Indian Institute of Information Technology, Nagpur

Registration and Feedback System

Database and Management System

Group 5

Computer Science & Engineering

5th Semester

Prepared By:

Raj Aryan (BT19CSE043) Prateek Shende (BT19CSE051)

Sarthak Gupta (BT19CSE054)

PART I

FEEDBACK SYSTEM

```
CREATE TABLE STUDENT_DETAILS(
    STUDENT_ID INTEGER PRIMARY KEY,
    ACADEMIC_YEAR INTEGER NOT NULL,
    SEMESTER INTEGER NOT NULL,
    BRANCH VARCHAR(4),
    SECTION VARCHAR(5),
    COURSE VARCHAR(20)
);

CREATE TABLE FEEDBACK_DETAILS(
    FEEDBACK_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    FEEDBACK_DATE DATE,
    STUDENT_ID INTEGER,
    COURSE VARCHAR(20),
    Q1 INTEGER,
    Q21 INTEGER,
    Q22 INTEGER,
    Q23 INTEGER,
    Q24 INTEGER,
    Q25 INTEGER,
    Q3 INTEGER,
    Q4 INTEGER,

    CONSTRAINT FD_K FOREIGN KEY (STUDENT_ID) REFERENCES
STUDENT_DETAILS (STUDENT_ID)
);
```

The student_details will be storing all the details about the students who have filled the Feedback form and feedback_details will be storing the information about what the students have filled in the form. The link between these two table schemas will STUDENT_ID through which you can select all the form entries the student has filled in all different Courses.

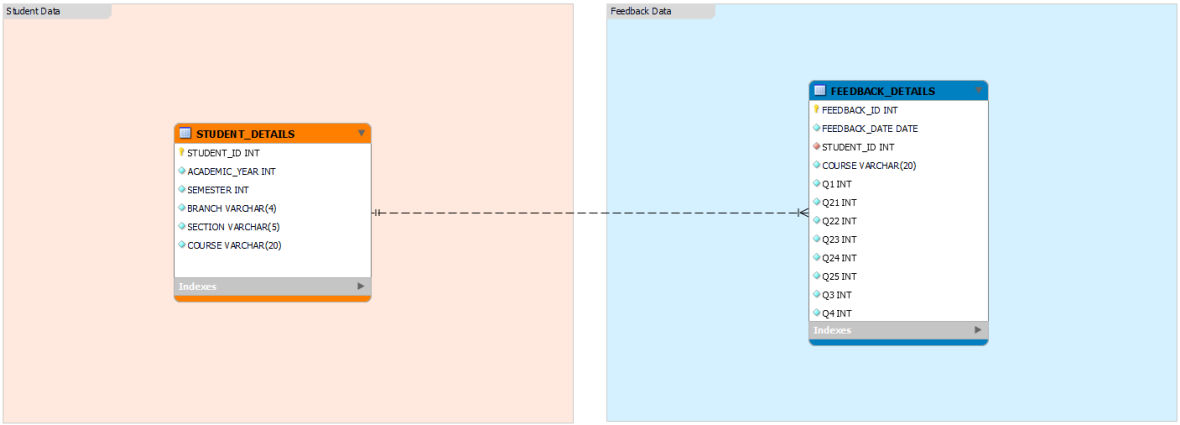
The Entities along with their Attributes have been listed below –

STUDENT_DETAILS
STUDENT_ID (PK)
ACADEMIC_YEAR
SEMESTER
BRANCH
SECTION
COURSE

FEEDBACK_DETAILS
FEEDBACK_ID (PK)(AI)
FEEDBACK_DATE
STUDENT_ID (FK)
COURSE
Q1
Q21
Q22
Q23
Q24
Q25
Q3
Q4

Here, the underlined Attributes denote Primary Keys and (FK) represents.

The Entity Relationships among all these Entities will look like -



PART – II

REGISTERATION SYSTEM

```
CREATE TABLE STUDENT_FORM(
    JEE_ROLL_NO INTEGER UNIQUE,
    FORM_ID INTEGER AUTO_INCREMENT PRIMARY KEY,
    DATE_OF_REG DATE
);



CREATE TABLE STUDENT_REG_DETAILS(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    SNAME VARCHAR(30),
    GENDER VARCHAR(1),
    BLOOD_GROUP VARCHAR(5),
    DOB DATE,
    MOBILE1 VARCHAR(11),
    EMAIL VARCHAR(30),
    AADHAR_NO VARCHAR(20),
    BRANCH VARCHAR(3),
    MINORITY VARCHAR(10),
    FATHER_NAME VARCHAR(30),
    FATHER_OCCUPATION VARCHAR(30),
    MOTHER_NAME VARCHAR(30),
    MOTHER_OCCUPATION VARCHAR(30),
    PARENT_MOBILE VARCHAR(11),
    HOSTEL_REQ VARCHAR(5),
    PHOTO VARCHAR(30),
    SIGN VARCHAR(30),

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);



CREATE TABLE STUDENT_JEE_DETAILS(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    ALLOTMENT_ROUND VARCHAR(2),
    AIR INTEGER,
    PERCENTILE INTEGER,
    ALLOTMENT_CATEGORY VARCHAR(10),
    CANDIDATE_CATEGORY VARCHAR(10),
    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);
```

```sql
CREATE TABLE STUDENT_DESEASE(
    JEE_ROLL_NO INTEGER,
    CHRONIC_DISEASE VARCHAR(4),
    DETAILS VARCHAR(100),

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);


CREATE TABLE DD_DB(
    DD_NO VARCHAR(20) PRIMARY KEY,
    DD_DATE DATE,
    DD_AMOUNT INTEGER
);


CREATE TABLE STUDENT_PAYMENTS(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    JOSSA_DD VARCHAR(20),
    INST_DD VARCHAR(20),

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO),
    FOREIGN KEY(JOSSA_DD) REFERENCES DD_DB(DD_NO),
    FOREIGN KEY(INST_DD) REFERENCES DD_DB(DD_NO)
);


CREATE TABLE STUDENT_10_DB(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    BOARD_NAME VARCHAR(30),
    PASSING_YEAR VARCHAR(4),
    PERCENTAGE INTEGER,

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);


CREATE TABLE STUDENT_12_DB(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    BOARD_NAME VARCHAR(30),
    SUBJECT VARCHAR(30),
    PASSING_YEAR VARCHAR(4),
    PERCENTAGE INTEGER,

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);
```

```sql
CREATE TABLE STUDENT_CURR_ADD(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    ADDRESS VARCHAR(100),
    CITY VARCHAR(30),
    STATE VARCHAR(30),
    PIN_CODE VARCHAR(6),
    PHONE VARCHAR(11),

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);


CREATE TABLE STUDENT_PER_ADD(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    ADDRESS VARCHAR(100),
    CITY VARCHAR(30),
    STATE VARCHAR(30),
    PIN_CODE VARCHAR(6),
    PHONE VARCHAR(11),

     FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);



CREATE TABLE STUDENT_DOCUMENTS(
    JEE_ROLL_NO INTEGER PRIMARY KEY,
    ALLOTMENT_LETTER VARCHAR(30),
    JEE_RANK_CARD VARCHAR(30),
    PHOTO_ID VARCHAR(30),
    DOB VARCHAR(30),
    QEXAM VARCHAR(30),
    INCOME_CERTIFICATE VARCHAR(30),
    CAST_CERTIFICATE VARCHAR(30),
    CAST_VALIDITY VARCHAR(30),
    OBC_CERTIFICATE VARCHAR(30),
    DISABILITY_CERTIFICATE VARCHAR(30),
    TRANSFER_CERTIFICATE VARCHAR(30),
    MIGRATION_CERTIFICATE VARCHAR(30),
    AADHAR_CARD VARCHAR(30),
    GAP_CARD VARCHAR(30),

    FOREIGN KEY(JEE_ROLL_NO) REFERENCES STUDENT_FORM(JEE_ROLL_NO)
);
```

The Entities along with their Attributes have been listed below –

STUDENT_FORM

JEE_ROLL_NO
FORM_ID(PK)(AI)
DATE_OF_REG

STUDENT_DESEASE

JEE_ROLL_NO(PK)(FK)
CHRONIC_DISEASE
DETAILS

DD_DB

DD_NO(PK)
DD_DATE
DD_AMOUNT

STUDENT_PAYMENTS

JEE_ROLL_NO(PK)(FK)
JOSSA_DD (FK)
INST_DD (FK)

STUDENT_10_DB

JEE_ROLL_NO(PK)(FK)
BOARD_NAME
PASSING_YEAR
PERCENTAGE

STUDENT_12_DB

JEE_ROLL_NO(PK)(FK)
BOARD_NAME
SUBJECT
PASSING_YEAR
PERCENTAGE

STUDENT_DOCUMENTS

JEE_ROLL_NO (PK)(FK)
Seat Allotment Letter
JEE Rank Card
Photo ID Proof
DOB
QEXAM
Income Certificate
Cast Certificate
Cast validity
Certificate for OBC
DISABILITY_CERTIFICATE
TRANSFER_CERTIFICATE
MIGRATION_CERTIFICATE
AADHAR_CARD
GAP_CARD

STUDENT_CURR_ADD

JEE_ROLL_NO(PK)(FK)
ADDRESS
CITY
STATE
PIN_CODE
PHONE

STUDENT_PER_ADD

JEE_ROLL_NO(PK)(FK)
ADDRESS
CITY
STATE
PIN_CODE
PHONE

STUDENT_REG_DETAILS

JEE_ROLL_NO (PK)(FK)
SNAME
GENDER
BLOOD_GROUP
DOB
MOBILE1
EMAIL
AADHAR_NO
BRANCH
MINORITY
FATHER_NAME
FATHER_OCCUPATION
MOTHER_NAME
MOTHER_OCCUPATION
PARENT_MOBILE
HOSTEL_REQ
PHOTO
SIGN

Entity	*Primary Key*	*Foreign Key*
STUDENT_FORM	FORM_ID	
STUDENT_REG_DETAILS	JEE_ROLL_NO	JEE_ROLL_NO
STUDENT_JEE_DETAILS	JEE_ROLL_NO	JEE_ROLL_NO
STUDENT_DISEASE	JEE_ROLL_NO	JEE_ROLL_NO
DD_DB	DD_NO	
STUDENT_PAYMENTS	JEE_ROLL_NO	JEE_ROLL_NO, JOSSA_DD, INST_DD
STUDENT_10_DB	JEE_ROLL_NO	JEE_ROLL_NO
STUDENT_12_DB	JEE_ROLL_NO	JEE_ROLL_NO
STUDENT_CURR_ADD	JEE_ROLL_NO	JEE_ROLL_NO
STUDENT_PER_ADD	JEE_ROLL_NO	JEE_ROLL_NO
STUDENT_DOCUMENTS	JEE_ROLL_NO	JEE_ROLL_NO

After implicit considerations, the constraints on these entities are as follows:

1. The entity integrity in STUDENT_FORM relation is maintained over the FORM_ID attribute.
2. The entity integrity in STUDENT_REG_DETAILS relation is maintained over the JEE_ROLL_NO attribute.
3. The entity integrity in STUDENT_JEE_DETAILS relation is maintained over the JEE_ROLL_NO attribute.
4. The entity integrity in STUDENT_DISEASE relation is maintained over the JEE_ROLL_NO attribute.
5. The entity integrity in DD_DB relation is maintained over the DD_NO attribute.
6. The entity integrity in STUDENT_PAYMENTS relation is maintained over the JEE_ROLL_NO attribute.
7. The entity integrity in STUDENT_10_DB relation is maintained over the JEE_ROLL_NO attribute.
8. The entity integrity in STUDENT_12_DB relation is maintained over the JEE_ROLL_NO attribute.
9. The entity integrity in STUDENT_CURR_ADD relation is maintained over the JEE_ROLL_NO attribute.
10. The entity integrity in STUDENT_PER_ADD relation is maintained over the JEE_ROLL_NO attribute.
11. The entity integrity in STUDENT_DOCUMENTS relation is maintained over the JEE_ROLL_NO attribute.
12. All the entities (apart from DD_DB and STUDENT_PAYMENTS) references STUDENT_FORM to enforce the referential integrity on data existence.
13. The entity STUDENT_PAYMENTS references DD_DB to enforce the referential integrity on data existence.

The Entity Relationships among all these Entities will look like –