

SIT 103 – DATA AND INFORMATION MANAGEMENT
ASSESSMENT TASK 2
PART 2
DATABASE DESIGN, IMPLEMENTATION, AND
INTERACTION

REPORTED BY
CHIRAKSHI GUPTA
221397163

CONTENTS

2.1 BUSINESS SCENARIO:.....	3
2.2 LIST OF ENTITIES:	4
2.3 NORMALISATION:	9
2.4 ENTITY RELATIONSHIP DIAGRAM (ERD):.....	14
2.5 IMPLEMENTATION OF THE DATABASE SCHEMA:	15
2.6 INSERTING DATA INTO TABLES:.....	24
2.7 ALTER AND UPDATE TABLE:	33
2.8 DATA MANIPULATION USING SQL COMMANDS:	36
2.9 DATA SECURITY AND PRIVACY:	40
2.10 DATA SECURITY AND PRIVACY:	41

2.1 BUSINESS SCENARIO:

Budha Group of Institutions offers educational services in form of courses and events in seven different streams through four of its branches (BCE, BCA, BCM and BCHE) and campuses around Karnal. The four colleges offer a total of 9 different courses with specialisation in 6 different fields. BCE offers B.Ed. and D.El.Ed, BCA offers B.Arch., BCM offers MBA, BBA and BCHE offers B.Sc. (Fashion Design, Interior Design), B.Com. and BFA. Every course has its own duration, fees, requirements, students, placement ratings, eligibility etc. Each course is specialised in one of the six (Fashion Design, Finance, Human Resources, International Business, IT & Systems, and Sales & Marketing) specialisations. Each branch, or campus has two types of employees: Management staff or teaching staff. There is a Head of Department for each specialisation field who updates the contents and manages the professors of his department. Each branch offers services like Gym, Yoga, Cafeteria, Library and Sports facilities for students only.

Each Head of Department can employ Professors, and Assistant Professors as teaching staff. Each professor can hire one intern at most. Each professor gets one lecture a day and a lecture takes place even if there is a single student but is cancelled if there is zero attendance. Each student can be taught by different professors, but one teacher can teach only one subject. Every intern reports to assistant professor, who in turn reports to main professor, who reports to the head of the department, who reports to the principal, who finally reports to the managing director. There is a finance department that prepares budget for each department and reports every month to the managing director. Finance reports contain monthly budgets, turn-over, profits, losses, assets, liabilities, pending payments, etc. It receives reports from accounts, reception, **events**, and marketing departments.

Accounts department reports to finance department every month. It keeps the total record of , scholarships provided, maintenance costs, advertisement, admission expenses, bank charges, telephone expenses, inspection charges, travelling expenses, staff wages and employee salaries, miscellaneous, university charges, events and special lectures expenses, student, and employee tours costs, etc. It receives fees from the students and also pays wages to the employees into their bank accounts. Students can pay fee as cash, through a card, or a valid cheque, but no other modes of payment are allowed. Account department is responsible for providing scholarships and hostels if available. Each college branch has a hostel, separate for boys and girls, a student has the choice to take a room and can choose from single, double, and three-bed room set.

Each college in the organization also has a reception to attend visitors, students and give them details. This department accepts student application forms, records applicant's data, frequently contacts applicants for updates, solves any existing student issues, replies to enquiries, daily needs and requests from students and other staffs and generates monthly reports. The applications are sent to the administration. The administration department rejects or interviews a student and final status is updated to reception. The applicants receive all the confirmations by email from reception and

also any further details after admission. The admin stores the student's data and status for current as well as old students. It updates receptions about the new events and notices. Admin also keeps the record of all the employees and staff, chairman and directors. Admin department will also update the website timely and record the data for last time it was updated.

Another important department is the Marketing department, which takes care of advertising colleges through pamphlets and news, increasing publicity. They socialise through social media platforms and other means to promote the institution and the final report is given to the finance department by them every month. The marketing department also reports to the director of marketing, who reports to the Chairman. There is also an events department that organises all the events related to academics as well as co-curricular. They will be storing name, venue, date, timings, and the participants of the event. The monthly reports will go to finance department again to prepare a budget and accommodate all the expenses. The Chairman is the highest authority and is reported by the managing directors and marketing director with combined reports from finance department and marketing department respectively.

2.2 LIST OF ENTITIES:

- 1) **EMPLOYEES** [Emp Id (PK), Emp Name (PK), Emp DOB, Emp Address, Emp Contact, Emp Job Status, Emp Joining Date, Emp Type]

Employee is considered as an entity so that all the data related to employees can be stored in a table. According to business scenario, employee is an important element as it includes teachers and management. It is a supertype because of different Emp Type and its subtypes are Teaching Staff and Management Staff.

- 2) **TEACHING STAFF** [Emp Id (PK, FK), Emp Name (PK, FK), Emp Designation, Working Tenure]

It has been chosen as an entity to store the data for all the employees that teach, including their designations specifically. It is a supertype of three types of teaching staff, i.e., professors, assistant professors, and interns.

- 3) **MANAGEMENT STAFF** [Emp Id (PK, FK), Emp Name (FK), Emp Department, Working Tenure, Emp Salary, Emp Designation, Salary Increment]

It is a completely different entity because it represents a type of employees that work in the colleges but are not teachers. It is essential to record their details for salary payments, contact and other things.

- 4) **PROFESSOR** [Emp Id (PK, FK), Emp Name (PK, FK), Department Name (FK), Lecture Name (FK), Emp Salary, Quarterly Increment, Number of Assistant Professors]

One of the designations in teaching staff, who reports to the head of department but also trains an intern. Professors are reported by Assistant professors but give only one lecture in a particular subject daily. It is important to store the details of a professor to know their schedule, personal details, and everything.

- 5) **ASSISTANT PROFESSOR** [Emp Id (PK, FK), Emp Name (PK, FK), Department Name (FK), Reports to Professor Id (FK), Emp Salary, Yearly Increment]

A designation below Professors, who takes all the reports of the interns and works in a specific department. They get salary and increments, and are an important entity so that their records can be kept separately for their tenure, personal details, head professor, etc.

- 6) **INTERN** [Emp Id (PK, FK), Emp Name (PK, FK), Emp Salary, Department Name (FK), Reports to Assistant Professor Id (FK)]

These are the newbies but paid ones. They work under professors and assistant professors. Their details are important to be kept as a separate entity for when required and also to make their identities, certificates and assign them professors, departments, etc.

- 7) **HEAD OF DEPARTMENT** [Emp Id (PK), Emp Name, Department Name (PK, FK), Working Tenure, Emp Salary, Quarterly Increment, Number of Professors Reporting]

Every department needs a head to manage everything, when teachers are busy tutoring, so HOD manages all the professors, department needs and get paid along with increments, so, it is important for such officials to have a separate table for records, which becomes quite easy to find when needed.

- 8) **PRINCIPAL** [P Name (PK), P DOB, P Contact, Working Tenure]

To manage every college, there is a highly qualified principal, whose details must be kept by the institution like any other employee for incentives, trips, awards, cases, payments, etc.

- 9) **SPECIALISATION/ DEPARTMENT** [Department Name (PK), Department Code, Number of Courses]

Because there are many different departments, we need a separate table that would store all the details of these departments. Now, addition of courses in department, removal of courses, or addition/ removal of new/ old departments will not affect the stored records reducing the chances of data inconsistencies.

10) **LECTURES** [Lect Name (PK), Lect Duration, Lect Timings]

Professor gives lectures and students attend them but there are numerous lectures and for students and professors to get aware of them, it is first necessary to store their details as a separate entity without any confusions.

11) **STUDENTS** [Stud Id (PK), Stud Full Name, Stud DOB, Stud Address, Stud Course Name (FK), Stud Accommodation Type, Stud Contact, Stud Emergency Number]

One of the most important entity in an educational institution, which would record the details of all the students it teaches.

12) **STUDENT-LECTURES** [Lect Name (PK, FK1), Stud Id (PK, FK2)]

It is an associative-entity to implement the many-many relationships between students and lectures. A student attends many lectures, and a lecture has different students, but to know exactly which lecture has which student, this entity is important one.

13) **COURSES** [Course Name (PK), Specialisation Name (FK), Branch Name (FK), Fees, Duration, Placement Ratings, Eligibility]

There are different courses provided by the university, but for a student to choose and university to improve, there must be details of the courses that already are offered by the university, so that any changes made to this data do not affect other records in any entity.

14) **BRANCHES** [Branch Name (PK), Location, Principal Name (FK), Number of Courses]

This group has four different colleges, this entity keeps them connected by keeping the records of all of them. These details kept in a separate entity are modifiable and important for people to know.

15) **SERVICES** [Service Name (PK), Monthly Charges, Availability Hours, Additional Comments]

Each branch provides some services, but they are a part of the institution, and all the profits or losses are included, so it becomes important to keep a record of them.

16) **BRANCH-SERVICE** [Branch Name (PK, FK1), Service Name (PK, FK2)]

This is another associative entity also called bridge entity. It is used to implement the many-many relationships between the services provided by different branches. It keeps a record of exactly what service is provided by which branch or college.

17) **HOSTEL** [Month Year (PK), Hostel Number (PK), Number of Rooms Available, Room Type, Fee per Room]

For those students, who are willing to take the hostel, database must keep records of every hostel available with desired rooms. This can be easily updated in case of construction or renovations.

18) **ACCOUNTS DEPARTMENT** [Month Year (PK), Report Id (PK), Scholarships, Expenses, Service Charges, Event Tour Costs, Salaries Paid, Fee Received]

It is an important department of the university and thus, it must have an extra entity to store all the accounts- related information of the college.

19) **RECEPTION** [Month Year (PK), Report Id (PK), Number of Applicants, Number of Enquiries, Number of Notices]

Every building has a reception that needs to store lots of data daily, so a separate entity is a requirement here.

20) **APPLICANT'S DATA** [Applicant Number (PK), Applicant Full Name, Applicant Address, Applicant DOB, Applicant Contact, Applicant Course, Applicant Qualifications, Date of Applying]

Another important entity of an educational institution is to keep records of those who apply, so that they can be admitted, rejected, provided scholarships, hostels, take payments etc.

21) **ADMIN** [Month Year (PK), Interviews Taken, New Admissions, Rejected Applications, Last Website Update]

Main block of a group, admin stores all the information that is very important safely.

22) EVENTS DEPARTMENT [Month Year (PK), Report Id (PK), Academic Events, Extra Events, Total Costs]

This university organises many events frequently, therefore it needs to store the details of past, current as well as future events under this table.

23) MARKETING DEPARTMENT [Month Year (PK), Report Id (PK), Advertising Costs, Advertising Profits]

Another important department of college that stores all the details of money and resources spent o marketing and if it is beneficial.

24) MONTHLY REPORTS [Month Year (PK), Finance Report Id (PK, FK1), Events Report Id (PK, FK2), Accounts Report Id (PK, FK3), Marketing Report Id (PK, FK4), Reception Report Id (PK, FK5)]

This is a crucial entity for analysis of smooth working of the organisation. It can be checked by any department to know budgets, profits, losses and especially by directors and Chairman to avoid any fraudulent activities that may happen.

25) FINANCE DEPARTMENT [Month Year (PK), Report Id (PK), Turn-Over, Pending Payment, Liabilities, Assets, Taxes, Equity]

The most important department to prepare all the budgets by collecting all the reports from different departments. It keeps records of all the assets, liabilities and other things for the university making it better to manage for the directors and chairman.

26) DIRECTORS [Director Initials (PK), Director Name(PK), Working Tenure, Income, Reported by (FK), Director Type]

They are the second highest authority to manage the functionality and maintain quality, but their records must be kept and open to all for fair working and no scams.

27) CHAIRMAN [CM Initials (PK), CM Name (PK), CM DOB, CM Address, CM Contact Number]

He owns the institution and there is only one at a time, so to keep the record of chain of chairmen over the years, it is important for this entity to exist.

2.3 NORMALISATION:

1) EMPLOYEES

Intersection of every row and column gives a unique value, that means the table is already in first Normal form.

There are no partial dependencies as all those things that are dependent on Emp Id are also dependent on Emp Name. Also, Emp Id is dependent on Emp Name. Thus, the table is also in 2NF.

Because all the attributes other than primary key are related to primary key and not to each other, this means that the table is normalised till level 3 and there are no partial dependencies.

2) TEACHING STAFF

There is only one value per cell, which is unique and also Emp Id + Emp Name is defined as primary key. This means table is normalised in 1NF.

None of the non-key attribute partially depends on primary key attributes, therefore 2NF is satisfied.

Working Tenure and Designation are both related to Employee Name and Id. None of them is related to each other, i.e., there is no transitive dependency, so 3NF is satisfied.

3) MANAGEMENT STAFF

All the records are unique with a defined primary key, so there is no need to normalise at level 1.

There is only one primary key, which moves us to third level directly.

In third level, we look for transitive dependencies and there is one between Emp Designation and Salary and Salary Increments because Designation decides salary and salary decides increments. So, we form a separate entity for these attributes with Emp Designation in both, but as primary key in new entity.

4) PROFESSOR

All the attributes will contain a single record without any repeating groups. Also, there is a primary key, which means that it satisfies 1NF.

Emp Id and Name are themselves dependent on each other so there are no partial dependencies, i.e., 2NF is also satisfied.

All the attributes are related to the employ but none to each other, leaving zero chance of any transitive dependency. So, it is normalised till 3NF.

5) **ASSISTANT PROFESSOR**

All the entries are unique with a composite primary key, but the attributes in the key are themselves dependent on each other, that means there is no chance of partial dependency. So, the table satisfies 1NF and 2NF. As there are no transitive dependencies in attributes: Department Name, Professor Id, Salary and Fixed Increment, it is automatically normalised up to 3NF.

6) **INTERN**

It is normalised up to 3NF, same as it was for Assistant Professor.

7) **HEAD OF DEPARTMENT**

All the attributes are unique with a composite primary key – 1NF.

No partial dependencies – 2NF.

No transitive dependencies – 3NF.

8) **PRINCIPAL**

All the attributes are unique with a composite primary key – 1NF.

Only one Primary key – 2NF.

No transitive dependencies, all depend on Primary Key – 3NF.

9) **SPECIALISATION/ DEPARTMENT**

Unique, non-repeating values with one Primary key – 1 & 2NF.

No transitive relation between Department Code and Number of Courses, so 3NF is also satisfied.

10) **LECTURES**

Unique, non-repeating values with one Primary key – 1 & 2NF.

Lecture Duration and Timings are dependent on each other but also on lecture name, so it is already normalised till 3NF.

11) **STUDENTS**

Unique, non-repeating values with one Primary key – 1 & 2NF.

No transitive dependencies as all the data depends on the student and Stud Id is the primary key. – 3NF

12) **STUDENT-LECTURES**

No attribute other than primary key, so unique records and leaves no chance of partial or transitional dependencies to occur. – 1, 2 & 3NF

13) **COURSES**

Unique, non-repeating values with one Primary key – 1 & 2NF.

Details related to the course can be retrieved by just course name so there is no significance of any transitive dependency here, so it is already normalised in 3rd form.

14) **BRANCHES**

Unique, non-repeating values with one Primary key – 1 & 2NF.

Principal's name, location and number of courses offered are not related to each other but the name of the branch that shows there are no signs of transitive dependencies. – 3NF

15) **SERVICES**

Unique, non-repeating values with one Primary key – 1 & 2NF.

Charges are independent of hours and comments are meant for the service, so no transitive dependency – 3NF satisfied.

16) **BRANCH-SERVICE**

No attribute other than primary key, so unique records and leaves no chance of partial or transitional dependencies to occur. – 1, 2 & 3NF

17) **HOSTEL**

Unique, non-repeating values with defined Primary key – 1 NF.

Time of the year and the quality of the building decides the availability of particular type of room and charges are not fixed. So, no partial dependency.

– 2NF

No transitive dependencies – 3NF

18) **ACCOUNTS DEPARTMENT**

All the attributes are unique with a composite primary key – 1NF.

Different expenses can only be mentioned in a report, and they surely will vary with month, so there is complete dependency only – 2NF satisfied

No transitive dependencies – 3NF

19) **RECEPTION**

All the attributes are unique with a composite primary key – 1NF.

Applications, Enquiries and Notices are part of a report and vary monthly, so there are no partial dependencies. 2NF satisfied

No dependent-determinant relation between attributes other than key so no transitive dependency – 3NF satisfied.

20) **APPLICANT'S DATA**

Unique, non-repeating values with one Primary key – 1 & 2NF.

No transitive dependencies as all the data depends on the Applicant and Application Number is the primary key. – 3NF

21) **ADMIN**

Unique, non-repeating values with one Primary key – 1 & 2NF.

Completely different information in rest of the attributes so no chance of existence of transitional dependencies. – 3NF

22) **EVENTS DEPARTMENT**

All the attributes are unique with a composite primary key – 1NF.

Report varies every month and changes are visible in combined report represented by Report Id, so no partial but only complete dependencies exist. – 2NF satisfied.

No transitive dependencies – 3NF

23) **MARKETING DEPARTMENT**

All the attributes are unique with a composite primary key – 1NF.

Report varies every month and changes are visible in combined report represented by Report Id, so no partial but only complete dependencies exist.
– 2NF satisfied.

No transitive dependencies – 3NF

24) MONTHLY REPORTS

All the attributes are unique with a composite primary key – 1NF.

Report varies every month and changes are visible in combined report represented by Report Id, so no partial but only complete dependencies exist.
– 2NF satisfied.

No transitive dependencies – 3NF

25) FINANCE DEPARTMENT

All the attributes are unique with a composite primary key – 1NF.

Report varies every month and changes are visible in combined report represented by Report Id, so no partial but only complete dependencies exist.
– 2NF satisfied.

No transitive dependencies – 3NF

26) DIRECTORS

Unique values with unique primary key composed of Name and the initials. – 1NF

All the attributes depend on both parts of the key because attributes of the key are dependent. – 2NF

No transitive dependency among Director's details. -3NF

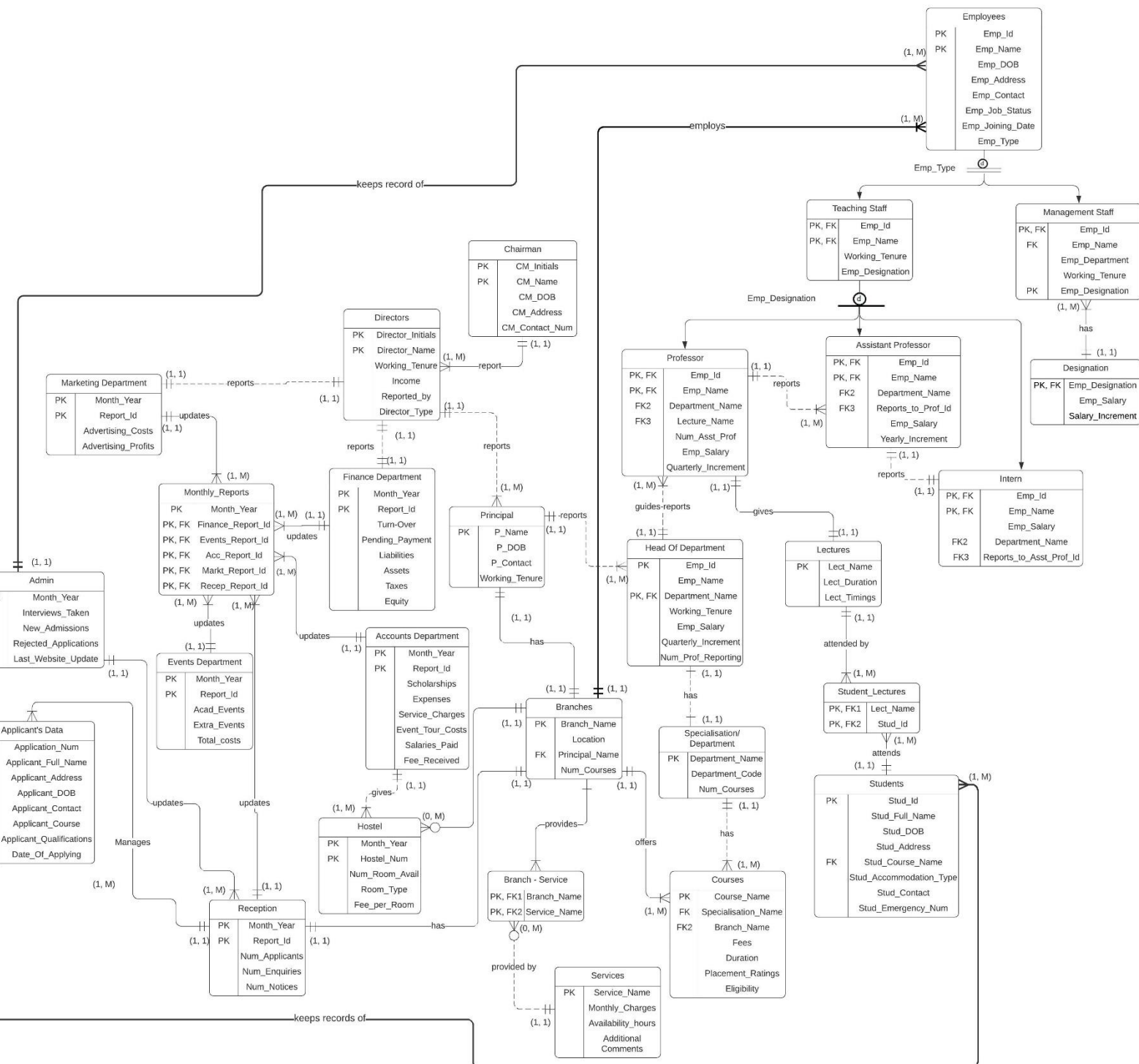
27) CHAIRMAN

Unique values with unique primary key – 1NF

All the attributes depend on both parts of the key because attributes of the key are dependent. – 2NF

No transitive dependency among Chairman's details. -3NF

2.4 ENTITY RELATIONSHIP DIAGRAM (ERD):



2.5 IMPLEMENTATION OF THE DATABASE SCHEMA:

```
/* Introduction to SQL          */
```

```
/* Script file for ORACLE DBMS */
```

```
/* Script file loads all 28 tables */
```

```
CREATE TABLE DEPARTMENT (
```

```
DEPARTMENT_NAME  VARCHAR2(35) NOT NULL  
CHECK(DEPARTMENT_NAME IN ('FASHION DESIGN', 'FINANCE', 'HUMAN  
RESOURCES', 'INTERNATIONAL BUSINESS', 'IT', 'SALES AND MARKETING')),
```

```
DEPARTMENT_CODE  CHAR(4) NOT NULL CHECK(DEPARTMENT_CODE IN  
('FD01', 'FN02', 'HR03', 'IB04', 'IT05', 'SM06')),
```

```
NUM_COURSES      INTEGER NOT NULL,
```

```
PRIMARY KEY (DEPARTMENT_NAME));
```

```
CREATE TABLE EMPLOYEES (
```

```
EMP_ID          VARCHAR2(10) NOT NULL,
```

```
EMP_NAME        VARCHAR2(35) NOT NULL,
```

```
EMP_DOB         DATE NOT NULL,
```

```
EMP_ADDRESS     VARCHAR2(40) NOT NULL,
```

```
EMP_CONTACT     CHAR(10) NOT NULL,
```

```
EMP_JOB_STATUS  VARCHAR2(25) NOT NULL CHECK(EMP_JOB_STATUS  
IN ('WORKING', 'ON LEAVE', 'NO SHIFT', 'OLD EMPLOYEE')),
```

```
EMP_JOINING_DATE DATE NOT NULL,
```

```
EMP_TYPE        VARCHAR2(20) NOT NULL,
```

```
PRIMARY KEY (EMP_ID, EMP_NAME),
```

```
CONSTRAINT EMP_ID_U1 UNIQUE(EMP_ID));
```

```
CREATE TABLE MANAGEMENT_STAFF (
```

```
EMP_ID          VARCHAR2(10) NOT NULL,
```

```
EMP_NAME        VARCHAR2(35) NOT NULL,
```

```
EMP_DEPARTMENT  VARCHAR2(35) NOT NULL,
```

```
WORKING_TENURE    VARCHAR2(15) NOT NULL,  
EMP_DESIGNATION   VARCHAR2(20) NOT NULL,  
PRIMARY KEY (EMP_ID, EMP_DESIGNATION),  
CONSTRAINT EMP_DES UNIQUE(EMP_DESIGNATION),  
FOREIGN KEY (EMP_ID, EMP_NAME) REFERENCES EMPLOYEES(EMP_ID,  
EMP_NAME) ON DELETE SET NULL);
```

```
CREATE TABLE DESIGNATION (  
EMP_DESIGNATION    VARCHAR2(20) NOT NULL,  
EMP_SALARY         INTEGER NOT NULL,  
SALARY_INCREMENT   INTEGER,  
PRIMARY KEY (EMP_DESIGNATION),  
FOREIGN KEY (EMP_DESIGNATION) REFERENCES  
MANAGEMENT_STAFF(EMP_DESIGNATION) ON DELETE SET NULL);
```

```
CREATE TABLE TEACHING_STAFF (  
EMP_ID            VARCHAR2(10) NOT NULL,  
EMP_NAME          VARCHAR2(35) NOT NULL,  
WORKING_TENURE    VARCHAR2(15) NOT NULL,  
EMP_DESIGNATION   VARCHAR2(20) NOT NULL CHECK(EMP_DESIGNATION  
IN ('PROFESSOR', 'ASSISTANT PROFESSOR', 'INTERN')),  
PRIMARY KEY (EMP_ID, EMP_NAME),  
FOREIGN KEY (EMP_ID, EMP_NAME) REFERENCES EMPLOYEES(EMP_ID,  
EMP_NAME) ON DELETE SET NULL);
```

```
CREATE TABLE LECTURES (  
LECT_NAME         VARCHAR2(35) NOT NULL,  
LECT_DURATION     INTEGER,  
LECT_TIMINGS      VARCHAR2(20),  
PRIMARY KEY (LECT_NAME));
```

```
CREATE TABLE PROFESSOR (  

```



```

EMP_ID          VARCHAR2(10) NOT NULL,
EMP_NAME        VARCHAR2(35) NOT NULL,
DEPARTMENT_NAME VARCHAR2(35) NOT NULL
CHECK(DEPARTMENT_NAME IN ('FASHION DESIGN', 'FINANCE', 'HUMAN
RESOURCES', 'INTERNATIONAL BUSINESS', 'IT', 'SALES AND MARKETING')),
LECTURE_NAME     VARCHAR2(35) NOT NULL,
NUM_ASST_PROF    INTEGER NOT NULL,
EMP_SALARY       INTEGER NOT NULL,
QUARTERLY_INCREMENT INTEGER,
PRIMARY KEY (EMP_ID, EMP_NAME),
FOREIGN KEY (EMP_ID, EMP_NAME) REFERENCES
TEACHING_STAFF(EMP_ID, EMP_NAME) ON DELETE SET NULL,
CONSTRAINT PROF_ID UNIQUE(EMP_ID),
FOREIGN KEY (DEPARTMENT_NAME) REFERENCES
DEPARTMENT(DEPARTMENT_NAME) ON DELETE SET NULL,
FOREIGN KEY (LECTURE_NAME) REFERENCES LECTURES(LECT_NAME) ON
DELETE SET NULL);

```

```

CREATE TABLE ASSISTANT_PROFESSOR (
EMP_ID          VARCHAR2(10) NOT NULL,
EMP_NAME        VARCHAR2(35) NOT NULL,
DEPARTMENT_NAME VARCHAR2(35) NOT NULL
CHECK(DEPARTMENT_NAME IN ('FASHION DESIGN', 'FINANCE', 'HUMAN
RESOURCES', 'INTERNATIONAL BUSINESS', 'IT', 'SALES AND MARKETING')),
REPORTS_TO_PROF_ID VARCHAR2(10) NOT NULL,
EMP_SALARY       INTEGER NOT NULL,
YEARLY_INCREMENT INTEGER,
PRIMARY KEY (EMP_ID, EMP_NAME),
FOREIGN KEY (EMP_ID, EMP_NAME) REFERENCES
TEACHING_STAFF(EMP_ID, EMP_NAME) ON DELETE SET NULL,
CONSTRAINT A_PROF_ID UNIQUE(EMP_ID),
FOREIGN KEY (DEPARTMENT_NAME) REFERENCES
DEPARTMENT(DEPARTMENT_NAME) ON DELETE SET NULL,

```

FOREIGN KEY (REPORTS_TO_PROF_ID) REFERENCES PROFESSOR(EMP_ID)
ON DELETE SET NULL);

CREATE TABLE INTERN (

EMP_ID VARCHAR2(10) NOT NULL,

EMP_NAME VARCHAR2(35) NOT NULL,

EMP_SALARY INTEGER NOT NULL,

DEPARTMENT_NAME VARCHAR2(35) NOT NULL

CHECK(DEPARTMENT_NAME IN ('FASHION DESIGN', 'FINANCE', 'HUMAN
RESOURCES', 'INTERNATIONAL BUSINESS', 'IT', 'SALES AND MARKETING')),

REPORTS_TO_ASST_PROF_ID VARCHAR2(10) NOT NULL,

PRIMARY KEY (EMP_ID, EMP_NAME),

FOREIGN KEY (EMP_ID, EMP_NAME) REFERENCES
TEACHING_STAFF(EMP_ID, EMP_NAME) ON DELETE SET NULL,

FOREIGN KEY (DEPARTMENT_NAME) REFERENCES
DEPARTMENT(DEPARTMENT_NAME) ON DELETE SET NULL,

FOREIGN KEY (REPORTS_TO_ASST_PROF_ID) REFERENCES
ASSISTANT_PROFESSOR(EMP_ID) ON DELETE SET NULL);

CREATE TABLE HEAD_OF_DEPARTMENT (

EMP_ID VARCHAR2(10) NOT NULL,

EMP_NAME VARCHAR2(35) NOT NULL,

DEPARTMENT_NAME VARCHAR2(35) NOT NULL

CHECK(DEPARTMENT_NAME IN ('FASHION DESIGN', 'FINANCE', 'HUMAN
RESOURCES', 'INTERNATIONAL BUSINESS', 'IT', 'SALES AND MARKETING')),

WORKING_TENURE VARCHAR2(15) NOT NULL,

EMP_SALARY INTEGER NOT NULL,

QUARTERLY_INCREMENT INTEGER,

NUM_PROF_REPORTING INTEGER NOT NULL,

PRIMARY KEY (EMP_ID, DEPARTMENT_NAME),

FOREIGN KEY (DEPARTMENT_NAME) REFERENCES
DEPARTMENT(DEPARTMENT_NAME) ON DELETE SET NULL);

CREATE TABLE PRINCIPAL (

P_NAME VARCHAR2(35) NOT NULL,
P_DOB DATE NOT NULL,
P_CONTACT CHAR(10) NOT NULL,
WORKING_TENURE VARCHAR2(15) NOT NULL,
PRIMARY KEY (P_NAME));

CREATE TABLE BRANCHES (
BRANCH_NAME VARCHAR2(35) NOT NULL,
LOCATION VARCHAR2(35) NOT NULL,
PRINCIPAL_NAME VARCHAR2(35) NOT NULL,
NUM_COURSES INTEGER NOT NULL,
PRIMARY KEY (BRANCH_NAME),
FOREIGN KEY (PRINCIPAL_NAME) REFERENCES PRINCIPAL(P_NAME) ON
DELETE SET NULL);

CREATE TABLE COURSES (
COURSE_NAME VARCHAR2(25) NOT NULL,
SPECIALISATION_NAME VARCHAR2(35) NOT NULL,
BRANCH_NAME VARCHAR2(35) NOT NULL,
FEES INTEGER NOT NULL,
DURATION VARCHAR2(10) NOT NULL,
PLACEMENT_RATINGS NUMBER,
ELIGIBILITY VARCHAR2(35),
PRIMARY KEY (COURSE_NAME),
FOREIGN KEY (SPECIALISATION_NAME) REFERENCES
DEPARTMENT(DEPARTMENT_NAME) ON DELETE SET NULL,
FOREIGN KEY (BRANCH_NAME) REFERENCES BRANCHES(BRANCH_NAME)
ON DELETE SET NULL);

CREATE TABLE STUDENTS (
STUD_ID VARCHAR2(15) NOT NULL,

```
STUD_FULL_NAME      VARCHAR2(35) NOT NULL,  
STUD_DOB            DATE,  
STUD_ADDRESS        VARCHAR2(35) NOT NULL,  
STUD_COURSE_NAME     VARCHAR2(25) NOT NULL,  
STUD_ACCOMMODATION_TYPE VARCHAR2(10) NOT NULL  
CHECK(STUD_ACCOMMODATION_TYPE IN ('HOSTEL', 'OWN', 'OTHER')),  
STUD_CONTACT        CHAR(10) NOT NULL,  
STUD_EMERGENCY_NUM   CHAR(10) NOT NULL,  
PRIMARY KEY (STUD_ID),  
FOREIGN KEY (STUD_COURSE_NAME) REFERENCES  
COURSES(COURSE_NAME) ON DELETE SET NULL);
```

```
CREATE TABLE STUDENT_LECTURES (  
LECT_NAME          VARCHAR2(35) NOT NULL,  
STUD_ID            VARCHAR2(15) NOT NULL,  
PRIMARY KEY (LECT_NAME, STUD_ID),  
FOREIGN KEY (LECT_NAME) REFERENCES LECTURES(LECT_NAME),  
FOREIGN KEY (STUD_ID) REFERENCES STUDENTS(STUD_ID) ON DELETE  
SET NULL);
```

```
CREATE TABLE SERVICES (  
SERVICE_NAME      VARCHAR2(35) NOT NULL,  
MONTHLY_CHARGES    INTEGER NOT NULL,  
AVAILABILITY_HOURS VARCHAR2(10) NOT NULL,  
ADDITIONAL_COMMENTS VARCHAR2(40),  
PRIMARY KEY (SERVICE_NAME));
```

```
CREATE TABLE BRANCH_SERVICE (  
BRANCH_NAME        VARCHAR2(35) NOT NULL,  
SERVICE_NAME       VARCHAR2(35) NOT NULL,  
PRIMARY KEY (BRANCH_NAME, SERVICE_NAME),
```

FOREIGN KEY (BRANCH_NAME) REFERENCES BRANCHES(BRANCH_NAME)
ON DELETE SET NULL,
FOREIGN KEY (SERVICE_NAME) REFERENCES SERVICES(SERVICE_NAME)
ON DELETE SET NULL);

CREATE TABLE CHAIRMAN (
CM_INITIALS CHAR(3) NOT NULL,
CM_NAME VARCHAR2(35) NOT NULL,
CM_DOB DATE NOT NULL,
CM_ADDRESS VARCHAR2(40) NOT NULL,
CM_CONTACT_NUM CHAR(10) NOT NULL,
PRIMARY KEY (CM_INITIALS, CM_NAME));

CREATE TABLE DIRECTORS (
DIRECTOR_INITIALS CHAR(3) NOT NULL,
DIRECTOR_NAME VARCHAR2(35) NOT NULL,
WORKING_TENURE VARCHAR2(15) NOT NULL,
INCOME INTEGER,
REPORTED_BY VARCHAR(25) NOT NULL CHECK(REPORTED_BY IN
('FINANCE DEPARTMENT', 'MARKETING DEPARTMENT')),
DIRECTOR_TYPE VARCHAR(20) NOT NULL CHECK(DIRECTOR_TYPE IN
('MANAGING DIRECTOR', 'MARKETING DIRECTOR')),
PRIMARY KEY (DIRECTOR_INITIALS, DIRECTOR_NAME));

CREATE TABLE MARKETING_DEPARTMENT (
MONTH_YEAR VARCHAR2(15) NOT NULL,
REPORT_ID CHAR(10) NOT NULL,
ADVERTISING_COSTS INTEGER NOT NULL,
ADVERTISING_PROFITS INTEGER NOT NULL,
PRIMARY KEY (MONTH_YEAR, REPORT_ID),
CONSTRAINT MAR_REP UNIQUE(REPORT_ID));

```
CREATE TABLE FINANCE_DEPARTMENT (  
MONTH_YEAR    VARCHAR2(15) NOT NULL,  
REPORT_ID     CHAR(10) NOT NULL,  
TURN_OVER     INTEGER NOT NULL,  
PENDING_PAYMENT INTEGER NOT NULL,  
LIABILITIES   INTEGER NOT NULL,  
ASSETS        INTEGER NOT NULL,  
TAXES         INTEGER NOT NULL,  
EQUITY        INTEGER NOT NULL,  
PRIMARY KEY (MONTH_YEAR, REPORT_ID),  
CONSTRAINT FIN_REP UNIQUE(REPORT_ID));
```

```
CREATE TABLE HOSTEL (  
MONTH_YEAR    VARCHAR2(15) NOT NULL,  
HOSTEL_NUM    CHAR(3) NOT NULL,  
NUM_ROOM_AVAIL INTEGER NOT NULL,  
ROOM_TYPE     VARCHAR2(10) NOT NULL CHECK(ROOM_TYPE IN  
( 'SINGLE', 'DOUBLE', 'TRIPLE' )),  
FEE_PER_ROOM  INTEGER NOT NULL,  
PRIMARY KEY (MONTH_YEAR, HOSTEL_NUM));
```

```
CREATE TABLE RECEPTION (  
MONTH_YEAR    VARCHAR2(15) NOT NULL,  
REPORT_ID     CHAR(10) NOT NULL,  
NUM_APPLICANTS INTEGER NOT NULL,  
NUM_ENQUIRIES INTEGER NOT NULL,  
NUM_NOTICES   INTEGER NOT NULL,  
PRIMARY KEY (MONTH_YEAR, REPORT_ID),  
CONSTRAINT REC_REP UNIQUE(REPORT_ID));
```

```
CREATE TABLE ACCOUNTS_DEPARTMENT (  

```

```
MONTH_YEAR      VARCHAR2(15) NOT NULL,  
REPORT_ID       CHAR(10) NOT NULL,  
SCHOLARSHIPS    INTEGER NOT NULL,  
EXPENSES        INTEGER NOT NULL,  
SERVICE_CHARGES INTEGER,  
EVENT_TOUR_COSTS INTEGER,  
SALARIES_PAID   INTEGER NOT NULL,  
FEE_RECEIVED    INTEGER NOT NULL,  
PRIMARY KEY (MONTH_YEAR, REPORT_ID),  
CONSTRAINT ACC_REP UNIQUE(REPORT_ID));
```

```
CREATE TABLE APPLICANT_DATA (  
APPLICATION_NUM      CHAR(12) NOT NULL,  
APPLICANT_FULL_NAME  VARCHAR2(35) NOT NULL,  
APPLICANT_ADDRESS    VARCHAR2(40) NOT NULL,  
APPLICANT_DOB        DATE,  
APPLICANT_CONTACT    CHAR(10) NOT NULL,  
APPLICANT_COURSE     VARCHAR2(25) NOT NULL,  
APPLICANT_QUALIFICATIONS VARCHAR2(35) NOT NULL,  
DATE_OF_APPLYING     DATE,  
PRIMARY KEY (APPLICATION_NUM));
```

```
CREATE TABLE ADMIN (  
MONTH_YEAR      VARCHAR2(15) NOT NULL,  
INTERVIEWS_TAKEN  INTEGER NOT NULL,  
NEW_ADMISSIONS    INTEGER NOT NULL,  
REJECTED_APPLICATIONS INTEGER NOT NULL,  
LAST_WEBSITE_UPDATE DATE NOT NULL,  
PRIMARY KEY (MONTH_YEAR));
```

```

CREATE TABLE EVENTS_DEPARTMENT (
MONTH_YEAR          VARCHAR2(15) NOT NULL,
REPORT_ID           CHAR(10) NOT NULL,
ACAD_EVENTS         INTEGER NOT NULL,
EXTRA_EVENTS        INTEGER NOT NULL,
TOTAL_COSTS         INTEGER NOT NULL,
PRIMARY KEY (MONTH_YEAR, REPORT_ID),
CONSTRAINT EVE_REP UNIQUE(REPORT_ID));

```

```

CREATE TABLE MONTHLY_REPORTS (
MONTH_YEAR          VARCHAR2(15) NOT NULL,
FINANCE_REPORT_ID   CHAR(10) NOT NULL,
EVENTS_REPORT_ID    CHAR(10) NOT NULL,
ACC_REPORT_ID       CHAR(10) NOT NULL,
MARKT_REPORT_ID     CHAR(10) NOT NULL,
RECEP_REPORT_ID     CHAR(10) NOT NULL,
PRIMARY KEY (MONTH_YEAR, FINANCE_REPORT_ID, EVENTS_REPORT_ID,
ACC_REPORT_ID, MARKT_REPORT_ID, RECEP_REPORT_ID),
FOREIGN KEY (FINANCE_REPORT_ID) REFERENCES
FINANCE_DEPARTMENT(REPORT_ID) ON DELETE SET NULL,
FOREIGN KEY (EVENTS_REPORT_ID) REFERENCES
EVENTS_DEPARTMENT(REPORT_ID) ON DELETE SET NULL,
FOREIGN KEY (ACC_REPORT_ID) REFERENCES
ACCOUNTS_DEPARTMENT(REPORT_ID) ON DELETE SET NULL,
FOREIGN KEY (MARKT_REPORT_ID) REFERENCES
MARKETING_DEPARTMENT(REPORT_ID) ON DELETE SET NULL,
FOREIGN KEY (RECEP_REPORT_ID) REFERENCES RECEPTION(REPORT_ID)
ON DELETE SET NULL);

```

2.6 INSERTING DATA INTO TABLES:

```

INSERT INTO DEPARTMENT VALUES ('FASHION DESIGN', 'FD01', 2);

```


INSERT INTO DEPARTMENT VALUES ('FINANCE', 'FN02', 1);
INSERT INTO DEPARTMENT VALUES ('HUMAN RESOURCES', 'HR03', 2);
INSERT INTO DEPARTMENT VALUES ('INTERNATIONAL BUSINESS', 'IB04', 1);
INSERT INTO DEPARTMENT VALUES ('IT', 'IT05', 2);
INSERT INTO DEPARTMENT VALUES ('SALES AND MARKETING','SM06', 1);

INSERT INTO EMPLOYEES VALUES ('EMPNO00001', 'Asif Ali', '01-Mar-1994',
'Kurukshetra', '9876543210', 'WORKING', '03-Nov-2018', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00045', 'Baljeet', '23-Dec-1996', 'Karnal',
'8654127956', 'ON LEAVE', '28-May-2017', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00756', 'David Nike', '18-Nov-1993', 'Rawa',
'9784568521', 'NO SHIFT', '26-Oct-2010', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00021', 'Daman', '13-May-1999', 'Karnal',
'7894587458', 'OLD EMPLOYEE', '04-Jun-2019', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00671', 'Florina', '12-Jun-1992', 'Karnal',
'9879879658', 'OLD EMPLOYEE', '21-Apr-2012', 'MANAGEMENT_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00980', 'Firoz Khan', '19-May-1995',
'Kurukshetra', '8546854685', 'NO SHIFT', '08-Jan-2013', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00004', 'Harish Kaur', '26-Sep-1995',
'Karnal', '8523698741', 'WORKING', '19-Nov-2010', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00554', 'Jai', '27-Aug-1994', 'Kurukshetra',
'9512436087', 'OLD EMPLOYEE', '11-Dec-2018', 'MANAGEMENT_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO12564', 'Jaya', '05-Jun-1993', 'Kurukshetra',
'9512435587', 'WORKING', '01-Aug-2019', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00234', 'Kamal', '30-Nov-1996', 'Karnal',
'9002323018', 'WORKING', '03-Aug-2019', 'MANAGEMENT_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00321', 'Kirandeep', '11-Mar-1996', 'Rawa',
'9870650650', 'NO SHIFT', '01-Jul-2016', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO05891', 'Manav Singh', '27-Feb-1999',
'Kurukshetra', '8457623591', 'OLD EMPLOYEE', '09-Jan-2013', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO09051', 'Mehak', '14-Jan-1998', 'Karnal',
'8965236523', 'ON LEAVE', '16-Oct-2019', 'MANAGEMENT_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO00501', 'Neeraj', '11-Oct-2000',
'Kurukshetra', '9876549870', 'WORKING', '18-Sep-2020', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO27001', 'Oliver', '22-Jul-2001', 'Kurukshetra',
'9856985698', 'WORKING', '10-Aug-2014', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO04401', 'Paras', '26-Nov-1993', 'Karnal',
'7854789654', 'WORKING', '30-Aug-2017', 'MANAGEMENT_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO09801', 'Paul', '17-Dec-1994', 'Rawa', '9632541087', 'NO SHIFT', '13-Apr-2020', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO30701', 'Rhea', '09-Jun-1996', 'Kurukshetra', '9874152630', 'ON LEAVE', '06-Jun-2011', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO33001', 'Sammy', '05-Mar-1997', 'Karnal', '8005400630', 'WORKING', '15-Jul-2017', 'TEACHING_STAFF');

INSERT INTO EMPLOYEES VALUES ('EMPNO49501', 'Stuart', '31-Jan-1997', 'Rawa', '9112053306', 'WORKING', '31-Dec-2013', 'TEACHING_STAFF');

INSERT INTO MANAGEMENT_STAFF VALUES ('EMPNO00671', 'Florina', 'Cleaning', '2012-2018', 'Janitor');

INSERT INTO MANAGEMENT_STAFF VALUES ('EMPNO00554', 'Jai', 'Reception', '2018-2020', 'Cashier');

INSERT INTO MANAGEMENT_STAFF VALUES ('EMPNO00234', 'Kamal', 'Events_Department', '2019-CURRENT', 'Events_Manager');

INSERT INTO MANAGEMENT_STAFF VALUES ('EMPNO09051', 'Mehak', 'Hostel', '2019-CURRENT', 'Warden');

INSERT INTO MANAGEMENT_STAFF VALUES ('EMPNO04401', 'Paras', 'Marketing and Website', '2017-CURRENT', 'Programmer');

INSERT INTO DESIGNATION VALUES ('Janitor', 10000, 1200);

INSERT INTO DESIGNATION VALUES ('Cashier', 12000, 1000);

INSERT INTO DESIGNATION VALUES ('Events_Manager', 17000, 1300);

INSERT INTO DESIGNATION VALUES ('Warden', 10000, 1000);

INSERT INTO DESIGNATION VALUES ('Programmer', 20000, 1700);

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00001', 'Asif Ali', '2018-CURRENT', 'ASSISTANT PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00045', 'Baljeet', '2017-CURRENT', 'ASSISTANT PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00756', 'David Nike', '2010-CURRENT', 'PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00021', 'Daman', '2019-2020', 'INTERN');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00980', 'Firoz Khan', '2013-CURRENT', 'PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00004', 'Harish Kaur', '2010-CURRENT', 'PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00321', 'Kirandeep', '2016-CURRENT', 'ASSISTANT PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO05891', 'Manav Singh', '2013-2016', 'INTERN');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO00501', 'Neeraj', '2020-CURRENT', 'INTERN');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO27001', 'Oliver', '2014-CURRENT', 'ASSISTANT PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO09801', 'Paul', '2020-CURRENT', 'INTERN');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO30701', 'Rhea', '2011-CURRENT', 'PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO33001', 'Sammy', '2017-CURRENT', 'ASSISTANT PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO49501', 'Stuart', '2013-CURRENT', 'PROFESSOR');

INSERT INTO TEACHING_STAFF VALUES ('EMPNO12564', 'Jaya', '2019-CURRENT', 'INTERN');

INSERT INTO LECTURES VALUES ('TECHNIQUES OF MONEY MANAGEMENT', 2, '9 TO 11 AM');

INSERT INTO LECTURES VALUES ('REUSE OF WASTE RESOURCES', 2, '1 TO 3 PM');

INSERT INTO LECTURES VALUES ('INCOME TARGETS AND TRICKS', 1, '8 TO 9 AM');

INSERT INTO LECTURES VALUES ('ART OF ANCIENT WORKS', 3, '11.30 AM TO 2.30 PM');

INSERT INTO LECTURES VALUES ('FOUNDATION OF IT', 2, '4 TO 6 PM');

INSERT INTO PROFESSOR VALUES ('EMPNO00756', 'David Nike', 'FINANCE', 'TECHNIQUES OF MONEY MANAGEMENT', 5, 60000, 1500);

INSERT INTO PROFESSOR VALUES ('EMPNO00980', 'Firoz Khan', 'HUMAN RESOURCES', 'REUSE OF WASTE RESOURCES', 4, 55000, 1300);

INSERT INTO PROFESSOR VALUES ('EMPNO00004', 'Harish Kaur', 'FINANCE', 'INCOME TARGETS AND TRICKS', 6, 60000, 1500);

INSERT INTO PROFESSOR VALUES ('EMPNO30701', 'Rhea', 'FASHION DESIGN', 'ART OF ANCIENT WORKS', 3, 63000, 1700);

INSERT INTO PROFESSOR VALUES ('EMPNO49501', 'Stuart', 'IT', 'FOUNDATION OF IT', 5, 70000, 2000);

INSERT INTO ASSISTANT_PROFESSOR VALUES ('EMPNO00001', 'Asif Ali', 'FASHION DESIGN', 'EMPNO30701', 30000, 1500);

INSERT INTO ASSISTANT_PROFESSOR VALUES ('EMPNO00045', 'Baljeet', 'HUMAN RESOURCES', 'EMPNO00980', 29000, 1300);

INSERT INTO ASSISTANT_PROFESSOR VALUES ('EMPNO00321', 'Kirandeep', 'FINANCE', 'EMPNO00004', 32000, 1700);

INSERT INTO ASSISTANT_PROFESSOR VALUES ('EMPNO27001', 'Oliver', 'FASHION DESIGN', 'EMPNO30701', 30000, 1500);

INSERT INTO ASSISTANT_PROFESSOR VALUES ('EMPNO33001', 'Sammy', 'IT', 'EMPNO49501', 35000, 2000);

INSERT INTO INTERN VALUES ('EMPNO00021', 'Daman', 5000, 'FASHION DESIGN', 'EMPNO00001');

INSERT INTO INTERN VALUES ('EMPNO05891', 'Manav Singh', 4000, 'FINANCE', 'EMPNO00321');

INSERT INTO INTERN VALUES ('EMPNO00501', 'Neeraj', 5000, 'FASHION DESIGN', 'EMPNO27001');

INSERT INTO INTERN VALUES ('EMPNO09801', 'Paul', 6000, 'IT', 'EMPNO33001');

INSERT INTO INTERN VALUES ('EMPNO12564', 'Jaya', 6000, 'IT', 'EMPNO33001');

INSERT INTO HEAD_OF_DEPARTMENT VALUES ('EMPNO78945', 'RAVI', 'FASHION DESIGN', '2010-CURRENT', 130000, 3000, 8);

INSERT INTO HEAD_OF_DEPARTMENT VALUES ('EMPNO96854', 'SHINKHA', 'HUMAN RESOURCES', '2009-CURRENT', 115000, 2600, 15);

INSERT INTO HEAD_OF_DEPARTMENT VALUES ('EMPNO06665', 'TAMANNA', 'FINANCE', '2013-CURRENT', 140000, 3400, 15);

INSERT INTO HEAD_OF_DEPARTMENT VALUES ('EMPNO27056', 'YASH', 'FASHION DESIGN', '2008-CURRENT', 130000, 3000, 8);

INSERT INTO HEAD_OF_DEPARTMENT VALUES ('EMPNO33081', 'ZAKIR', 'IT', '2011-CURRENT', 150000, 5000, 13);

INSERT INTO PRINCIPAL VALUES ('JANE PAULO', '11-Jul-1985', '9865986325', '2008-CURRENT');

INSERT INTO PRINCIPAL VALUES ('DEAN TOOR', '16-Nov-1979', '9857412360', '2015-CURRENT');

INSERT INTO PRINCIPAL VALUES ('HUGH BANSAL', '28-Jun-1981', '9758496200', '2011-CURRENT');

INSERT INTO PRINCIPAL VALUES ('LOVELY SINGH', '30-Mar-1988', '8000321605', '2019-CURRENT');

INSERT INTO BRANCHES VALUES ('BCHE', 'KARNAL, SECTOR 9', 'JANE PAULO', 4);

INSERT INTO BRANCHES VALUES ('BCE', 'KARNAL, SECTOR 8', 'DEAN TOOR', 2);

INSERT INTO BRANCHES VALUES ('BCA', 'KARNAL, SECTOR 12', 'HUGH BANSAL', 1);

INSERT INTO BRANCHES VALUES ('BCM', 'KARNAL, SECTOR 13', 'LOVELY SINGH', 2);

INSERT INTO COURSES VALUES ('MBA', 'FINANCE', 'BCM', 300000, '2 YEARS', 3.8, 'CLASS 12 OR EQUIVALENT');

INSERT INTO COURSES VALUES ('B.COM', 'SALES AND MARKETING', 'BCHE', 200000, '3 YEARS', 4.5, 'CLASS 12 OR EQUIVALENT');

INSERT INTO COURSES VALUES ('B.SC', 'FASHION DESIGN', 'BCHE', 400000, '4 YEARS', 3.3, 'CLASS 12 OR EQUIVALENT');

INSERT INTO COURSES VALUES ('BFA', 'HUMAN RESOURCES', 'BCHE', 300000, '3 YEARS', 4.9, 'CLASS 12 OR EQUIVALENT');

INSERT INTO COURSES VALUES ('B.ARCH', 'HUMAN RESOURCES', 'BCA', 500000, '5 YEARS', 3.6, 'CLASS 12 OR EQUIVALENT');

INSERT INTO COURSES VALUES ('B.ED', 'IT', 'BCE', 400000, '3 YEARS', 4.2, 'CLASS 12 OR EQUIVALENT');

INSERT INTO STUDENTS VALUES ('STU_FIN_00001', 'AIYSHA', '04-May2002', 'KARNAL', 'MBA', 'HOSTEL', '9865986532', '8798795462');

INSERT INTO STUDENTS VALUES ('STU_HRS_00001', 'JANISHA', '11-Aug-2001', 'RAWA', 'B.ARCH', 'OWN', '8794563210', '8965230145');

INSERT INTO STUDENTS VALUES ('STU_FIN_00012', 'KAVITA', '26-Dec-2004', 'KKR', 'MBA', 'HOSTEL', '7878784545', '7894586320');

INSERT INTO STUDENTS VALUES ('STU_FDS_00004', 'LAKSH', '20-Jul-1999', 'DELHI', 'B.SC', 'OTHER', '7485962301', '9003690036');

INSERT INTO STUDENTS VALUES ('STU_ITD_00031', 'PARUL', '15-Mar-1997', 'RAWA', 'B.ED', 'HOSTEL', '8794856123', '9807465321');

INSERT INTO STUDENT_LECTURES VALUES ('TECHNIQUES OF MONEY MANAGEMENT', 'STU_FIN_00001');

INSERT INTO STUDENT_LECTURES VALUES ('REUSE OF WASTE RESOURCES', 'STU_HRS_00001');

INSERT INTO STUDENT_LECTURES VALUES ('INCOME TARGETS AND TRICKS', 'STU_FIN_00012');

INSERT INTO STUDENT_LECTURES VALUES ('ART OF ANCIENT WORKS',
'STU_FDS_00004');

INSERT INTO STUDENT_LECTURES VALUES ('FOUNDATION OF IT','STU_ITD_00031');

INSERT INTO SERVICES VALUES ('GYM', 500, '5AM TO 11PM', 'SEPARATE FOR BOYS
AND GIRLS');

INSERT INTO SERVICES VALUES ('SPORTS', 1000, '9AM TO 7PM', 'FREE EQUIPMENT
AVAILABLE');

INSERT INTO SERVICES VALUES ('YOGA', 800, '6AM TO 9AM', 'BRETHE FRESH AND
STAY HEALTHY');

INSERT INTO SERVICES VALUES ('CAFE', 300, '24HOURS', 'TAX-FREE COFFEE
ANYTIME');

INSERT INTO SERVICES VALUES ('CANTEEN', 450, '24HOURS', 'COME AND GRAB
SOME FOOD');

INSERT INTO BRANCH_SERVICE VALUES ('BCHE', 'CANTEEN');

INSERT INTO BRANCH_SERVICE VALUES ('BCA', 'CANTEEN');

INSERT INTO BRANCH_SERVICE VALUES ('BCM', 'SPORTS');

INSERT INTO BRANCH_SERVICE VALUES ('BCHE', 'CAFE');

INSERT INTO BRANCH_SERVICE VALUES ('BCE', 'CAFE');

INSERT INTO CHAIRMAN VALUES ('LKR', 'LALA KHUSHI RAM','22-Feb-1965',
'LUDHIANA', '6523987001');

INSERT INTO CHAIRMAN VALUES ('RMG', 'RAM MOHAN GUPTA', '22-Jun-1972',
'PAKHOWAL', '8659653207');

INSERT INTO DIRECTORS VALUES ('LKS', 'LAAL KHUSH SINGH', 'FOUNDATION-2000',
200000, 'FINANCE DEPARTMENT', 'MANAGING DIRECTOR');

INSERT INTO DIRECTORS VALUES ('RMG', 'RAM MOHAN GUPTA', 'FOUNDATION-
2000', 400000, 'MARKETING DEPARTMENT', 'MARKETING DIRECTOR');

INSERT INTO DIRECTORS VALUES ('NTG', 'NITESH GUPTA', '2000-CURRENT',
1000000, 'FINANCE DEPARTMENT', 'MANAGING DIRECTOR');

INSERT INTO DIRECTORS VALUES ('MKG', 'MEENAKSHI GUPTA', '2000-CURRENT',
1000000, 'FINANCE DEPARTMENT', 'MANAGING DIRECTOR');

INSERT INTO DIRECTORS VALUES ('ASG', 'ASHISH GUPTA', '2002-CURRENT', 800000,
'MARKETING DEPARTMENT', 'MARKETING DIRECTOR');

INSERT INTO MARKETING_DEPARTMENT VALUES ('APR-21', 'MKT_REP01', 20000, 55000);

INSERT INTO MARKETING_DEPARTMENT VALUES ('MAY-21', 'MKT_REP02', 30000, 40000);

INSERT INTO MARKETING_DEPARTMENT VALUES ('AUG-21', 'MKT_REP03', 10000, 60000);

INSERT INTO MARKETING_DEPARTMENT VALUES ('SEP-21', 'MKT_REP04', 10000, 50000);

INSERT INTO MARKETING_DEPARTMENT VALUES ('OCT-21', 'MKT_REP05', 29000, 50000);

INSERT INTO FINANCE_DEPARTMENT VALUES ('APR-21', 'FIN_REP01', 20000, 55000, 50000, 56000, 300000, 500000);

INSERT INTO FINANCE_DEPARTMENT VALUES ('MAY-21', 'FIN_REP02', 30000, 40000, 70000, 46000, 600000, 500000);

INSERT INTO FINANCE_DEPARTMENT VALUES ('AUG-21', 'FIN_REP03', 100000, 60000, 30000, 73000, 300000, 500000);

INSERT INTO FINANCE_DEPARTMENT VALUES ('SEP-21', 'FIN_REP04', 100000, 50000, 50000, 50000, 600000, 500000);

INSERT INTO FINANCE_DEPARTMENT VALUES ('OCT-21', 'FIN_REP05', 29000, 30000, 56000, 60000, 300000, 500000);

INSERT INTO HOSTEL VALUES ('APR-21', 'A01', 10, 'SINGLE', 6000);

INSERT INTO HOSTEL VALUES ('MAY-21', 'B03', 8, 'DOUBLE', 4500);

INSERT INTO HOSTEL VALUES ('AUG-21', 'C02', 7, 'DOUBLE', 4000);

INSERT INTO HOSTEL VALUES ('SEP-21', 'A02', 15, 'TRIPLE', 4000);

INSERT INTO HOSTEL VALUES ('OCT-21', 'B01', 4, 'SINGLE', 5500);

INSERT INTO RECEPTION VALUES ('APR-21', 'REC_REP01', 100, 300, 50);

INSERT INTO RECEPTION VALUES ('MAY-21', 'REC_REP02', 75, 200, 70);

INSERT INTO RECEPTION VALUES ('AUG-21', 'REC_REP03', 70, 100, 30);

INSERT INTO RECEPTION VALUES ('SEP-21', 'REC_REP04', 150, 130, 20);

INSERT INTO RECEPTION VALUES ('OCT-21', 'REC_REP05', 40, 120, 56);

INSERT INTO ACCOUNTS_DEPARTMENT VALUES ('APR-21', 'ACC_REP01', 20000, 55000, 50000, 56000, 300000, 500000);

INSERT INTO ACCOUNTS_DEPARTMENT VALUES ('MAY-21', 'ACC_REP02', 30000, 40000, 70000, 46000, 600000, 500000);

INSERT INTO ACCOUNTS_DEPARTMENT VALUES ('AUG-21', 'ACC_REP03', 100000, 60000, 30000, 73000, 300000, 500000);

INSERT INTO ACCOUNTS_DEPARTMENT VALUES ('SEP-21', 'ACC_REP04', 100000, 50000, 50000, 50000, 600000, 500000);

INSERT INTO ACCOUNTS_DEPARTMENT VALUES ('OCT-21', 'ACC_REP05', 29000, 30000, 56000, 60000, 300000, 500000);

INSERT INTO APPLICANT_DATA VALUES ('APP_FIN_010','AIANA', 'KARNAL', '02-Dec-2002', '9865986532', 'MBA', 'CLASS 12', '28-May-2021');

INSERT INTO APPLICANT_DATA VALUES ('APP_HRS_001','JATIN', 'RAWA', '12-Apr-2001', '8794563210', 'B.ARCH', 'CLASS 12', '25-May-2021');

INSERT INTO APPLICANT_DATA VALUES ('APP_FIN_012','KOMAL', 'KKR', '28-Mar-2004', '7878784545', 'MBA', 'CLASS 10', '27-May-2021');

INSERT INTO APPLICANT_DATA VALUES ('APP_FDS_004','LAVISH', 'DELHI', '12-Jul-1999', '7485962301', 'B.SC', 'CLASS 10', '12-May-2021');

INSERT INTO APPLICANT_DATA VALUES ('APP_ITD_031','SITA', 'RAWA', '05-Aug-1997', '8794856123', 'B.ED', 'CLASS 12', '15-May-2021');

INSERT INTO ADMIN VALUES ('APR-21', 100, 300, 50, '02-May-2021');

INSERT INTO ADMIN VALUES ('MAY-21', 75, 200, 70, '02-Jun-2021');

INSERT INTO ADMIN VALUES ('AUG-21', 70, 100, 30, '02-Jul-2021');

INSERT INTO ADMIN VALUES ('SEP-21', 150, 130, 20, '02-Aug-2021');

INSERT INTO ADMIN VALUES ('OCT-21', 40, 120, 56, '02-Sep-2021');

INSERT INTO EVENTS_DEPARTMENT VALUES ('APR-21', 'EVE_REP01', 2, 5, 50000);

INSERT INTO EVENTS_DEPARTMENT VALUES ('MAY-21', 'EVE_REP02', 3, 4, 70000);

INSERT INTO EVENTS_DEPARTMENT VALUES ('AUG-21', 'EVE_REP03', 1, 6, 30000);

INSERT INTO EVENTS_DEPARTMENT VALUES ('SEP-21', 'EVE_REP04', 1, 5, 50000);

INSERT INTO EVENTS_DEPARTMENT VALUES ('OCT-21', 'EVE_REP05', 2, 3, 56000);

INSERT INTO MONTHLY_REPORTS VALUES ('APR-21', 'FIN_REP01', 'EVE_REP01', 'ACC_REP01', 'MKT_REP01', 'REC_REP01');

INSERT INTO MONTHLY_REPORTS VALUES ('MAY-21', 'FIN_REP02', 'EVE_REP02', 'ACC_REP02', 'MKT_REP02', 'REC_REP02');


```
INSERT INTO MONTHLY_REPORTS VALUES ('AUG-21', 'FIN_REP03', 'EVE_REP03',  
'ACC_REP03', 'MKT_REP03', 'REC_REP03');
```

```
INSERT INTO MONTHLY_REPORTS VALUES ('SEP-21', 'FIN_REP04', 'EVE_REP04',  
'ACC_REP04', 'MKT_REP04', 'REC_REP04');
```

```
INSERT INTO MONTHLY_REPORTS VALUES ('OCT-21', 'FIN_REP05', 'EVE_REP05',  
'ACC_REP05', 'MKT_REP05', 'REC_REP05');
```

2.7 ALTER AND UPDATE TABLE:

a) Name of the Table – EMPLOYEES

Name of the Column – Emergency Contact

data type - CHAR, size = 10

default value - 1234567890

SQL QUERY:

```
ALTER TABLE EMPLOYEES ADD (EMERGENCY_CONTACT CHAR(10)  
DEFAULT '1234567890');
```

SCREENSHOTS:

```
SQL> ALTER TABLE EMPLOYEES ADD (EMERGENCY_CONTACT CHAR(10) DEFAULT '1234567890')  
;  
Table altered.
```

EMP_ID	EMP_NAME	EMP_DOB
EMP_ADDRESS	EMP_CONTAC	EMP_JOB_STATUS
EMP_JOINI	EMP_TYPE	EMERGENCY
Karnal 16-OCT-19	MANAGEMENT_STAFF	1234567890
EMPNO00501 Neeraj		8965236523 ON LEAVE
Kurukshetra 18-SEP-20	TEACHING_STAFF	1234567890
		11-OCT-00 9876549870 WORKING
EMP_ID	EMP_NAME	EMP_DOB
EMP_ADDRESS	EMP_CONTAC	EMP_JOB_STATUS
EMP_JOINI	EMP_TYPE	EMERGENCY
EMPNO27001 Oliver		22-JUL-01
Kurukshetra 10-AUG-14	TEACHING_STAFF	1234567890
		9856985698 WORKING
EMPNO004401 Paras		26-NOV-93
Karnal 30-AUG-17	MANAGEMENT_STAFF	1234567890
		7854789654 WORKING
EMP_ID	EMP_NAME	EMP_DOB
EMP_ADDRESS	EMP_CONTAC	EMP_JOB_STATUS
EMP_JOINI	EMP_TYPE	EMERGENCY
EMPNO009801 Paul		17-DEC-94
Rawa 13-APR-20	TEACHING_STAFF	1234567890
		9632541087 NO SHIFT
EMPNO30701 Rhea		09-JUN-96
Kurukshetra		9874152630 ON LEAVE
EMP_ID	EMP_NAME	EMP_DOB
EMP_ADDRESS	EMP_CONTAC	EMP_JOB_STATUS
EMP_JOINI	EMP_TYPE	EMERGENCY
06-JUN-11	TEACHING_STAFF	1234567890
EMPNO33001 Sammy		05-MAR-97
Karnal 15-JUL-17	TEACHING_STAFF	1234567890
		8005400630 WORKING
EMPNO49501 Stuart		31-JAN-97
EMP_ID	EMP_NAME	EMP_DOB
EMP_ADDRESS	EMP_CONTAC	EMP_JOB_STATUS
EMP_JOINI	EMP_TYPE	EMERGENCY
Rawa 31-DEC-13	TEACHING_STAFF	1234567890
		9112053306 WORKING

20 rows selected.

b) Condition – EMP_JOB_STATUS = 'WORKING'

New Value of EMERGENCY CONTACT = 9876543210

SQL QUERY:

```
UPDATE EMPLOYEES SET EMERGENCY_CONTACT = 9876543210 WHERE  
EMP_JOB_STATUS = 'WORKING';
```

SCREENSHOTS:

```
SQL> UPDATE EMPLOYEES SET EMERGENCY_CONTACT = 9876543210 WHERE EMP_JOB_STATUS = 'WORKING';  
9 rows updated.
```

```
SQL> SELECT EMERGENCY_CONTACT, EMP_JOB_STATUS FROM EMPLOYEES;  
EMERGENCY_CONTACT  EMP_JOB_STATUS  
-----  
9876543210 WORKING  
1234567890 ON LEAVE  
1234567890 NO SHIFT  
1234567890 OLD EMPLOYEE  
1234567890 OLD EMPLOYEE  
1234567890 NO SHIFT  
9876543210 WORKING  
1234567890 OLD EMPLOYEE  
9876543210 WORKING  
9876543210 WORKING  
1234567890 NO SHIFT  
EMERGENCY_CONTACT  EMP_JOB_STATUS  
-----  
1234567890 OLD EMPLOYEE  
1234567890 ON LEAVE  
9876543210 WORKING  
9876543210 WORKING  
9876543210 WORKING  
1234567890 NO SHIFT  
1234567890 ON LEAVE  
9876543210 WORKING  
9876543210 WORKING  
20 rows selected.
```

2.8 DATA MANIPULATION USING SQL COMMANDS:

- 2.8.1 List Assistant Professor salary and name from Assistant Professor table, merged with Intern salary and Name from Intern table where Assistant Professor salary is more than Intern salary and ordered by Assistant Professor salary.

SQL QUERY:

```
SELECT ASSISTANT_PROFESSOR.EMP_NAME, ASSISTANT_PROFESSOR.EMP_SALARY,  
INTERN.EMP_NAME, INTERN.EMP_SALARY FROM ASSISTANT_PROFESSOR INNER  
JOIN INTERN ON ASSISTANT_PROFESSOR.EMP_SALARY > INTERN.EMP_SALARY  
ORDER BY ASSISTANT_PROFESSOR.EMP_SALARY;
```

SCREENSHOT:

Asif Ali	30000
Jaya	6000
EMP_NAME	EMP_SALARY
EMP_NAME	EMP_SALARY
Kirandeep	32000
Paul	6000
Kirandeep	32000
Daman	5000
Kirandeep	32000
Neeraj	5000
EMP_NAME	EMP_SALARY
EMP_NAME	EMP_SALARY
Kirandeep	32000
Manav Singh	4000
Kirandeep	32000
Jaya	6000
Sammy	35000
Neeraj	5000
EMP_NAME	EMP_SALARY
EMP_NAME	EMP_SALARY
Sammy	35000
Manav Singh	4000
Sammy	35000
Daman	5000
Sammy	35000
Paul	6000
EMP_NAME	EMP_SALARY
EMP_NAME	EMP_SALARY
Sammy	35000
Jaya	6000
25 rows selected.	

2.8.2 List interns whose salary is in (4500, 5000)

SQL QUERY:

SELECT * FROM INTERN WHERE EMP_SALARY IN (4500, 5000);

SCREENSHOT:

```
SQL> SELECT * FROM INTERN WHERE EMP_SALARY IN (4500, 5000);
```

EMP_ID	EMP_NAME	EMP_SALARY
DEPARTMENT_NAME	REPORTS_TO	
EMPN000021 Daman		5000
FASHION DESIGN	EMPN000001	
EMPN000501 Neeraj		5000
FASHION DESIGN	EMPN027001	

- 2.8.3 List the employee name, employee type and student name if the address is same using Full(Outer) Join

SQL QUERY:

```
SELECT EMPLOYEES.EMP_NAME, EMPLOYEES.EMP_TYPE,  
STUDENTS.STUD_FULL_NAME FROM EMPLOYEES FULL JOIN STUDENTS ON  
EMPLOYEES.EMP_ADDRESS = STUDENTS.STUD_ADDRESS;
```

SCREENSHOT:

Mehak	MANAGEMENT_STAFF
Neeraj	TEACHING_STAFF
Oliver	TEACHING_STAFF
EMP_NAME	EMP_TYPE
STUD_FULL_NAME	
Paras	MANAGEMENT_STAFF
Paul	TEACHING_STAFF
Rhea	TEACHING_STAFF
EMP_NAME	EMP_TYPE
STUD_FULL_NAME	
Sammy	TEACHING_STAFF
Stuart	TEACHING_STAFF
KAVITA	
EMP_NAME	EMP_TYPE
STUD_FULL_NAME	
LAKSH	
JANISHA	
PARUL	
EMP_NAME	EMP_TYPE
STUD_FULL_NAME	
AIYSHA	
25 rows selected.	
SQL> □	

- 2.8.4 Using System Date function to calculate working days of employees using their joining date.

SQL QUERY:

```
SELECT EMP_NAME, ROUND((SYSDATE - EMP_JOINING_DATE), 0) AS  
"WORKING_DAYS" FROM EMPLOYEES;
```

SCREENSHOTS:

```
SQL> SELECT EMP_NAME, ROUND((SYSDATE - EMP_JOINING_DATE), 0) AS "WORKING_DAYS" FROM EMPLOYEES;  
EMP_NAME                                WORKING_DAYS  
-----                                -  
Asif Ali                                1075  
Baljeet                                1599  
David Nike                              4005  
Daman                                   862  
Florina                                 3462  
Firoz Khan                              3200  
Harish Kaur                             3981  
Jai                                     1037  
Jaya                                    804  
Kamal                                   802  
Kirandeep                              1930  
EMP_NAME                                WORKING_DAYS  
-----                                -  
Manav Singh                             3199  
Mehak                                   728  
Neeraj                                  390  
Oliver                                 2621  
Paras                                  1505  
Paul                                   548  
Rhea                                   3782  
Sammy                                  1551  
Stuart                                 2843  
20 rows selected.  
SQL>
```

- 2.8.5 A view which shows name of the person whose initials are a part of both director and chairman's initials list.

SQL QUERY:

```
CREATE VIEW VIEW01 AS SELECT CHAIRMAN.CM_INITIALS,  
DIRECTORS.DIRECTOR_INITIALS, CHAIRMAN.CM_NAME FROM CHAIRMAN INNER  
JOIN DIRECTORS ON CHAIRMAN.CM_INITIALS = DIRECTORS.DIRECTOR_INITIALS;
```

SCREENSHOTS:

```
SQL> CREATE VIEW VIEW01 AS SELECT CHAIRMAN.CM_INITIALS, DIRECTOR.DIRECTOR_INITIALS, CHAIRMAN.CM_NAME FROM CHAIRMAN INNER  
JOIN DIRECTORS ON CHAIRMAN.CM_INITIALS = DIRECTORS.DIRECTOR_INITIALS;  
View created.  
SQL> SELECT * FROM VIEW01;  
CM_  DIR CM_NAME  
-----  
RMG  RMG RAM MOHAN GUPTA  
SQL>
```

2.9 DATA SECURITY AND PRIVACY:

Looking in a broader way, data security and privacy are both a part of data protection. With the introduction of great technology, we have already shifted our dependency to computer systems and networks to this extent that even the protection is now a duty that system must follow. In the current situations of work from home, people are completely relied on online transmission of data, whether it is for knowledge, some confidential or any personal data. This makes us more prone to the cybercrimes unethical breaches that happen leading to damages. There are people called attackers, or hackers who breach networks and steal the information they desire and then use it for illegal, and personal gains. This can not be avoided completely but we can take our precautions to reduce them.

Let's first go thru the cybercrimes that can harm an educational institutional, like this college:

- Stolen data from the servers might contain student or employee's personal information, which can be used for malicious purposes even by the people involved in this institution. Thus, access must be restricted for different authorities.
- Usage of faulty hardware like hard-drives, pen-drives is equally dangerous.
- Computer viruses without any backups is a huge loss for any organisation.
- Intentional system hacking and gaining access to restricted content in a completely wrong manner can become very dangerous.

To avoid these things, basic measures that must be followed include:

- Preventing Social Engineering (*Hurix, 2021*)
At this stage it's easy to be trapped in trojan viruses hidden in phishing mails. WE must be sensible enough before we open them so as to not to welcome a virus into our computer.
- Efficient Data Storage and Transmission Techniques
Leaving behind old stacks of paperwork, we must now shift to digitalised security measures that involve:
 - Encryption: Encoding information into a non-common language but a decodable one.
 - Authentication: A safe way of matching credentials, that people other than you are not supposed to know.
 - Data Loss Prevention: It is a practice that we all must follow by preparing back-ups, copies that can be retrieved in case of data loss.
 - Legislation: It means to consider some effective measures that can be used for data protection as legal, irrespective of the fact that they are illegal or not.
 - Threat Monitoring: There are quite advanced systems for the purpose that universities must get installed in cast there are signs of data breach even after good level of protection.

Other than these digital securities, following points must be taken care of:

- i. Physical Security: The server of our database should always be in an environment, which is secure, and climate controlled. Cloud is a modern concept of online storage and cloud servers are considered quite safe in terms of the purpose they fulfil.
- ii. Database software security: Software being used in our system should always be latest, leaving no chance of data leaks or malfunctions.
- iii. Application security: It is equally important to continuously test the safe working of the interface as well, through which we are interacting to the database.
- iv. Device Security: Make sure that the system is in right hands and no dangers can be avoided if a wrong person catches the physical hold of the system. *(IBM Cloud Education, 2019)*

2.10 DATA SECURITY AND PRIVACY:

As we inculcate the use of modern technologies in our lives, a lot of data processing, storing, and transmitting comes with them. Data has become an essential part of lives but when it becomes impossible for our traditional tools to handle some data, we say it as the 'Big Data'. Big data comes with three main characteristics if we generally consider it:

- VELOCITY – Batch, real-time, stream processing
More data generally means more processing time, which would mean excess amounts of data can easily crash our traditional systems, but there is a limit before that which is very useful to us in terms of managing it. We can not only manage the data efficiently with good outputs to a certain extent but also draw amazing conclusions that lead to developments in modern science.
- VOLUME – Terabytes of data, Billions of Records
Considering the data for every millisecond that a machine can create, it is not hard enough to have petabytes of data and no tool to store it. We have been encountering terabytes in our lives, but when we look deeply in this vast field, there is an infinite volume of data that needs to be stored, and that is what we call bigdata.
- VARIETY – Structured, Unstructured, Semi-structured data
It matters to a great amount while accessing data that how it is stored. Even very small data stored poorly in an unorganised will make its processing ages long, but a great amount of data stored using efficient systems such as DBMS at small level helps in quick data accessing and processing.

Let's consider few points about how big data has productive applications in the education sector.

- Enhanced Performance: Creating a network of data gives better ways of learnings to the students, thus enhancing their performance. An interactive study way as they get quick response from the technology they are using. The credit here is to good structure of material, which surely improves efficiency.

- Grading System: A fair grading system based on student's responses and question's difficulty level can be created giving everyone a good chance to outshine. Fair grading is an essential component for evaluation of what has been learned.
- Attentive classes: Using sensor-based technology, a student's reactions, heartbeat, pulse, and temperature can show the level of interest shown by him in the class without even himself knowing about it. It gives a chance to teachers to bring these children up by more focus and hard work.
- Interactive sessions: Awesome programs, hardware and software technology automatically draws interest of students when it is considered as fun rather than an assignment that needs to be marked.

REFERENCES:

- Importance of Data Security in Higher Ed Institutions – By Hurix – 2 February 2021 – Accessed on October 7, 2021
<https://www.hurix.com/data-security-in-higher-ed-institutions/>
- Database Security – IBM Cloud Education – 27 August 2019 – Accessed on October 7, 2021
<https://www.ibm.com/in-en/cloud/learn/database-security>
- What is Big Data? A Quick Introduction for Analytics and Data Engineering Beginners – Siddharth Sonkar – 25 November 2020 – Accessed on October 4, 2021
<https://www.analyticsvidhya.com/blog/2020/11/what-is-big-data-a-quick-introduction-for-analytics-and-data-engineering-beginners/>
- Top 5 Interesting Big Data Applications in Education – By Rohit Sharma – 6 January 2021 – Accessed on October 3, 2021
<https://www.upgrad.com/blog/big-data-applications-in-education/>