 'Delete' buttons for each row. At the bottom of the table, there are navigation links 1, 2, 3, 4, 5, 6.

LESSONNO	TITLE	COURSEID	Course Name
26	Mobile App Design and Development	13	Mobile Development
27	Desktop Development Fundamentals	14	Desktop Development
28	Advanced Desktop Application Development	14	Desktop Development
29	Introduction to Game Development	15	Game Development
30	Game Modeling	15	Game Development

Figure 87. Updated Lesson



This screenshot shows the same application interface as Figure 87, but with a new lesson form open. The sidebar and table are identical. The new form has fields for 'LESSONNO' (set to 31), 'TITLE' (set to '3d Images'), and 'COURSEID' (set to 'Game Development'). Below these fields are two buttons: 'Insert' (green) and 'Cancel' (blue).

Figure 88. Inserting new Lesson

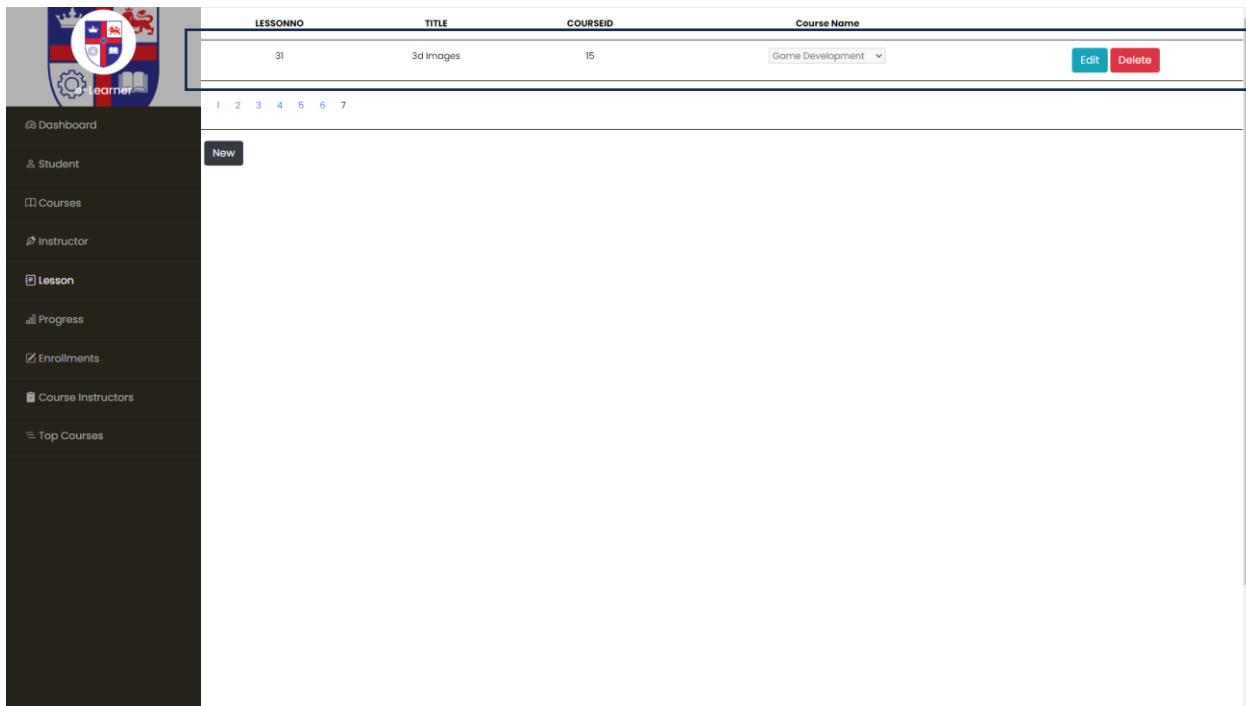


Figure 89. New lesson Inserted

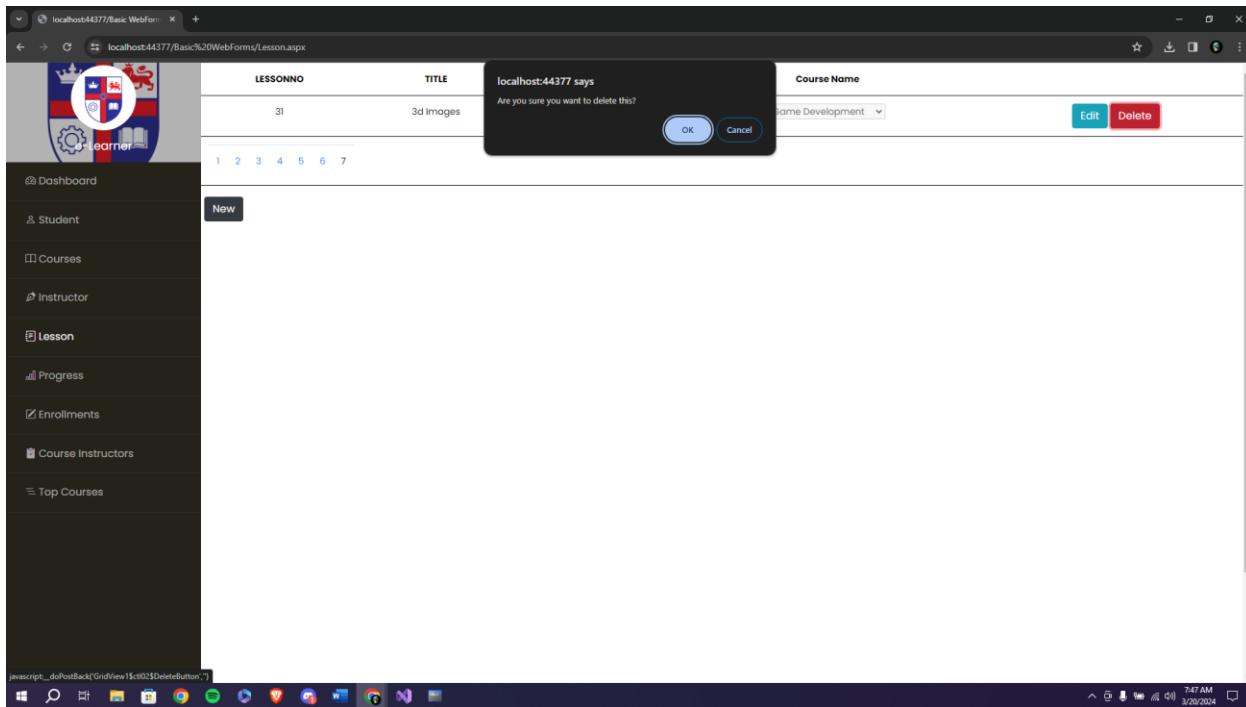
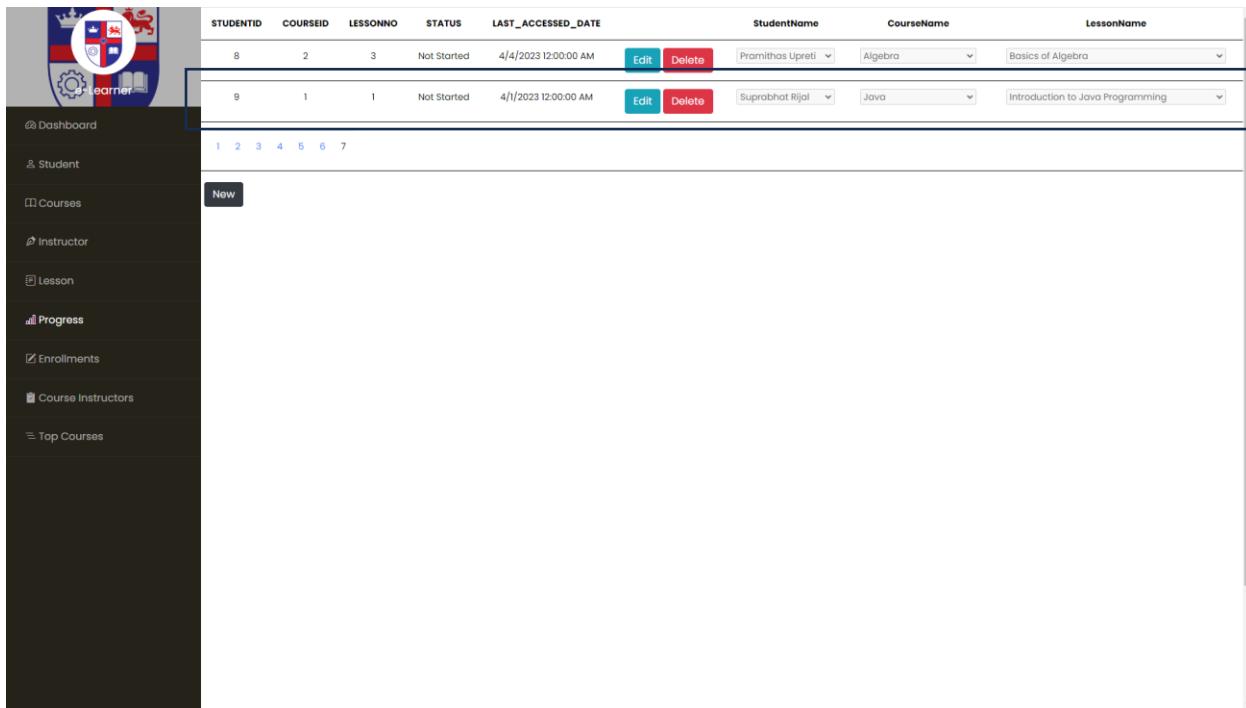


Figure 90. Deleting Lesson Data

LESSONNO	TITLE	COURSEID	Course Name
26	Mobile App Design and Development	13	Mobile Development
27	Desktop Development Fundamentals	14	Desktop Development
28	Advanced Desktop Application Development	14	Desktop Development
29	Introduction to Game Development	15	Game Development
30	Game Modeling	15	Game Development

Figure 91. Deleted Lesson Data

13.1.5 Progress Page

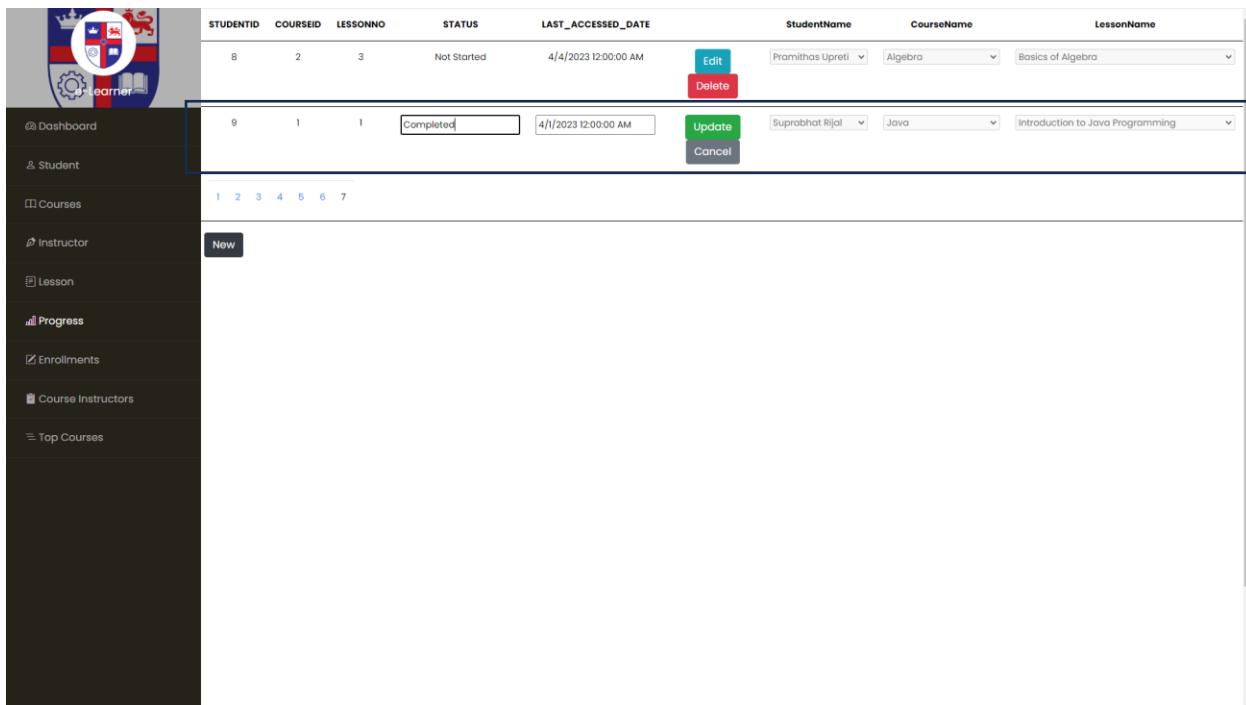


The screenshot shows a web-based application interface for managing student progress. On the left is a dark sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress (which is currently selected), Enrollments, Course Instructors, and Top Courses. The main content area displays a table of student progress data. The table has columns: STUDENTID, COURSEID, LESSONNO, STATUS, LAST_ACCEDED_DATE, StudentName, CourseName, and LessonName. Two rows are visible:

STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCEDED_DATE	StudentName	CourseName	LessonName
8	2	3	Not Started	4/4/2023 12:00:00 AM	Promithas Upadhyay	Algebra	Basics of Algebra
9	1	1	Not Started	4/1/2023 12:00:00 AM	Suprabhat Rijal	Java	Introduction to Java Programming

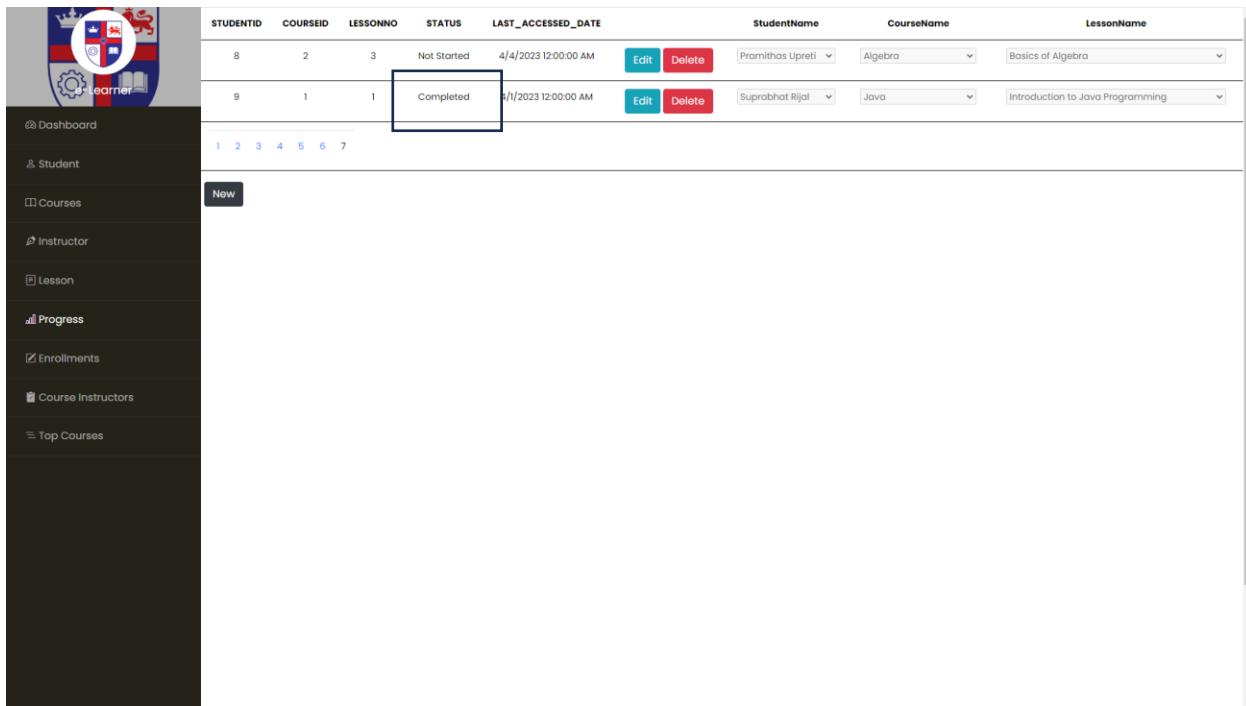
Each row includes 'Edit' and 'Delete' buttons. Below the table is a navigation bar with links 1 through 7. At the bottom of the page is a 'New' button.

Figure 92. Editing Progress



This screenshot shows the same application interface as Figure 92, but with a modal dialog box open over the second row of the progress table. The dialog is used for updating the progress status. It contains fields for STATUS (set to 'Completed'), LAST_ACCEDED_DATE (set to '4/1/2023 12:00:00 AM'), and several dropdown menus for StudentName, CourseName, and LessonName. At the bottom of the dialog are 'Update' and 'Cancel' buttons.

Figure 93. Updating Progress

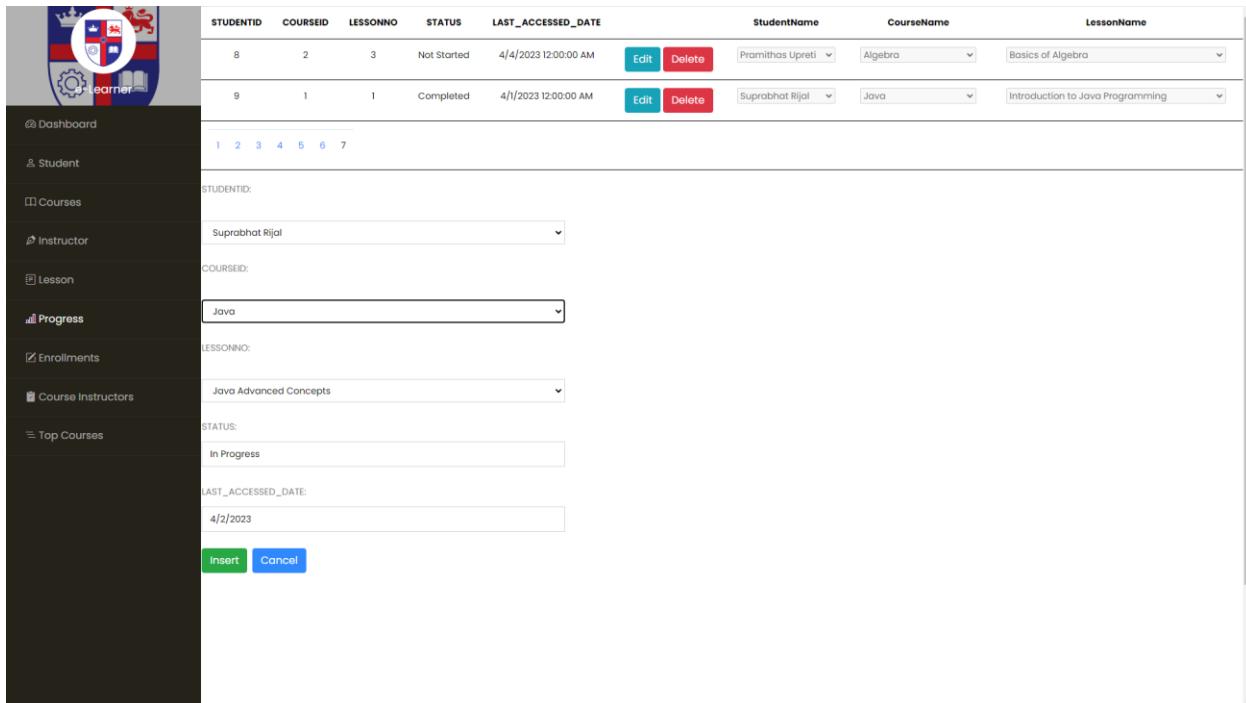


STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSED_DATE	StudentName	CourseName	LessonName
8	2	3	Not Started	4/4/2023 12:00:00 AM	Edit Delete	Pramithas Upreti	Algebra
9	1	1	Completed	4/1/2023 12:00:00 AM	Edit Delete	Suprabhat Rijal	Java

1 2 3 4 5 6 7

New

Figure 94. Updated Progress



STUDENTID: Suprabhat Rijal

COURSEID: Java

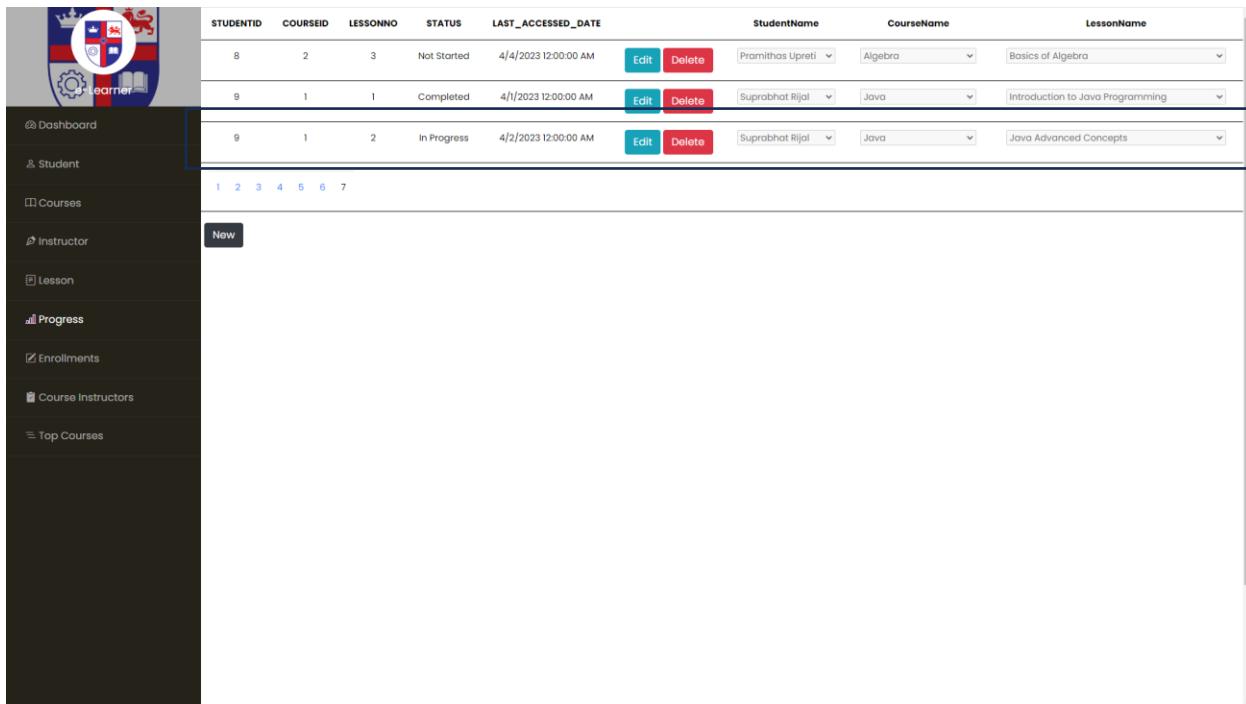
LESSONNO: Java Advanced Concepts

STATUS: In Progress

LAST_ACCESSED_DATE: 4/2/2023

[Insert](#) [Cancel](#)

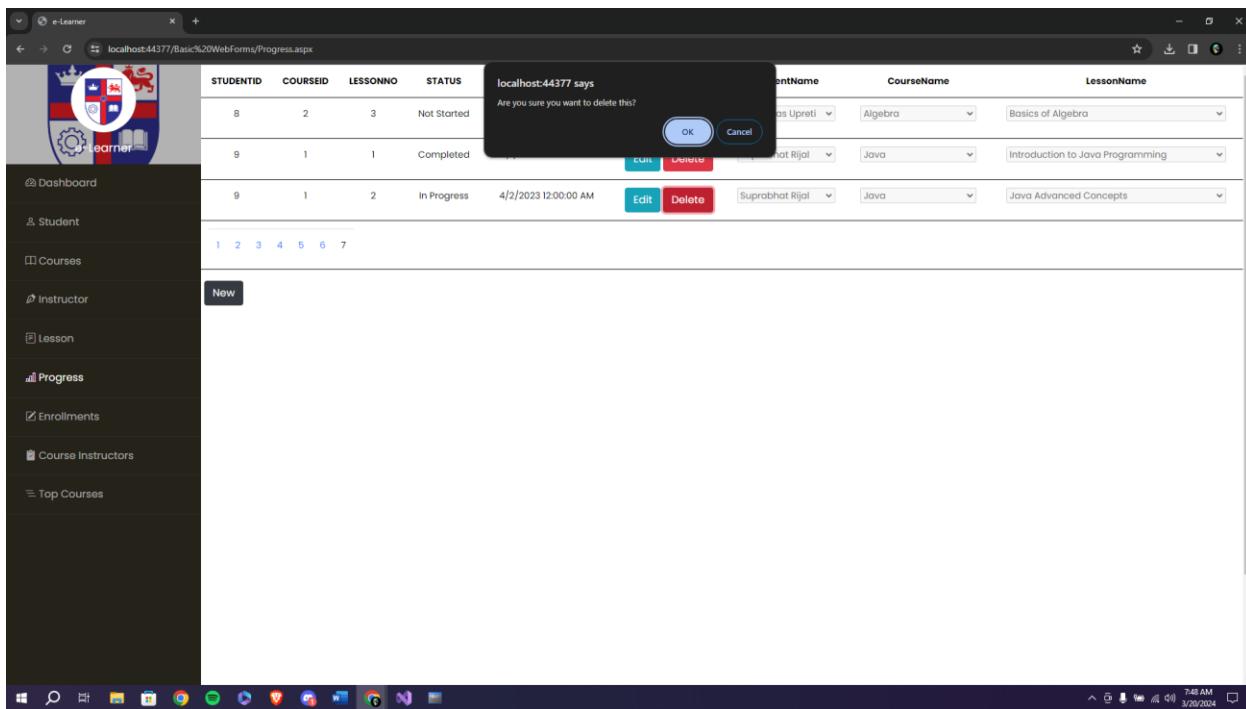
Figure 95. Adding new Progress



STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSED_DATE	StudentName	CourseName	LessonName
8	2	3	Not Started	4/4/2023 12:00:00 AM	Edit Delete	Pramithas Upreti	Algebra
8	1	1	Completed	4/1/2023 12:00:00 AM	Edit Delete	Suprabhat Rijal	Java
9	1	2	In Progress	4/2/2023 12:00:00 AM	Edit Delete	Suprabhat Rijal	Java Advanced Concepts

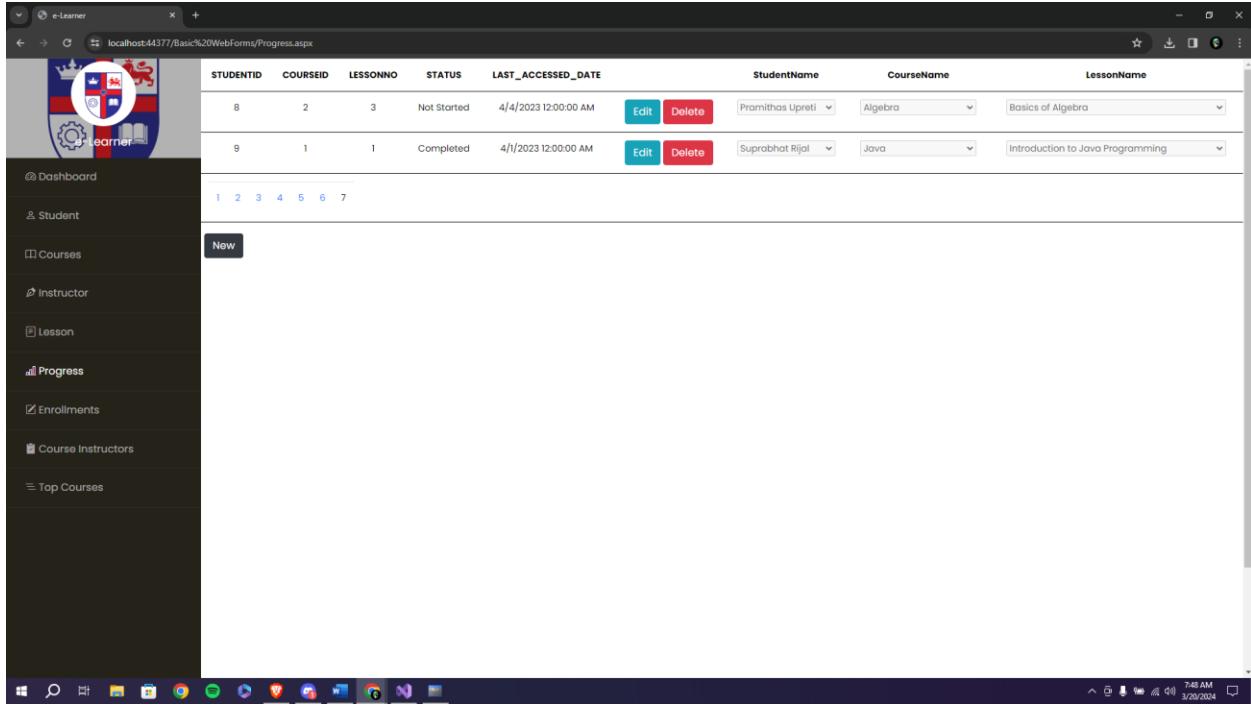
New

Figure 96. New Progress added



STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSED_DATE	StudentName	CourseName	LessonName
8	2	3	Not Started	4/4/2023 12:00:00 AM	Edit Delete	Pramithas Upreti	Algebra
8	1	1	Completed	4/1/2023 12:00:00 AM	Edit Delete	Suprabhat Rijal	Java
9	1	2	In Progress	4/2/2023 12:00:00 AM	Edit Delete	Suprabhat Rijal	Java Advanced Concepts

Figure 97. Deleting Progress data



The screenshot shows a web-based application titled "e-Learner" with a dark-themed sidebar menu. The main content area displays a table of student progress data. The table has columns: STUDENTID, COURSEID, LESSONNO, STATUS, LAST_ACCESSION_DATE, StudentName, CourseName, and LessonName. Two rows of data are visible:

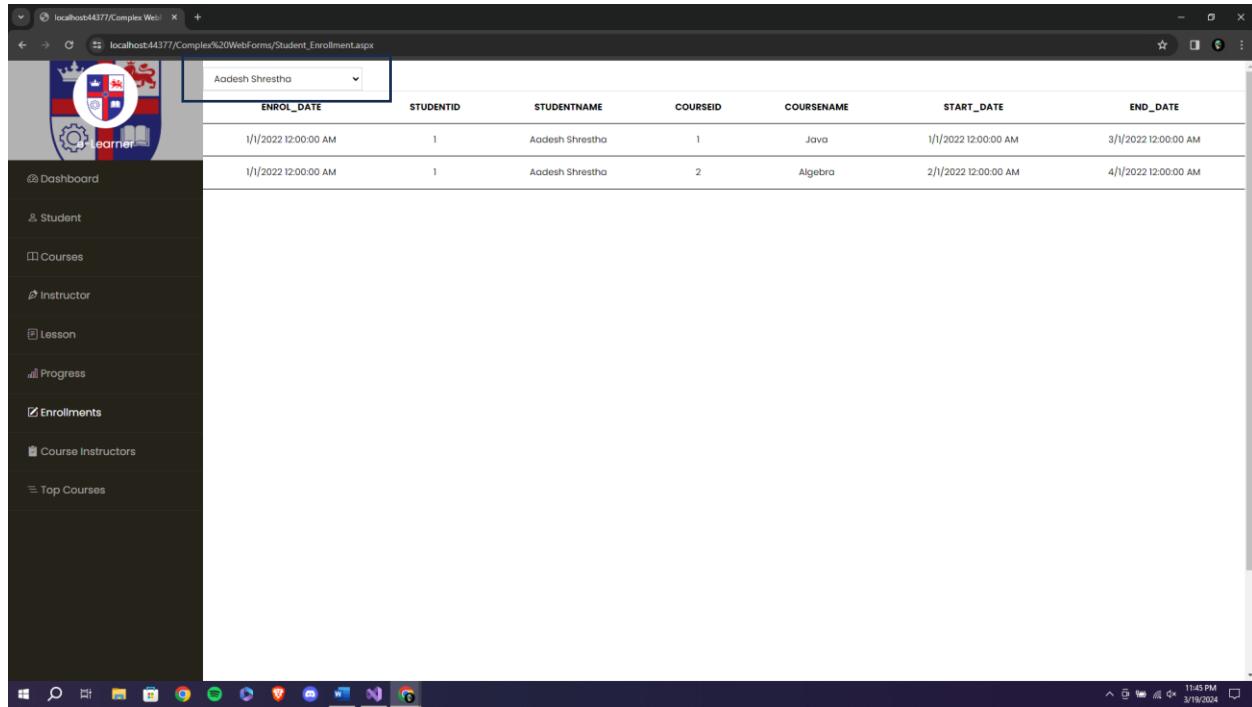
STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSION_DATE	StudentName	CourseName	LessonName
8	2	3	Not Started	4/4/2023 12:00:00 AM	Edit Delete	Pramithas Upreti	Algebra
9	1	1	Completed	4/1/2023 12:00:00 AM	Edit Delete	Suprabhat Rijol	Java

A navigation bar at the bottom of the sidebar includes links for Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. A "New" button is located below the table. The system status bar at the bottom right shows the date as 3/20/2024 and the time as 7:48 AM.

Figure 98. Deleted Progress data

13.2 Complex Forms

13.2.1 Enrollment Page



The screenshot shows a web application interface for managing student enrollments. On the left, there is a sidebar with various navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments (which is currently selected), Course Instructors, and Top Courses. The main content area displays a table of student enrollments. One row in the table is highlighted with a red circle, and a dropdown menu is open over the student's name in that row. The dropdown menu contains the student's name, "Aadesh Shrestha". The table has columns: ENROL_DATE, STUDENTID, STUDENTNAME, COURSEID, COURSENAME, START_DATE, and END_DATE. The data in the table is as follows:

ENROL_DATE	STUDENTID	STUDENTNAME	COURSEID	COURSENAME	START_DATE	END_DATE
1/1/2022 12:00:00 AM	1	Aadesh Shrestha	1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM
1/1/2022 12:00:00 AM	1	Aadesh Shrestha	2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM

Figure 99. Enrollment Dropdown

The screenshot shows a web application interface. On the left is a sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments (which is checked), Course Instructors, and Top Courses. A logo featuring a shield with a crown and a book is at the top of the sidebar.

A dropdown menu is open under the 'Student' link, listing student names: Aadesh Shrestha, Bhavesh Ronido, Aashraya Jr., Aayam Pokharel, Abhishek Yadav, and Prabhesh Basnet. 'Prabhesh Basnet' is highlighted with a blue selection bar.

The main content area displays a table of student enrollments:

STUDENTID	STUDENTNAME	COURSEID	COURSENAME	START_DATE	END_DATE
1	Aadesh Shrestha	1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM
1	Aadesh Shrestha	2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM

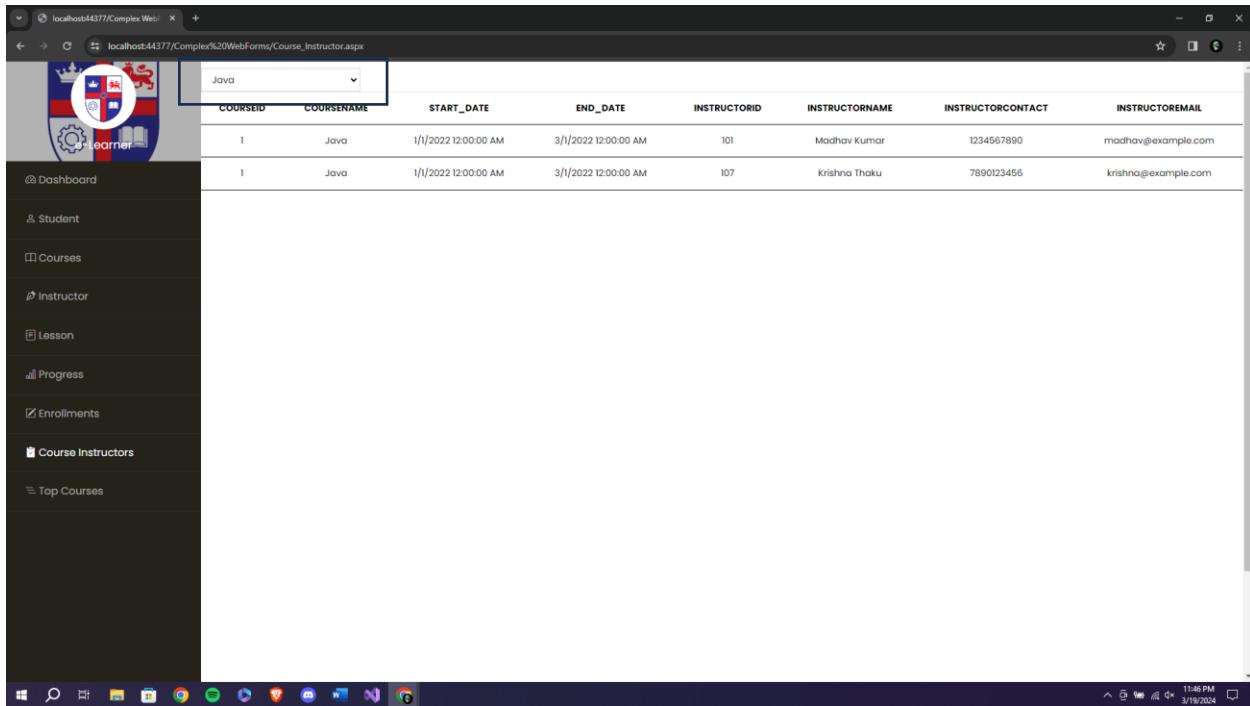
Figure 100. Dropdown List

The screenshot shows the same web application interface as Figure 100, but the dropdown menu is now closed. The main content area displays an updated table of student enrollments:

ENROL_DATE	STUDENTID	STUDENTNAME	COURSEID	COURSENAME	START_DATE	END_DATE
6/1/2022 12:00:00 AM	6	Prabhesh Basnet	1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM
6/1/2022 12:00:00 AM	6	Prabhesh Basnet	2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM
6/1/2022 12:00:00 AM	6	Prabhesh Basnet	3	Operating System	3/1/2022 12:00:00 AM	5/1/2022 12:00:00 AM
6/1/2022 12:00:00 AM	6	Prabhesh Basnet	5	Problem Solving	5/1/2022 12:00:00 AM	7/1/2022 12:00:00 AM

Figure 101. Updated Table

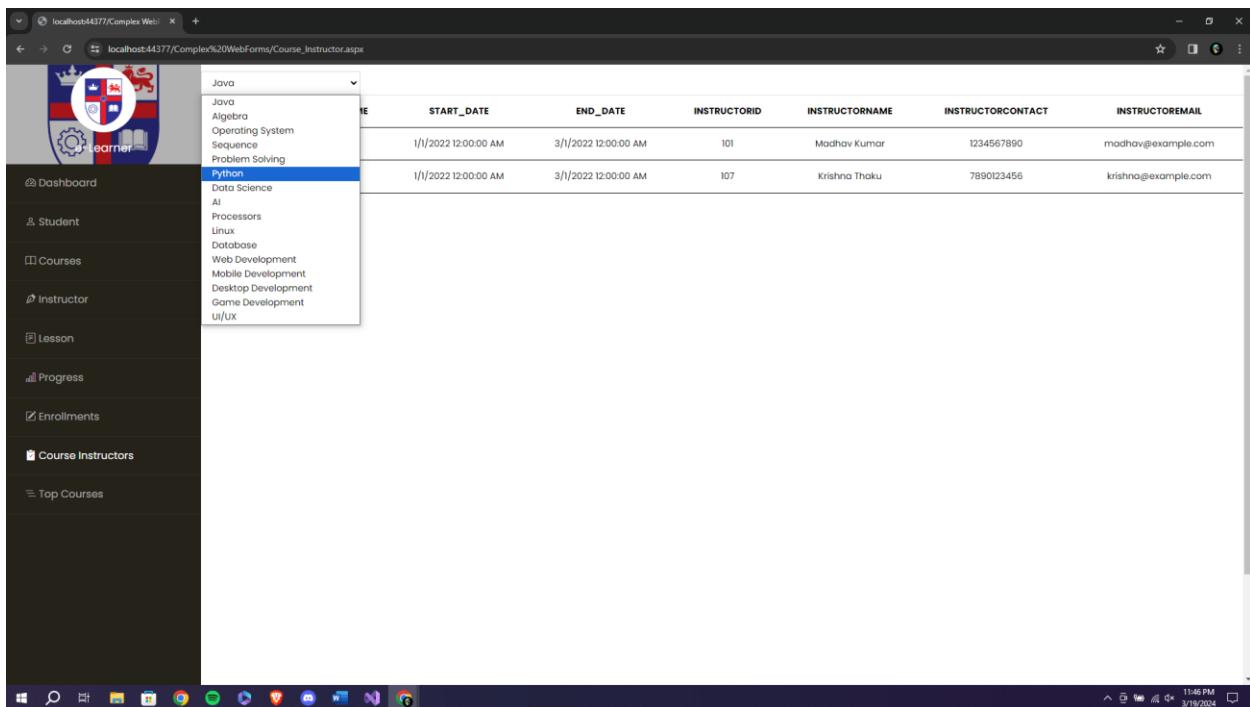
13.2.2 Course Instructors Page



A screenshot of a web browser displaying a course management system. The left sidebar has a dark theme with various navigation links. A dropdown menu is open over the course list table, specifically over the 'COURSENAME' column for the first row. The dropdown shows 'Java' at the top, followed by a list of other courses: Algebra, Operating System, Sequence, Problem Solving, Python, Data Science, AI, Processors, Linux, Database, Web Development, Mobile Development, Desktop Development, Game Development, and UI/UX.

COURSEID	COURSENAME	START_DATE	END_DATE	INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	101	Madhav Kumar	1234567890	madhav@example.com
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	107	Krishna Thaku	7890123456	krishna@example.com

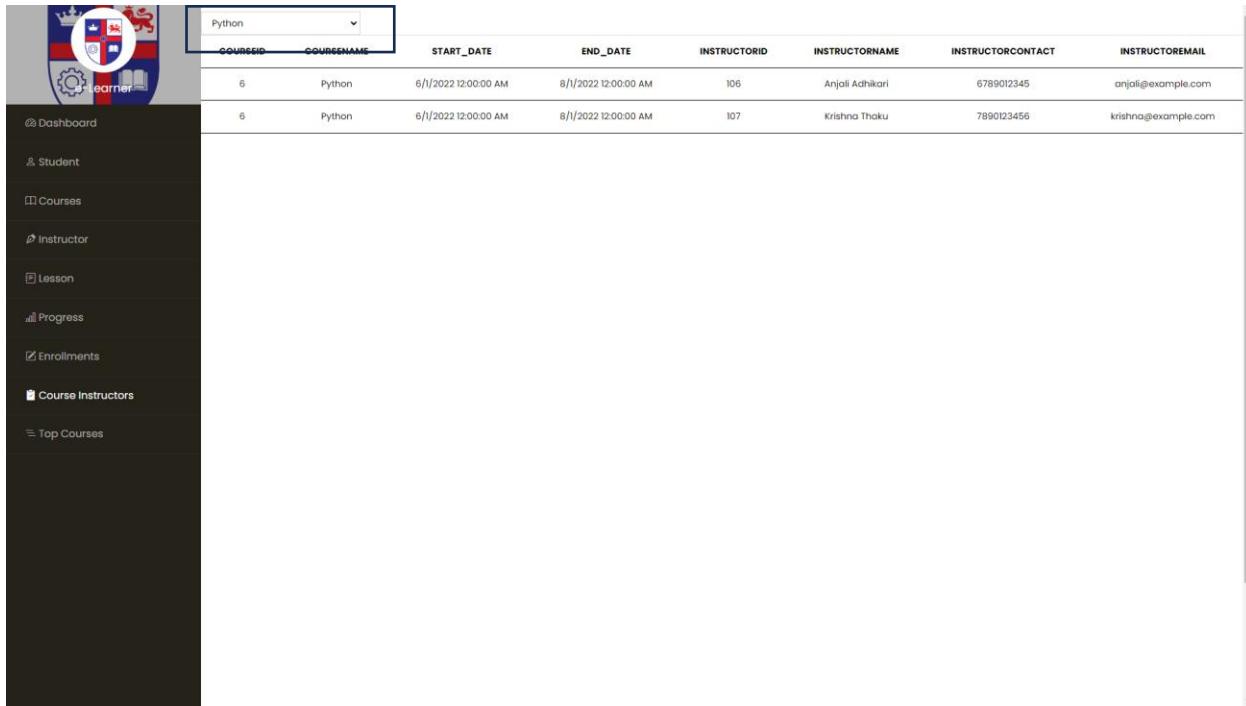
Figure 102. Course Instructor Dropdown



A screenshot of the same web browser and course management system as Figure 102. The dropdown menu is now fully expanded, showing the list of courses: Java, Algebra, Operating System, Sequence, Problem Solving, Python, Data Science, AI, Processors, Linux, Database, Web Development, Mobile Development, Desktop Development, Game Development, and UI/UX. The 'Python' option is currently selected.

COURSEID	COURSENAME	START_DATE	END_DATE	INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	101	Madhav Kumar	1234567890	madhav@example.com
1	Python	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM	107	Krishna Thaku	7890123456	krishna@example.com

Figure 103. Course Instructor Dropdown list



The screenshot shows a web-based application interface. On the left is a dark sidebar menu with the following items:

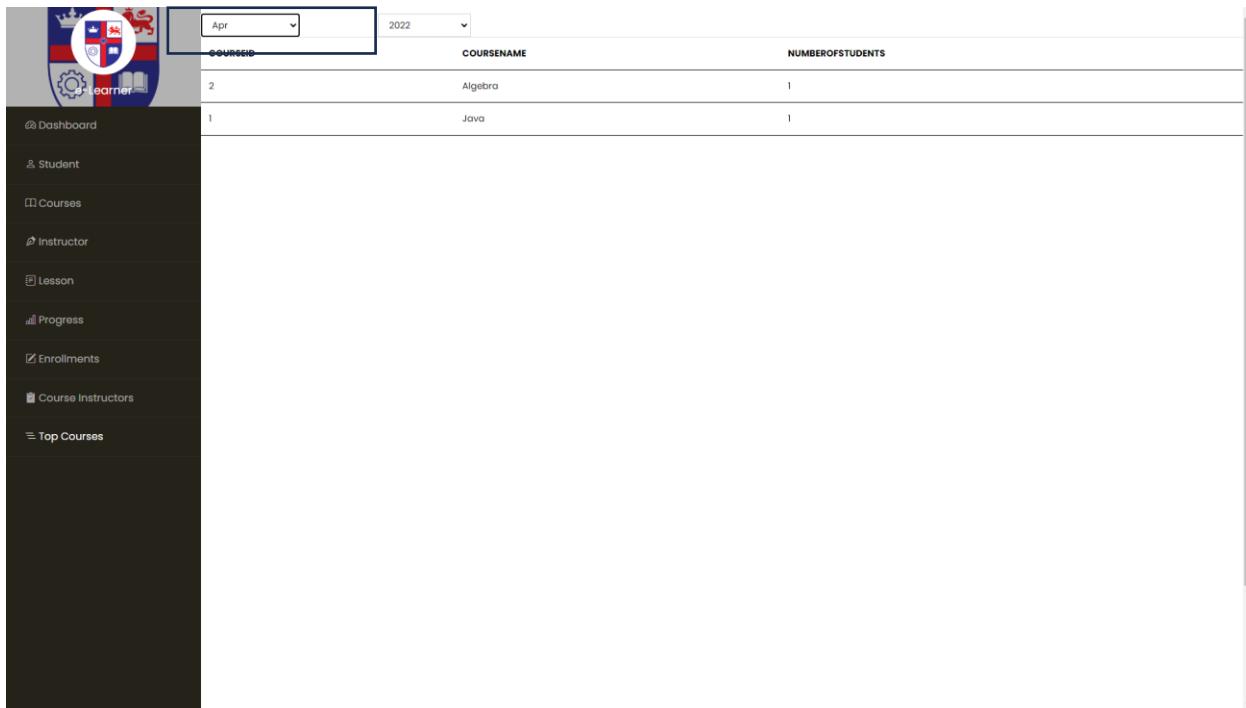
- Dashboard
- Student
- Courses
- Instructor
- Lesson
- Progress
- Enrollments
- Course Instructors** (highlighted with a blue background)
- Top Courses

The main content area has a header "Python" with a dropdown arrow. Below it is a table with the following data:

COURSEID	COURSENNAME	START_DATE	END_DATE	INSTRUCTORID	INSTRUCTORNAME	INSTRUCTORCONTACT	INSTRUCTOREMAIL
6	Python	6/1/2022 12:00:00 AM	8/1/2022 12:00:00 AM	106	Anjali Adhikari	6789012345	anjali@example.com
6	Python	6/1/2022 12:00:00 AM	8/1/2022 12:00:00 AM	107	Krishna Thaku	7890123456	krishna@example.com

Figure 104. New view table

13.2.3 Top Courses Page



The screenshot shows a user interface for managing courses. On the left is a sidebar with various navigation options: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The 'Top Courses' option is currently selected, indicated by a grey background. At the top of the main content area, there is a dropdown menu for selecting a month, with 'Apr' highlighted. Below this, there is a table with three columns: COURSEID, COURSENAME, and NUMBEROFSTUDENTS. The table contains two rows: one for 'Algebra' with 1 student and another for 'Java' with 1 student.

COURSEID	COURSENAME	NUMBEROFSTUDENTS
2	Algebra	1
1	Java	1

Figure 105. Top Course Dropdown menu

The screenshot shows a user interface for a learning management system. On the left is a dark sidebar with various navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The 'Top Courses' link is highlighted with a blue icon. The main content area has two dropdown menus at the top: one for the month ('Apr' to 'May') and one for the year ('2022'). Below these is a table titled 'COURSENAME' with a single row: 'Algebra' and 'NUMBEROFSTUDENTS' both set to '1'. A vertical scroll bar is visible on the right side of the main content area.

Figure 106. Top Courses Dropdown List

This screenshot shows the same application interface as Figure 106, but with a different table structure. The table now includes three columns: 'COURSEID' (with values 2, 1, and 3), 'COURSENAME' (with values 'Algebra', 'Java', and 'Operating System'), and 'NUMBEROFSTUDENTS' (all set to '1'). The rest of the interface, including the sidebar and dropdown menus, remains identical to Figure 106.

Figure 107. New table view for Top Courses

13.3 Test Cases

13.3.1 Case 1

Test	To view id for students with same name
Steps	Click on dropdown menu and select similar names
Expected Outcome	Display students data for Similar Names with their id
Actual Outcome	Display student names only
Result	Failure

Table 12. Testing Failure 1

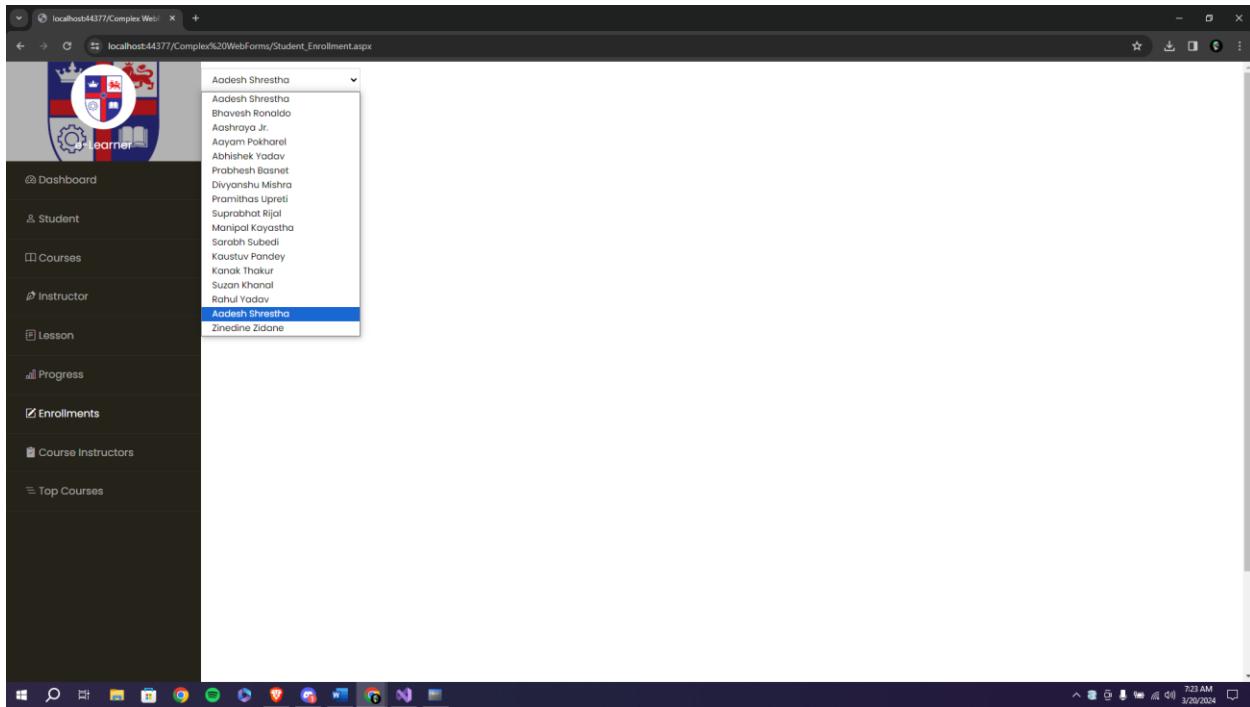


Figure 108. Student with the Same Name

The screenshot shows a web application interface for managing student enrollments. On the left, there is a sidebar with a logo and navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments (which is selected), Course Instructors, and Top Courses. A dropdown menu under 'Enrollments' shows a list of student names, with 'Aadesh Shrestha' being the currently selected item. The main content area displays a table of student enrollment records:

STUDENTID	STUDENTNAME	COURSEID	COURSENAME	START_DATE	END_DATE
1	Aadesh Shrestha	1	Java	1/1/2022 12:00:00 AM	3/1/2022 12:00:00 AM
1	Aadesh Shrestha	2	Algebra	2/1/2022 12:00:00 AM	4/1/2022 12:00:00 AM

Figure 109. Displaying ID with the Student Name

13.3.2 Case 2

Test	Check Foreign Key
Steps	Insert into Form course number
Expected Outcome	Data is inserted
Actual Outcome	Parent Key not found error
Result	Failure

Table 13. Testing Failure 2

The screenshot shows a web application interface titled "e-Learner". On the left, there is a sidebar with navigation links: "Student", "Courses", "Instructor", "Lesson", "Progress", "Enrollments", "Course Instructors", and "Top Courses". The main content area displays a table of progress data:

STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSED_DATE
Aadesh Shrestha	3	1	Not Started	3/7/2023 12:00:00 AM
Aadesh Shrestha	3	4	Not Started	3/8/2023 12:00:00 AM
Bhavesh Ronaldo	1	1	Not Started	3/9/2023 12:00:00 AM

Below the table, there is a form for inserting new data:

STUDENTID: Aadesh Shrestha

COURSEID: 3

LESSONNO: Introduction to Java Programming

STATUS: Complete

LAST_ACCESSED_DATE: 2/12/2022

Buttons: Insert, Cancel

Figure 110. Inserting into Progress

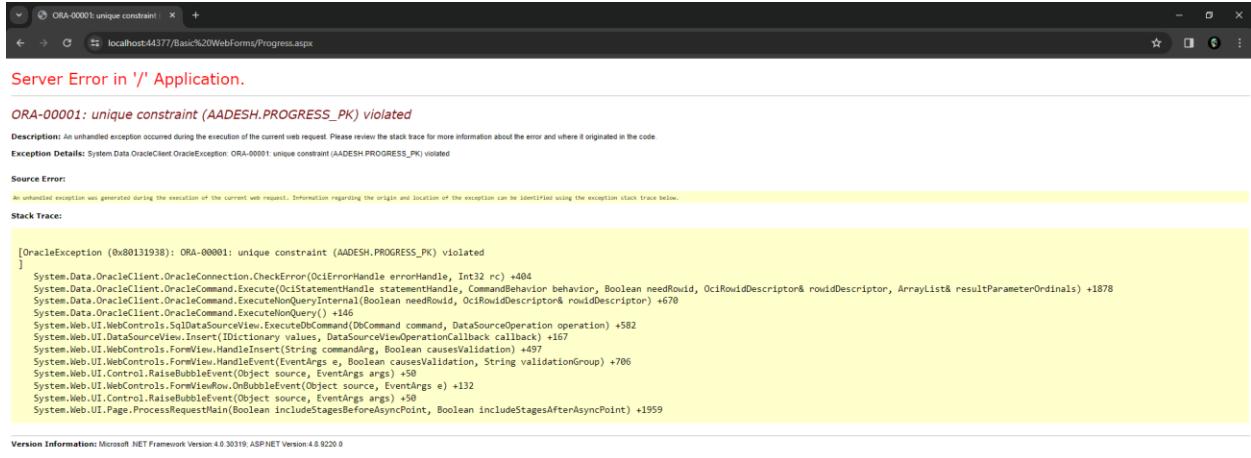


Figure 111. Parent Key Not Found



STUDENTID:

Aadesh Shrestha

COURSEID:

Java

LESSONNO:

Introduction to Java Programming

STATUS:

In Progress

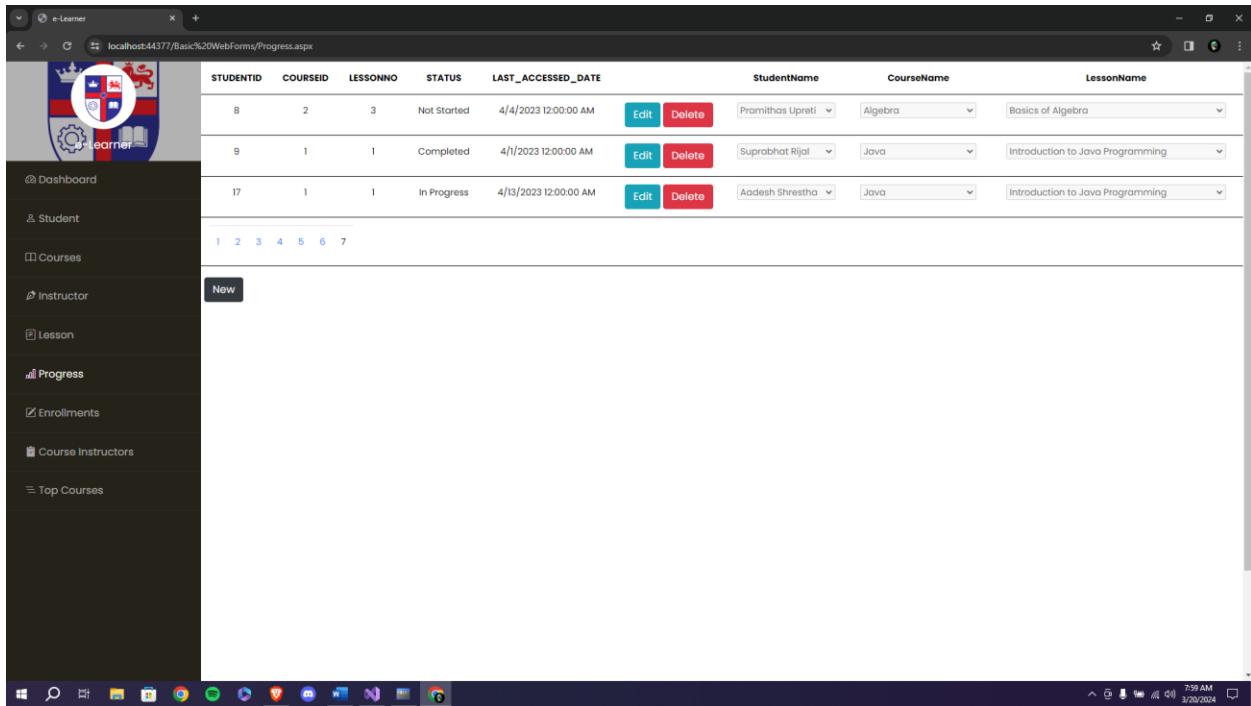
LAST_ACCESSED_DATE:

4/15/2024 1:00:00 AM

Insert

Cancel

Figure 112. Dropdown to show parent key



The screenshot shows a web-based application titled "e-Learner" with a dark theme. On the left is a sidebar with navigation links: Dashboard, Student, Courses, Instructor, Lesson, Progress, Enrollments, Course Instructors, and Top Courses. The main content area displays a table titled "Progress" with the following data:

STUDENTID	COURSEID	LESSONNO	STATUS	LAST_ACCESSION_DATE	StudentName	CourseName	LessonName	
8	2	3	Not Started	4/4/2023 12:00:00 AM	Edit Delete	Pramithas Upreti	Algebra	Basics of Algebra
9	1	1	Completed	4/1/2023 12:00:00 AM	Edit Delete	Suprabhat Rijal	Java	Introduction to Java Programming
17	1	1	In Progress	4/13/2023 12:00:00 AM	Edit Delete	Aadesh Shrestha	Java	Introduction to Java Programming

Below the table, there is a navigation bar with links 1, 2, 3, 4, 5, 6, 7. A "New" button is located at the bottom right of the table area. The status bar at the bottom right shows the time as 7:59 AM and the date as 3/20/2024.

Figure 113. Successful Insertion

14 Further Discussion

The project was a great opportunity to learn new technologies. I used Visual Studio 2022, Oracle SQL Developer, and Modeler, along with ASP.NET Web Application and Web Forms to create a management system. These tools helped me understand how to build websites and applications. Programmers can construct websites, web apps, online services, and mobile applications using Visual Studio, a development environment. It's a strong tool with excellent editing and debugging capabilities. (Murali, 2023)

The relational database management system (RDBMS) known as Oracle Database is produced by Oracle Corporation. The features, background, and editions of the Oracle database will all be thoroughly explained in this article. We must first understand the database before moving on to the oracle. With Oracle SQL Developer and Modeler, I learned how to work with advanced databases. I could create databases, manage users, and add and get information. This experience taught me about database design and how to use SQL to get the data I need. (Javatpoint, n.d.)

Using ASP.NET Web Forms made it easy to create the website's interface. I could focus more on making the website work well, rather than spending too much time on the design. Visual Studio's debugging tools were also helpful in finding and fixing issues in the code. Overall, this project has given me a good foundation in web development that I can use for future projects.

15 Conclusion

The project was a valuable learning experience that provided knowledge to a range of technologies essential for web development. Through working with Visual Studio 2022, Oracle SQL Developer, Modeler, ASP.NET Web Application, and Web Forms, I gained practical insights into database management, SQL querying, and web application development. These skills are not only applicable to this project but also transferable to future endeavors in the field of software development.

Additionally, the project helped me understand the importance of a well-structured database and efficient querying methods. It also highlighted the significance of user-friendly interfaces, which can be achieved through ASP.NET Web Forms. Overall, this project has equipped me with a solid foundation in web development, enabling me to create functional and visually appealing web applications.

16 References

Javatpoint, n.d. *What is Oracle.* [Online]

Available at: <https://www.javatpoint.com/what-is-oracle>

Murali, M., 2023. *What is Visual Studio? All you need to know about Visual Studio.* [Online]

Available at: <https://blog.hubspot.com/website/what-is-visual-studio>