Assignment Title:

Coursework

Coursework Type:

Individual

Module Name:

<ST4005CEM Database System>

Intake: April

Submitted By:

CU ID:14811405

College ID:230330

Name: Prajwal Chaulagain

Submitted to:

Ayush Kaji Dangol

Table of Contents

Abstract

The goal of this project is to construct a database system with an emphasis on data visualization, ER diagrams, and normalization. By getting rid of redundant information, normalization enhances data integrity. Database design is made easier with the use of ER diagrams, which graphically map relationships between data elements. Patterns and trends can be found in data visualization by using Python tools such as Matplotlib and Seaborn. These elements work together to provide a strong foundation for efficient data handling and analysis.

Acknowledge

My heartfelt gratitude goes out to my students for their passion and commitment, which were invaluable to our effort. I am also grateful to my coworkers for their insightful criticism and encouragement. I also thank MySQL Workbench, Visual Paradigm, and Python modules like Pandas and Matplotlib for their invaluable use in the development of this project.

Introduction

Effective data management is crucial in today's world. I'm working on a database system with three primary components for my assignment: data visualization, ER diagrams, and normalization. Normalization ensures that the data is accurate and helps to clean it up by eliminating duplicates. The ER diagram illustrates how the related data aid several database design components. I'll utilize Matplotlib and other Python utilities.
I'll be able to identify patterns and trends in the data by using Seaborn to make charts and graphs. By concentrating on these areas ER diagrams, data visualization, and normalization. I hope to create a robust and effective framework that facilitates data usage and understanding.

Normalization

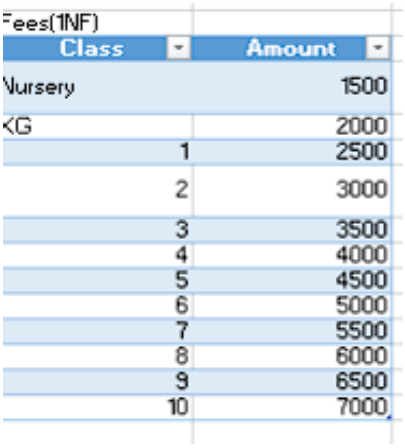
Normalization is the process of organizing a database to reduce redundancy and improve data integrity. Normalization also simplifies the database design to achieve the optimal structure composed of atomic elements (i.e. elements that cannot be broken down into smaller parts). (Ian, 2017)

It has three different parts:

1) First Normal Form(1NF)
2) Second Normal Form(2NF)
3) Third Normal Form(3NF)

1NF

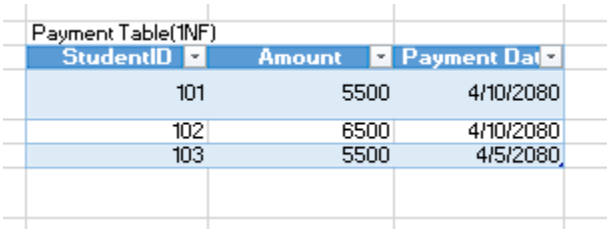
Figure 1Fees Table



Fees(1NF)

Class	Amount
Nursery	1500
KG	2000
1	2500
2	3000
3	3500
4	4000
5	4500
6	5000
7	5500
8	6000
9	6500
10	7000

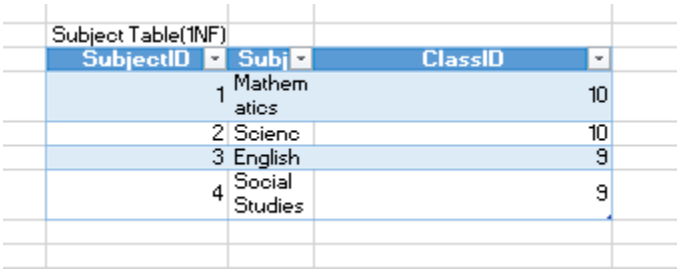
Figure 2 Payment Table



Payment Table(1NF)

StudentID	Amount	Payment Dat
101	5500	4/10/2080
102	6500	4/10/2080
103	5500	4/5/2080

Figure 3 Subject Table



Subject Table(1NF)

SubjectID	Subj	ClassID
1	Mathematics	10
2	Scienc	10
3	English	9
4	Social Studies	9

Figure 4 Subject Table

Subject Table(1NF)

SubjectID	Subj	ClassID
1	Mathem atics	10
2	Scienc	10
3	English	9
4	Social Studies	9

Figure 5 Teachers Table

Teachers Table (1NF)

TeacherID	Teacher Na	Date of Bi	Gende	Address	Phone	Salary
1	Ashok Aryal	9/5/1980	Male	Tansen-8, Palpa, Lumbini	9867389098	40000
2	Basanta Poudel	4/25/1975	Male	Bhairahawa-1, Rupandehi, Lumbini	9876787887	45000
3	Lekhnath Panta	1/28/1985	Male	Butwal-12, Rupandehi, Lumbini	9856788993	60000
4	Roman Shrestha	6/2/1990	Male	Butwal-12, Rupandehi, Lumbini	9843678567	55000
5	Ritika Banjade	5/11/1992	Female	Tansen-8, Palpa, Lumbini	9841678939	50000

Figure 6 Attendance Table

Attendance Table(1NF)

AttendanceID	Student Nam	Date	Status
1	Rajesh Acharya	7/1/2023	Present
2	Sita Sharma	7/2/2023	Absent
3	Asha Rai	7/3/2023	Present
4	Rajesh Rai	7/1/2023	Present
5	Sita Sharma	7/2/2023	Present
6	Asha Sharma	7/3/2023	Absent

Figure 7 Result Table

Result of Exam Table(1NF)

ResultID	Student Nam	Subject	Date of Exa	Score
1	Rajesh Acharya	Mathematics	6/15/2023	85
2	Sita Sharma	Science	6/20/2023	78
3	Asha Rai	Nepali	6/25/2023	92
4	Sita Sharma	Sanskrit	6/15/2023	78
5	Asha Sharma	English	6/20/2023	90
6	Asha Sharma	Mathematics	6/25/2023	65

Figure 8 Event Table

Events Table(1NF)

EventID	Event Name	Date	Location
1	Annual Day	8/10/2023	School
2	Sports Day	9/20/2023	Sports
3	Science Fair	10/15/2023	Science Lab
4	Parent-Teacher Meeting	7/30/2023	School Auditorium

Figure 9 Library Record Table

Library Records Table(1NF)

RecordID	Student Nam	Book Titl	Author	Issue Date	Return Date
1	Asha Rai	Java the complete reference (12th edition)	Herbert Schildt	7/5/2023	7/20/2023
2	Sita Sharma	Spidering HACKS	Tara Calishain	6/30/2023	7/10/2023
3	Asha Sharma	Windows System Programming	Johnson M. Hart	7/10/2023	7/25/2023
4	Asha Sharma	C# in Depth	Jon Skeet	7/15/2023	7/25/2023

2NF

The second Normal Form (2NF) ensures non-prime attributes fully depend on the entire composite key, eliminating partial dependencies. Single-attribute key tables are automatically in 2NF.

Figure 10 Fees Table

Fees Table(2NF)

ClassID	Class	Amount
1	Nursery	1500
2	KG	2000
3	1	2500
4	2	3000
5	3	3500
6	4	4000
7	5	4500
8	6	5000
9	7	5500
10	8	6000
11	9	6500
12	10	7000

Figure 11 Payment Table

Payment Table(2NF)

PaymentID	StudentID	Amount	PaymentDat
1	101	5500	4/10/2080
2	102	6500	4/10/2080
3	103	5500	4/5/2080

Figure 12 Subject Table

Subject Table(2NF)

SubjectID	Subject	Clas
1	Mathematics	10
2	Science	10
3	English	9
4	Social Studies	9

Figure 13 Teacher Table

Teacher Table(2NF)

TeacherID	Teacher Nam	Date of	Gender	Address	Phone	Salary
1	Ashok Aryal	9/5/1980	Male	Tansen-8, Palpa, Lumbini	9867389038	40000
2	Basanta Poudel	4/25/1975	Male	Bhairahawa-1, Rupandehi,	9876787887	45000
3	Lekhnath Panta	1/28/1985	Male	Butwal-12, Rupandehi,	9856788393	60000
4	Roman Shrestha	6/2/1990	Male	Butwal-12, Rupandehi,	9843678567	55000
5	Ritika Banjade	5/11/1992	Female	Tansen-8, Palpa, Lumbini	9841678939	50000

Figure 14 Student Table

Student Table(2NF)

StudentID	Student Nam	Date of	Gender	Address
1	Ramesh Acharya	3/15/2005	Male	Butwal-10, Rupandehi,
2	Sita Sharma	6/20/2006	Female	Taulihawa-5, Kapilvastu,
3	Asha Rai	11/10/2004	Female	Siddharth nagar-12, Lumbini

Figure 15 Attendance Table

Attendance Table(2NF)

AttendanceID	StudentID	Date	Status
1	1	7/1/2023	Present
2	2	7/2/2023	Absent
3	3	7/3/2023	Present
4	4	7/1/2023	Present
5	5	7/2/2023	Present
6	6	7/3/2023	Absent

Figure 16 Result of Exam Table

Result of Exam Table(2NF)

ResultID	StudentID	SubjectID	Date of Exam	Score
1	1	1	6/15/2023	85
2	2	2	6/20/2023	78
3	3	3	6/25/2023	92
4	2	4	6/15/2023	78
5	3	5	6/20/2023	90
6	3	6	6/25/2023	65

Figure 17 Event Table

Event Table(2NF)

EventID	Event Name	Date	Location
1	Annual Day	8/10/2023	School Grounds
2	Sports Day	9/20/2023	Sports
3	Science Fair	10/15/2023	Science Lab
4	Parent-Teacher Meeting	7/30/2023	School Auditorium

Figure 18 Library Table

Library Record Table(2NF)

RecordID	StudentID	Book Title	Author	Issue Date	Return Date
1	1	Java the complete reference (12th	Herbert Schildt	7/5/2023	7/20/2023
2	2	Spidering	Tara Calishain	6/30/2023	7/10/2023
3	3	Windows System Programming	Johnson M. Hart	7/10/2023	7/25/2023
4	4	C# in Depth	Jon Skeet	7/15/2023	7/25/2023

3NF

There are no transitive dependencies and 2NF appears in the Three Normal Forms; all non-key attributes depends only on the primary key.

Figure 19 Fees Table

Fees Table(3NF)

ClassID	Class	Amount
1	Nursery	1500
2	KG	2000
3	1	2500
4	2	3000
5	3	3500
6	4	4000
7	5	4500
8	6	5000
9	7	5500
10	8	6000
11	9	6500
12	10	7000

Figure 20 Payment Table

Payment Table(3NF)

PaymentID	StudentID	Amount	PaymentDat
1	101	5500	4/10/2080
2	102	6500	4/10/2080
3	103	5500	4/5/2080

Figure 21 Subject Table

Subject Table(3NF)

SubjectID	Subject	ClassID
1	Mathematics	10
2	Science	10
3	English	9
4	Social Studies	9

Figure 22 Teacher Table

Teacher Table(3NF)

TeacherID	Teacher Nam	Date of Birt	Gender	Address	Phone	Salary
1	Ashok Aryal	9/5/1980	Male	Tansen-8, Palpa, Lumbini	9867389038	40000
2	Basanta Poudel	4/25/1975	Male	Bhairahawa-1, Rupandehi, Lumbini	9876787887	45000
3	Lekhnath Panta	1/28/1985	Male	Butwal-12, Rupandehi, Lumbini	9856788993	60000
4	Roman Shrestha	6/2/1990	Male	Butwal-12, Rupandehi, Lumbini	9843678567	55000
5	Ritika Banjade	5/11/1992	Female	Tansen-8, Palpa, Lumbini	9841678939	50000

Figure 23 Student Table

Student Table(3NF)

StudentID	Student Nam	Date of	Gender	Address
1	Ramesh Acharya	3/15/2005	Male	Butwal-10, Rupandehi,
2	Sita Sharma	6/20/2006	Female	Taulihawa-5, Kapilvastu,
3	Asha Rai	11/10/2004	Female	Siddharth nagar-12, Lumbini

Figure 24 Guardian Table

Guardian Table

GuardianID	StudentID	Guardian Nan	Guardian Phon
1	1	Gita Acharya	9847011111
2	2	Ram Sharma	9816412121
3	3	Hari Rai	9867066700

Figure 25 Attendance Table

Attendance Table(3NF)

AttendanceID	StudentID	Date	Status
1	1	7/1/2023	Present
2	2	7/2/2023	Absent
3	3	7/3/2023	Present
4	4	7/1/2023	Present
5	5	7/2/2023	Present
6	6	7/3/2023	Absent

Figure 26 Result of Exam Table

Result of Exam Table(3NF)

ResultID	StudentID	SubjectID	Date of Exam	Score
1	1	1	6/15/2023	85
2	2	2	6/20/2023	78
3	3	3	6/25/2023	92
4	2	4	6/15/2023	78
5	3	5	6/20/2023	90
6	3	6	6/25/2023	65

Figure 27 Event Table

Event Table(3NF)

EventID	Event Name	Date	Location
1	Annual Day	8/10/2023	School Grounds
2	Sports Day	9/20/2023	Sports
3	Science Fair	10/15/2023	Science Lab
4	Parent-Teacher Meeting	7/30/2023	School Auditorium

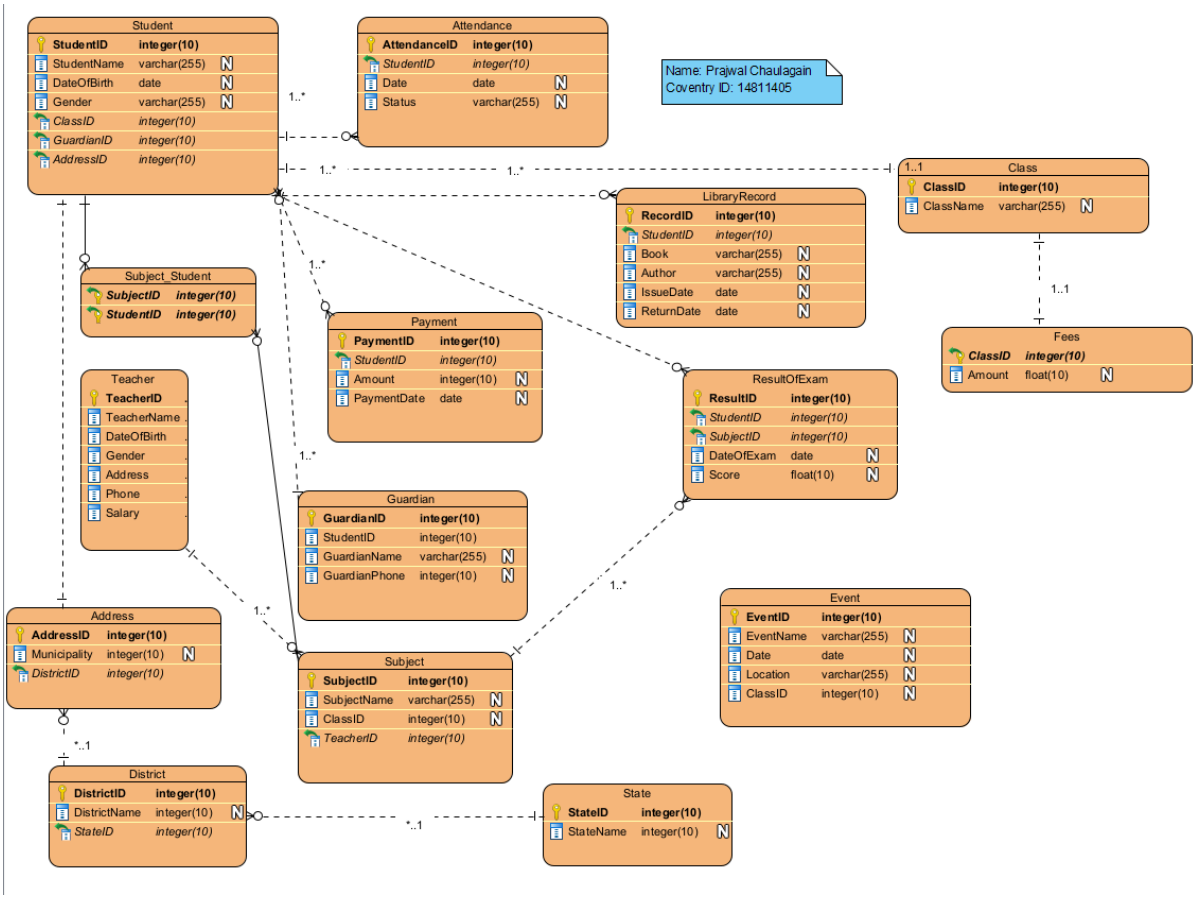
Figure 28 Library Record Table

Library Record Table(3NF)

RecordID	StudentID	Book Title	Author	Issue Date	Return Date
1	1	Java the complete reference (12th	Herbert Schildt	7/5/2023	7/20/2023
2	2	Spidering	Tara Calishain	6/30/2023	7/10/2023
3	3	Windows System Programming	Johnson M. Hart	7/10/2023	7/25/2023
4	4	C# in Depth	Jon Skeet	7/15/2023	7/25/2023

Explaining the ER Diagram

Figure 29 ER Diagram



An Entity-Relationship a visual tool used to represent an information system, showing how entities within a database are related. For New Horizon Boarding Secondary School, The ER diagram includes entities like Student, Teacher, Subject, Exam Result, Payment, Guardain, Event, Library Record, and Attendence. Each of these entities, such as a student or a subject, represents a unique component of the system and is defined by specific attributes, like StudentID, Student Name, and DateOfBirth for the Student entity. (Nishadha & Creately, 2024)

SQL Queries

SQL (Structure Query Language) is a programming language for managing relational databases. It enables the storage, retrieval, updating, entry, and analysis of data in tables. This make SQL essential for data management and analysis tasks. (Vaitkun, 2022)

Figure 30 Guardian Table

```
5  CREATE TABLE IF NOT EXISTS Guardian (
6      GuardianID INT PRIMARY KEY,
7      GuardianName VARCHAR(100),
8      GuardianPhone VARCHAR(15)
9  );
0
```

sult Grid | Filter Rows: | Export: | Wrap Cell Conter

Field	Type	Null	Key	Default	Extra
GuardianID	int	NO	PRI	NULL	
GuardianName	varchar(100)	YES		NULL	
GuardianPhone	varchar(15)	YES		NULL	

```
11  DESCRIBE Guardian;
12  SELECT * FROM Guardian;
13
```

Result Grid | Filter Rows:

GuardianID	GuardianName	GuardianPhone
1	Gita Acharya	9847011111
2	Ram Sharma	9816412121
3	Hari Rai	9867066700
NULL	NULL	NULL

Figure 31 Student Table

```
12  CREATE TABLE IF NOT EXISTS Student (
13      StudentID INT PRIMARY KEY,
14      Name VARCHAR(50),
15      DateOfBirth DATE,
16      Gender VARCHAR(10),
17      Address VARCHAR(100),
18      GuardianID INT,
19      FOREIGN KEY (GuardianID) REFERENCES Guardian(GuardianID)
20  );
```

Result Grid | Filter Rows: | Edit: | Export/Import:

StudentID	Name	DateOfBirth	Gender	Address	GuardianID
101	Ramesh Acharya	2005-03-15	Male	Butwal-10, Rupandehi, Lumbini	1
102	Sita Sharma	2006-06-20	Female	Taulihawa-5, Kapilvastu, Lumbini	2
103	Asha Rai	2004-11-10	Female	Siddharthanagar-12, Lumbini	3
104	Rita Rai	2007-01-25	Female	Butwal-14, Rupandehi, Lumbini	4
NULL	NULL	NULL	NULL	NULL	NULL

Figure 32 Teacher Table

```
30 • CREATE TABLE IF NOT EXISTS Teacher (
31       TeacherID INT PRIMARY KEY,
32       Name VARCHAR(100) NOT NULL,
33       DateOfBirth DATE,
34       Gender VARCHAR(10),
35       Address VARCHAR(255),
36       Phone VARCHAR(15),
37       Salary DECIMAL(10, 2)
38   );
39
```

esult Grid | Filter Rows: | Export: | Wrap Cell Cx

Field	Type	Null	Key	Default	Extra
TeacherID	int	NO	PRI	NULL	
Name	varchar(100)	NO		NULL	
DateOfBirth	date	YES		NULL	
Gender	varchar(10)	YES		NULL	
Address	varchar(255)	YES		NULL	
Phone	varchar(15)	YES		NULL	
Salary	decimal(10,2)	YES		NULL	

```
41 • DESCRIBE Teacher;
42 • SELECT * FROM Teacher;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

TeacherID	Name	DateOfBirth	Gender	Address	Phone	Salary
1	Ashok Aryal	1980-05-09	Male	Tansen-8, Palpa, Lumbini	9867389098	40000.00
2	Basanta Poudel	1975-04-25	Male	Bhairahawa-1, Rupandehi, Lumbini	9876787887	45000.00
3	Lekhnath Panta	1985-01-28	Male	Butwal-12, Rupandehi, Lumbini	9856788993	60000.00
4	Roman Shrestha	1990-06-02	Male	Butwal-12, Rupandehi, Lumbini	9843678567	55000.00
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 33 Class Subject

```
45 • CREATE TABLE IF NOT EXISTS Subject (
46       SubjectID INT PRIMARY KEY,
47       SubjectName VARCHAR(100) NOT NULL
48   );
```

Result Grid | Filter Rows: | Export: | Wrap

Field	Type	Null	Key	Default	Extra
SubjectID	int	NO	PRI	NULL	
SubjectName	varchar(100)	NO		NULL	

```
51 •  DESCRIBE Subject;
52 •  SELECT * FROM Subject;
```

Result Grid | Filter Rows:

SubjectID	SubjectName
1	Mathematics
2	Science
3	English
4	Social Studies
NULL	NULL

Figure 34 Fees Table

```
44 • CREATE TABLE IF NOT EXISTS Fee (
45       Class VARCHAR(10) PRIMARY KEY,
46       Amount DECIMAL(10, 2)
47   );
48
```

Result Grid | Filter Rows: | Edit:

Class	Amount
1	3000.00
10	8400.00
2	3600.00
3	4200.00
4	4800.00
5	5400.00
6	6000.00
7	6600.00
8	7200.00
9	7800.00
KG	2400.00
Nurs...	1800.00
NULL	NULL

Figure 35 Payment Table

```
84 • CREATE TABLE IF NOT EXISTS Payment (
85       PaymentID INT PRIMARY KEY,
86       StudentID INT,
87       Amount DECIMAL(10, 2) NOT NULL,
88       PaymentDate DATE,
89       FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
90   );
```

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

Field	Type	Null	Key	Default	Extra
PaymentID	int	NO	PRI	NULL	
StudentID	int	YES	MUL	NULL	
Amount	decimal(10,2)	NO		NULL	
PaymentDate	date	YES		NULL	

```
54 • CREATE TABLE IF NOT EXISTS Payment (
55       PaymentID INT PRIMARY KEY,
56       StudentID INT,
57       Amount DECIMAL(10, 2),
58       PaymentDate DATE,
59       FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
60   );
61
```

Result Grid | Filter Rows: | Edit: | Export/Import:

PaymentID	StudentID	Amount	PaymentDate
1	101	5500.00	2023-07-10
2	102	6500.00	2023-07-10
3	103	5500.00	2023-07-05
4	104	6500.00	2023-08-01
NULL	NULL	NULL	NULL

Figure 36 Attendance Table

```
63 • CREATE TABLE IF NOT EXISTS Attendance (
64       AttendanceID INT PRIMARY KEY,
65       StudentID INT,
66       Date DATE,
67       Status VARCHAR(10),
68       FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
69   );
70
```

Result Grid | Filter Rows: | Edit: | Export/Import:

AttendanceID	StudentID	Date	Status
1	101	2023-07-01	Present
2	102	2023-07-02	Absent
3	103	2023-07-03	Present
4	104	2023-08-01	Present
NULL	NULL	NULL	NULL

Figure 37 Result Table

```
72 • CREATE TABLE IF NOT EXISTS ExamResult (
73       ResultID INT PRIMARY KEY,
74       StudentID INT,
75       SubjectID INT,
76       DateOfExam DATE,
77       Score INT,
78       FOREIGN KEY (StudentID) REFERENCES Student(StudentID),
79       FOREIGN KEY (SubjectID) REFERENCES Subject(SubjectID)
80   );
```

Result Grid | Filter Rows: | Edit: | Export/Import:

ResultID	StudentID	SubjectID	DateOfExam	Score
1	101	1	2023-06-15	85.00
2	102	2	2023-06-20	78.00
3	103	3	2023-06-25	92.00
4	104	4	2023-08-01	88.00
NULL	NULL	NULL	NULL	NULL

Figure 38 Event Table

```
130 • ⊖ CREATE TABLE IF NOT EXISTS Event (
131         EventID INT PRIMARY KEY,
132         EventName VARCHAR(100),
133         Date DATE,
134         Location VARCHAR(255)
135     );
136
```

Result Grid | Filter Rows: | Export: | Wrap

Field	Type	Null	Key	Default	Extra
EventID	int	NO	PRI	NULL	
EventName	varchar(100)	YES		NULL	
Date	date	YES		NULL	
Location	varchar(255)	YES		NULL	

```
138 •   DESCRIBE Event;
139 •   SELECT * FROM Event;
```

Result Grid | Filter Rows: | Edit: | E

EventID	EventName	Date	Location
1	Annual Day	2023-08-15	School Ground
2	Sports Meet	2023-09-10	School Stadium
3	Science Fair	2023-10-05	School Auditorium
4	Parent-Teacher Meeting	2023-07-30	School Auditorium
NULL	NULL	NULL	NULL

Figure 39 Library Table

```
91  CREATE TABLE IF NOT EXISTS LibraryRecords (
92      RecordID INT PRIMARY KEY,
93      StudentID INT,
94      BookTitle VARCHAR(100),
95      Author VARCHAR(50),
96      IssueDate DATE,
97      ReturnDate DATE,
98      FOREIGN KEY (StudentID) REFERENCES Student(StudentID)
99  );
100
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

RecordID	StudentID	BookTitle	Author	IssueDate	ReturnDate
1	101	Mathematics for Class 10	John Doe	2023-06-10	2023-06-24
2	102	Science for Class 9	Jane Smith	2023-06-15	2023-07-01
3	103	English Grammar	Michael Johnson	2023-06-20	2023-07-05
4	104	Social Studies Book	Emily Davis	2023-08-01	2023-08-15
5	102	Up from Slavery	Booker T. Washington	2023-07-11	2023-07-20
6	101	The Book of HACKER SCHOOL	Peter Krumins	2023-07-01	2023-07-14
7	102	Pragmatic Thinking & Learning	Andy Hunt	2023-06-25	2023-07-03
8	103	Data Mining Concepts and Techniques	Jiawei Han and Micheline Kamber	2023-07-07	2023-07-16
NULL	NULL	NULL	NULL	NULL	NULL

Data Visualization

We cleaned up the AirlineReview.csv dataset by filling in or removing missing values and then created several visualizations to analyze the data. The line graphs showed how the Overall Score changed over time, while the bar graph compared the number of reviews for each airline. The histogram displays the distribution of Overall Scores, and the box-whisker plot highlights score variations across different airlines. Lastly, the pie chart illustrates the proportion of reviews for each airline, giving us insights into market share and review distribution.

1. Line graph

Figure 40 Line graph

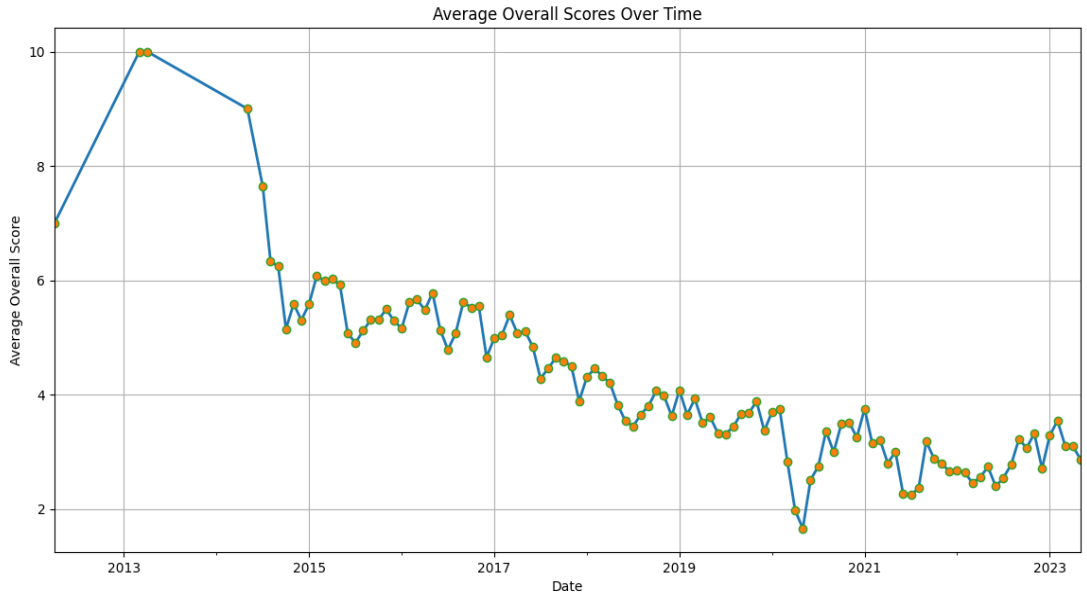


Figure 41 Imported libraries and file

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from dateutil import parser


link = r'C:\java\New folder\AirlineReviews.csv.zip'
airline_reviews = pd.read_csv(link)

# Convert 'DateFlown' to datetime format for time series analysis
airline_reviews['DateFlown'] = pd.to_datetime(airline_reviews['DateFlown'], errors='coerce')
```

Figure 42 Line graph code

```
# Average Overall Scores Over TimeAverage Overall Scores Over Time
monthly_avg_scores = airline_reviews.groupby(airline_reviews['DateFlown'].dt.to_period('M'))['OverallScore'].mean().dropna()

 (variable) monthly_avg_scores: Series[Any]
monthly_avg_scores.plot(kind='line', marker='o', color='#1f77b4', linestyle='-', linewidth=2, markerfacecolor='#ff7f0e', markeredgecolor='#2ca02c')
plt.title('Average Overall Scores Over Time')
plt.xlabel('Date')
plt.ylabel('Average Overall Score')
plt.grid(True)
plt.show()
```

2. Bar Graph

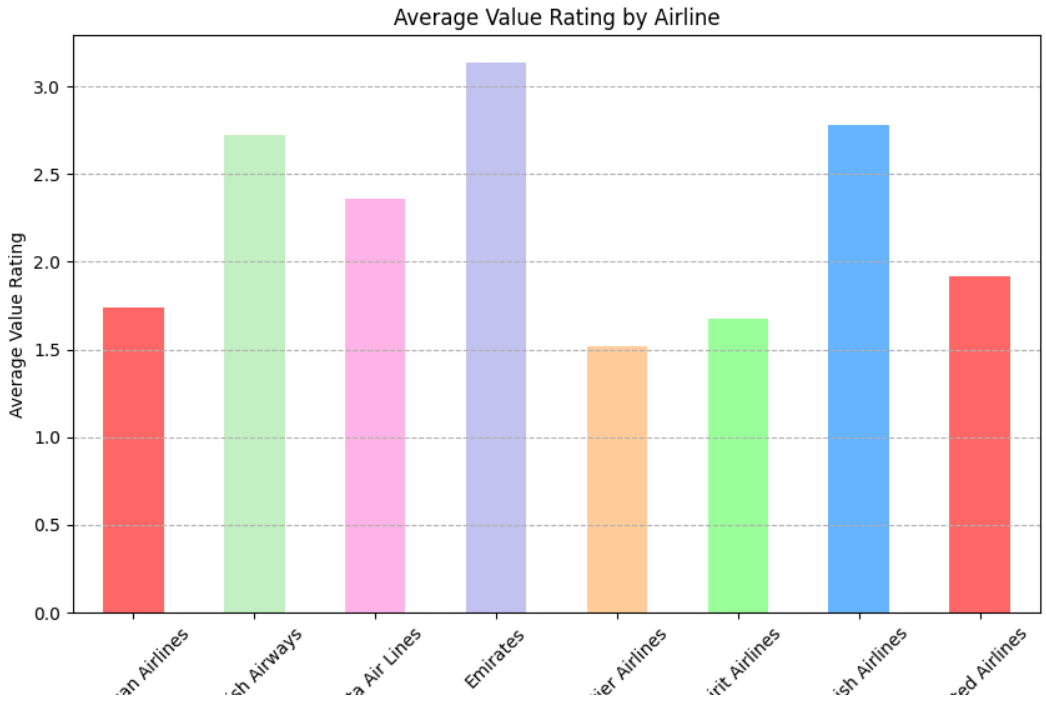


Figure 43 Bar graph code

```
# Create the bar graph
# Select 8 airlines to compare
selected_airlines = airline_reviews['AirlineName'].value_counts().nlargest(8).index
df_selected = airline_reviews[airline_reviews['AirlineName'].isin(selected_airlines)]

# Calculate average ValueRating by airline
avg_value_ratings = df_selected.groupby('AirlineName')['ValueRating'].mean()

# Define individual colors for each bar
colors = ['#ff6666', '#c2f0c2', '#ffb3e6', '#c2c2f0', '#ffcc99', '#99ff99', '#66b3ff']

# Plotting the bar graph
plt.figure(figsize=(10, 6))
avg_value_ratings.plot(kind='bar', color=colors)
plt.title('Average Value Rating by Airline')
plt.xlabel('Airline')
plt.ylabel('Average Value Rating')
plt.xticks(rotation=45)
plt.grid(axis='y', linestyle='--')
plt.show()
```

3. Histography

Figure 44 Histography

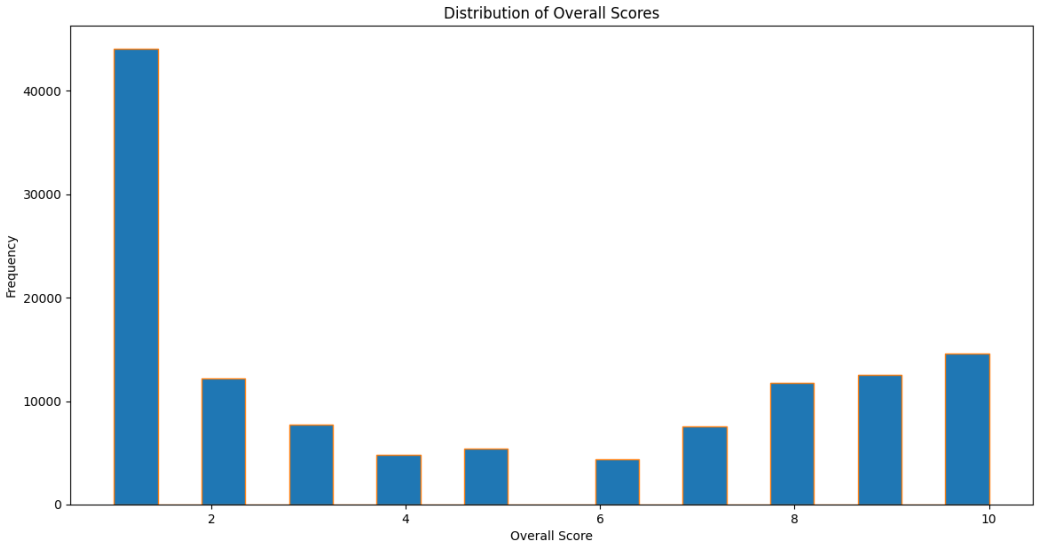


Figure 45 Histography code

```
# Distribution of Overall Scores
plt.figure(figsize=(14, 7))
airline_reviews['OverallScore'].dropna().plot(kind='hist', bins=20, color='#1f77b4', edgecolor='#ff7f0e')
plt.title('Distribution of Overall Scores')
plt.xlabel('Overall Score')
plt.ylabel('Frequency')
plt.show()
```

4. Box-whisker plot

Figure 46 Box-Whisker Plot

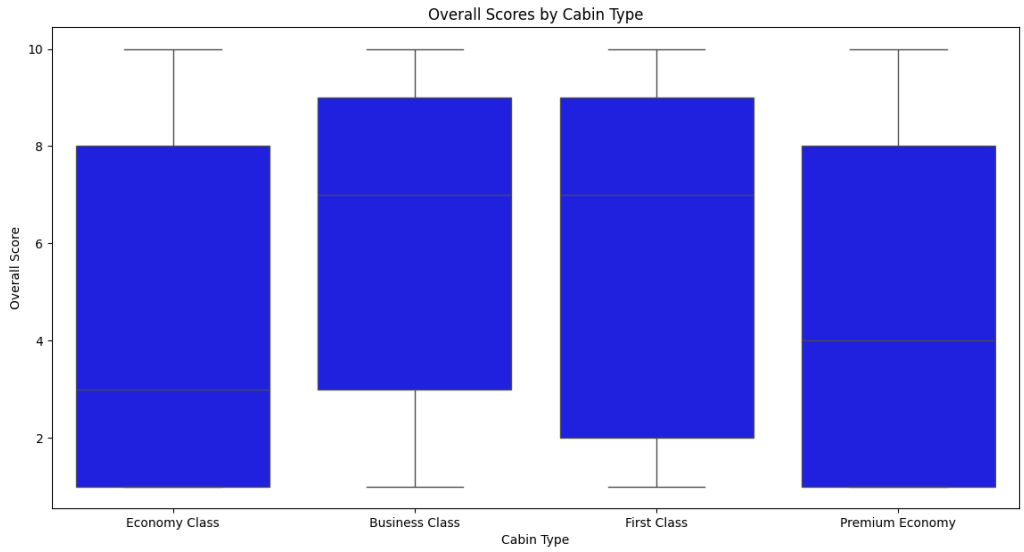


Figure 47 Box-Whisker Plot Code

```
# Overall Scores by Cabin Type
plt.figure(figsize=(14, 7))
sns.boxplot(data=airline_reviews, x='CabinType', y='OverallScore',color="Blue")
plt.title('Overall Scores by Cabin Type')
plt.xlabel('Cabin Type')
plt.ylabel('Overall Score')
plt.show()
```

5. Pie Chart

Figure 48 Pie Chart

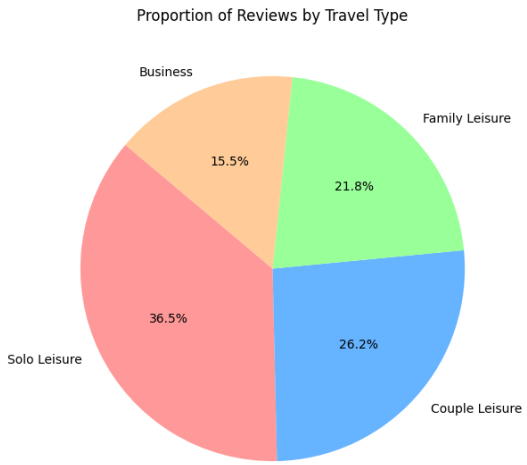


Figure 49 Pie Chart Code

```
69  # Proportion of Reviews by Travel Type
70  travel_type_counts = airline_reviews['TravelType'].value_counts()
71
72  (variable) travel_type_counts: Series[int]
73  travel_type_counts.plot(kind='pie', autopct='%1.1f%%', startangle=140, colors=['#ff9999', '#66b3ff', '#99ff99', '#ffcc99'])
74  plt.title('Proportion of Reviews by Travel Type')
75  plt.ylabel('')
76  plt.show()
```

Conclusion

The job also required fixing the New Horizon Institute's database system. Visual Paradigm was used in this project's normalization process for its thorough design and organization. MySQL Workbench was used to run database queries, which made effective data management possible. Python packages like Pandas and Matplotlib's Pyplot were used for data visualization. Pandas was utilized for data analysis and manipulation, while Matplotlib's Pyplot allowed for the production of a number of intelligent visual representations of the data.

Reference

Ian. (2017, October 6). *What is Normalization?* https://database.guide/what-is-normalization/

Vaitkun, D. (2022, August 9). *What is an SQL query?* LearnSQL.com. https://learnsql.com/blog/what-is-sql-query/

Nishadha, & Creately. (2024, March 23). *What is an Entity Relationship Diagram (ERD)? | An Introduction to ER Diagram*. Creately. https://creately.com/guides/er-diagrams-tutorial/