LOAN MANAGEMENT USING SQL

What is the project about?

- Efficient management of customer loan data
- Interest Calculations
- CIBIL Score evaluation

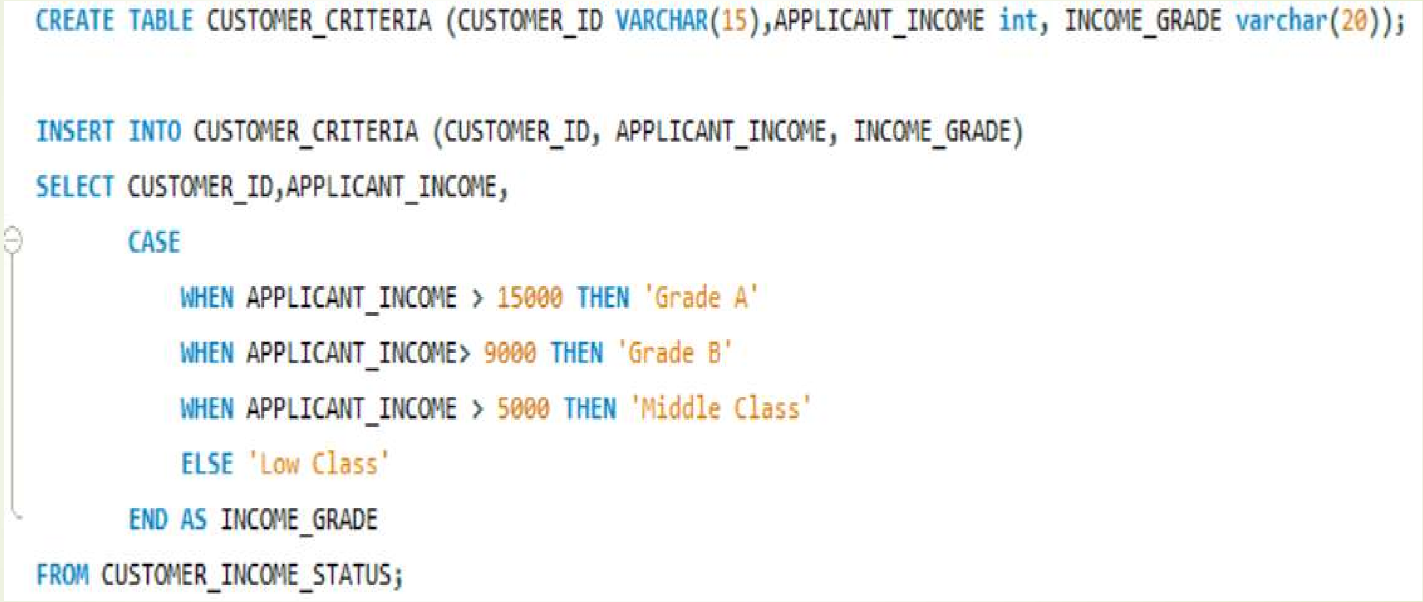
Project Objective

- Automate loan Processing
- Categorizing customers
- Apply Interest rates
- Maintain CIBIL score
- Using Triggers

TABLES IN LOAN MANAGEMENT

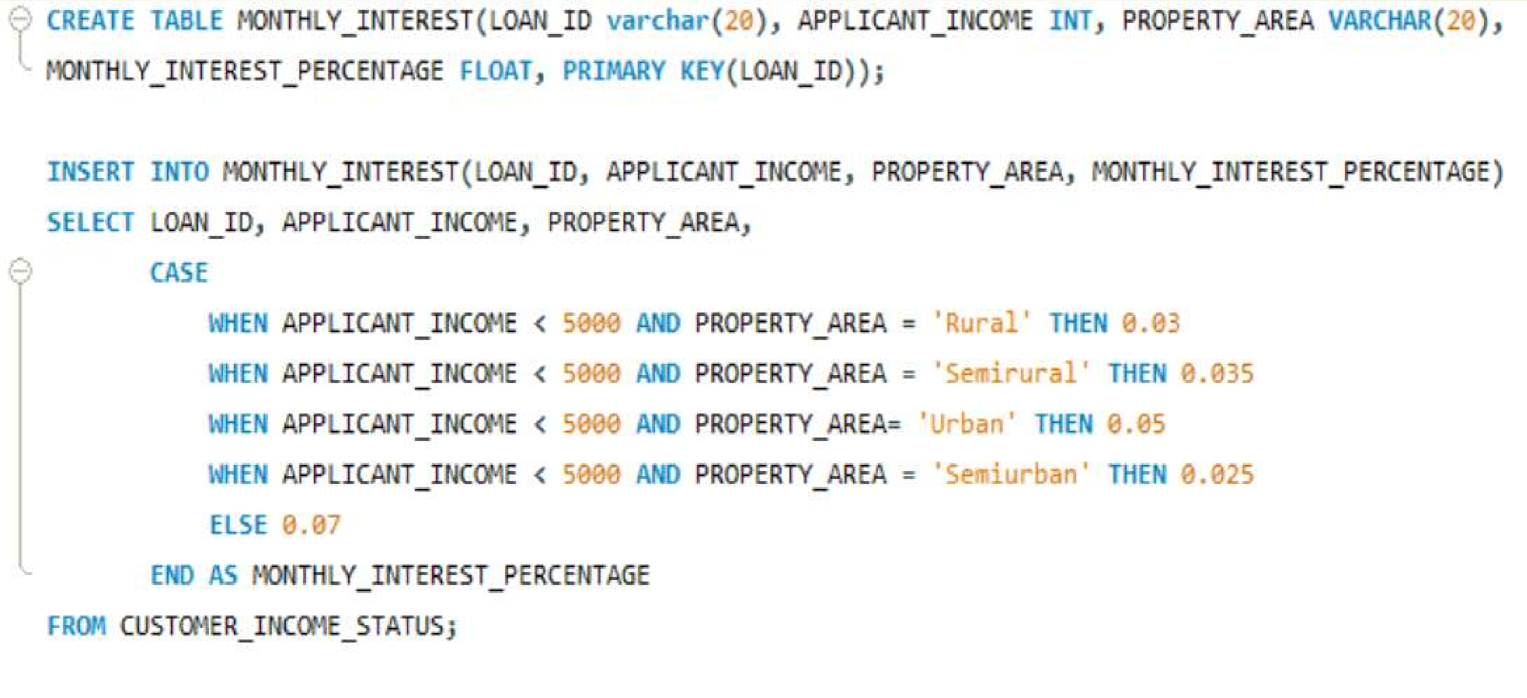
- Country_state_region
- Customer_income_status
- Loan_cibilscore_status
- Monthly_interest
- Customer_info

- Query for assigning grades



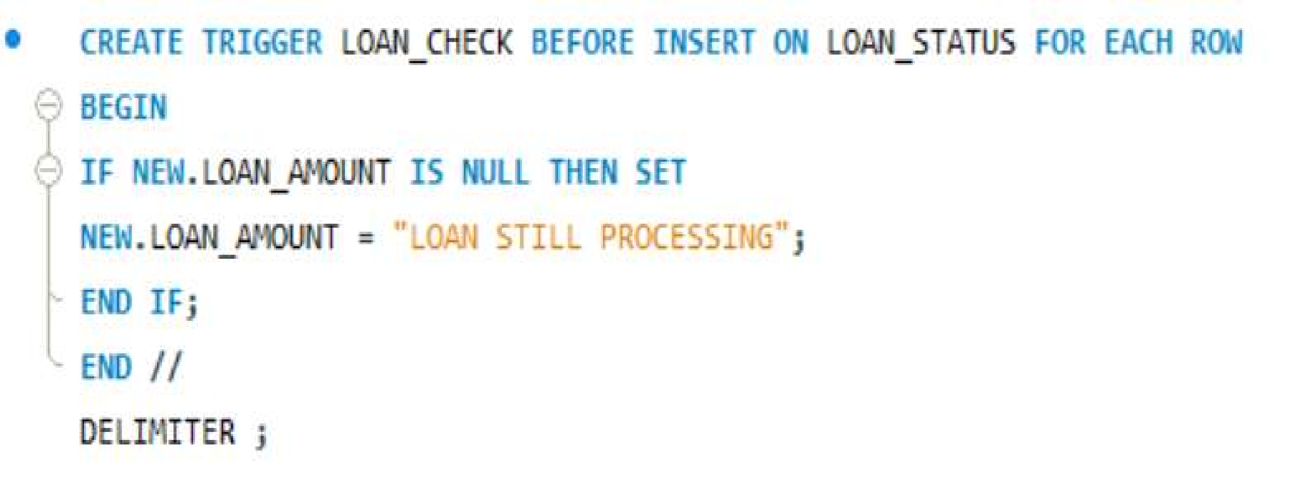
```
CREATE TABLE CUSTOMER_CRITERIA (CUSTOMER_ID VARCHAR(15),APPLICANT_INCOME int, INCOME_GRADE varchar(20));


INSERT INTO CUSTOMER_CRITERIA (CUSTOMER_ID, APPLICANT_INCOME, INCOME_GRADE)
SELECT CUSTOMER_ID,APPLICANT_INCOME,
       CASE
           WHEN APPLICANT_INCOME > 15000 THEN 'Grade A'
           WHEN APPLICANT_INCOME> 9000 THEN 'Grade B'
           WHEN APPLICANT_INCOME > 5000 THEN 'Middle Class'
           ELSE 'Low Class'
       END AS INCOME_GRADE
FROM CUSTOMER_INCOME_STATUS;
```

- Query for Monthly Interest



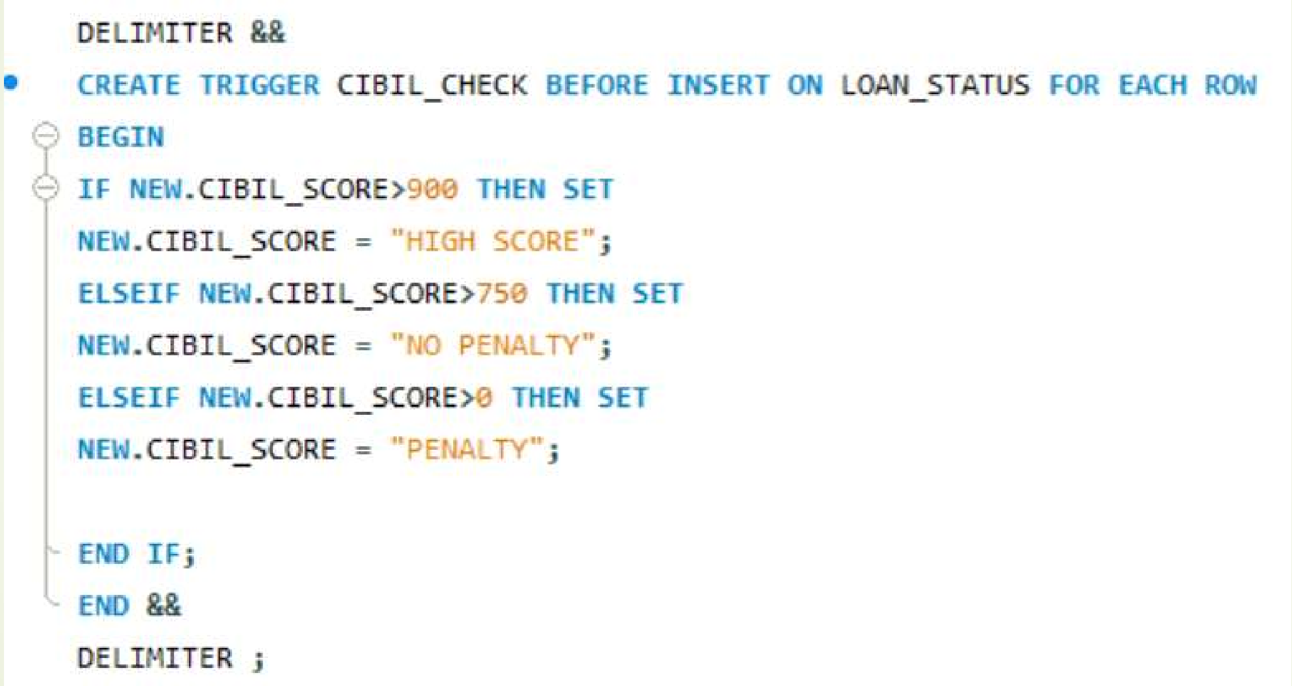
```
CREATE TABLE MONTHLY_INTEREST(LOAN_ID varchar(20), APPLICANT_INCOME INT, PROPERTY_AREA VARCHAR(20),
MONTHLY_INTEREST_PERCENTAGE FLOAT, PRIMARY KEY(LOAN_ID));

INSERT INTO MONTHLY_INTEREST(LOAN_ID, APPLICANT_INCOME, PROPERTY_AREA, MONTHLY_INTEREST_PERCENTAGE)
SELECT LOAN_ID, APPLICANT_INCOME, PROPERTY_AREA,
       CASE
           WHEN APPLICANT_INCOME < 5000 AND PROPERTY_AREA = 'Rural' THEN 0.03
           WHEN APPLICANT_INCOME < 5000 AND PROPERTY_AREA = 'Semirural' THEN 0.035
           WHEN APPLICANT_INCOME < 5000 AND PROPERTY_AREA= 'Urban' THEN 0.05
           WHEN APPLICANT_INCOME < 5000 AND PROPERTY_AREA = 'Semiurban' THEN 0.025
           ELSE 0.07
       END AS MONTHLY_INTEREST_PERCENTAGE
FROM CUSTOMER_INCOME_STATUS;
```

- Trigger for Loan Processing



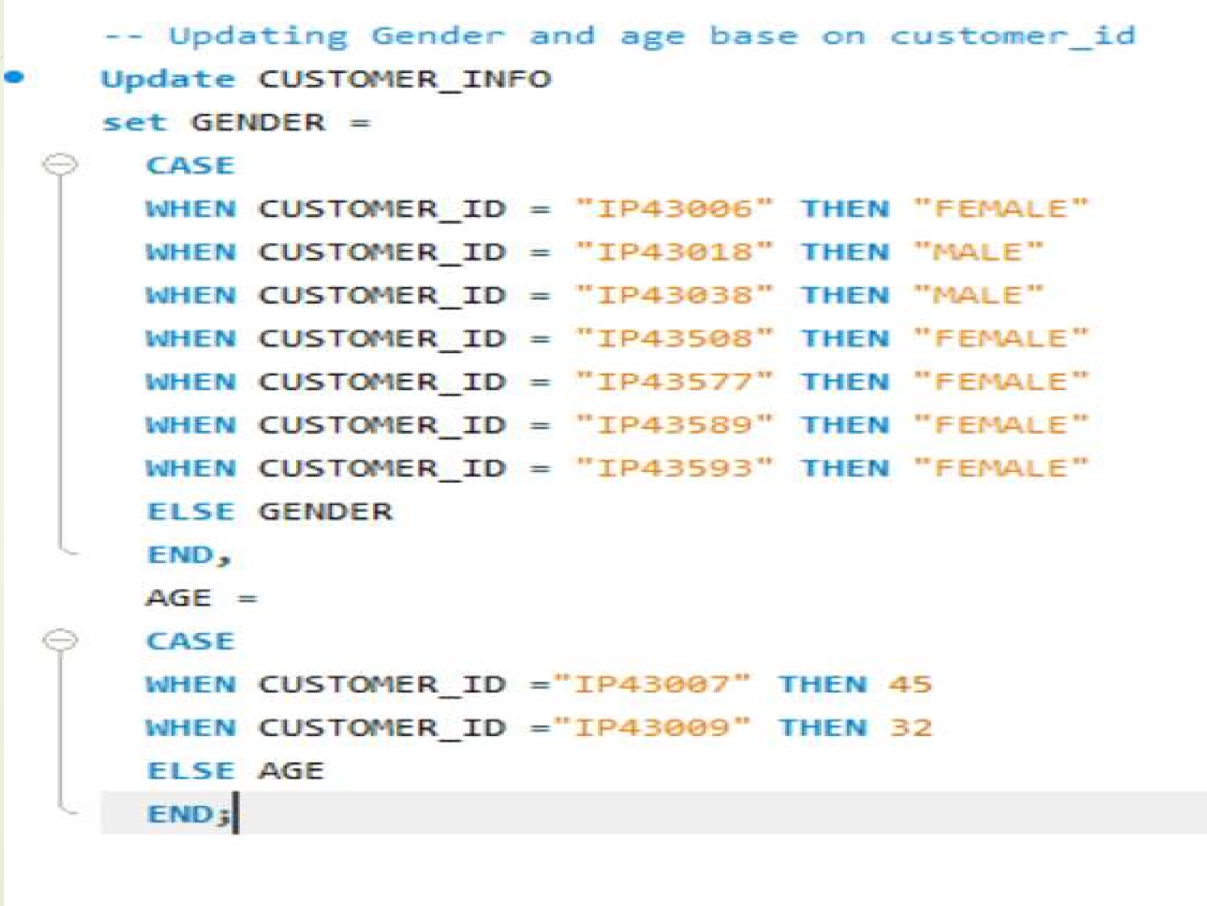
```
CREATE TRIGGER LOAN_CHECK BEFORE INSERT ON LOAN_STATUS FOR EACH ROW
BEGIN
IF NEW.LOAN_AMOUNT IS NULL THEN SET
NEW.LOAN_AMOUNT = "LOAN STILL PROCESSING";
END IF;
END //
DELIMITER ;
```

- Trigger for Row level Statement



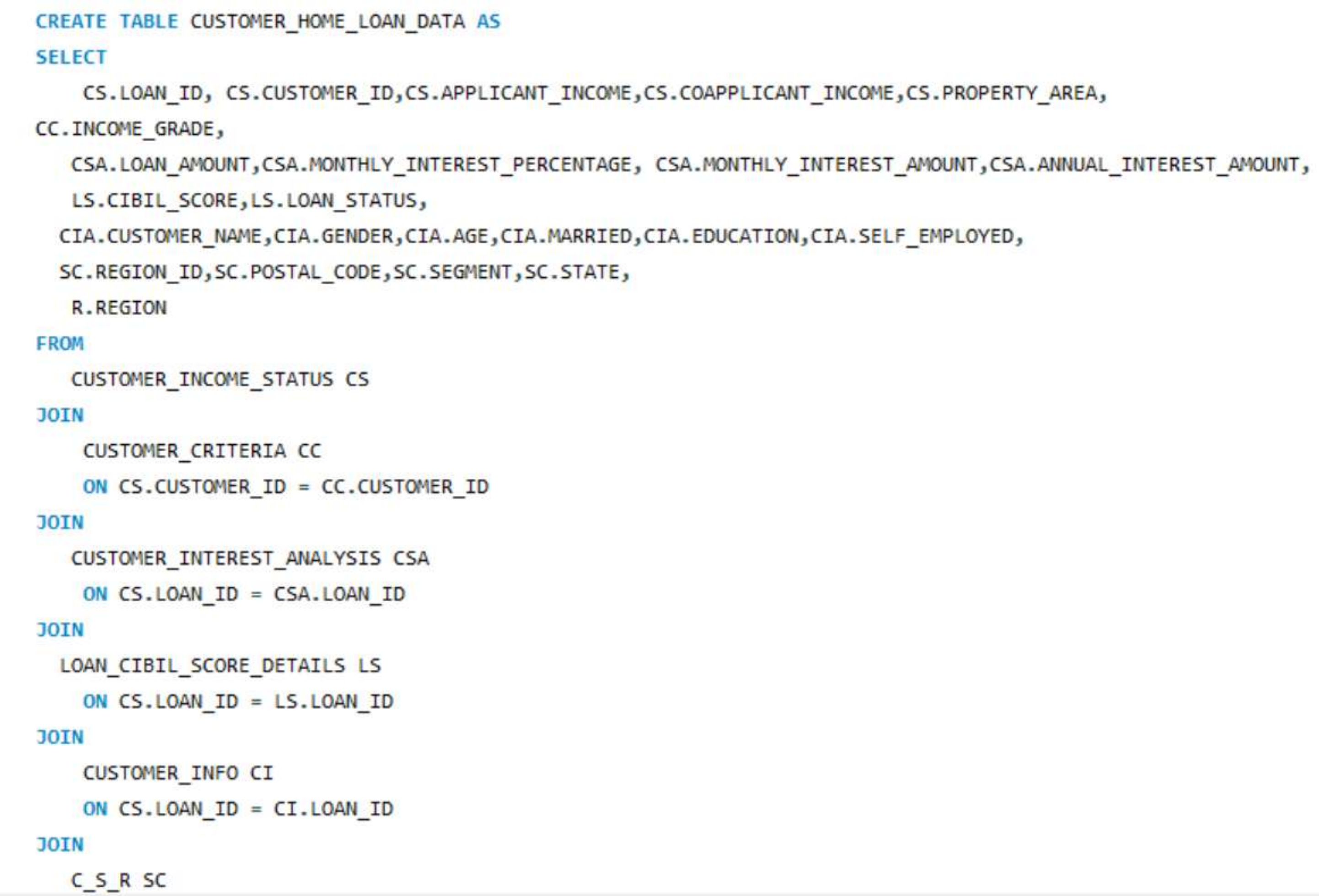
```
DELIMITER &&
CREATE TRIGGER CIBIL_CHECK BEFORE INSERT ON LOAN_STATUS FOR EACH ROW
BEGIN
IF NEW.CIBIL_SCORE>900 THEN SET
NEW.CIBIL_SCORE = "HIGH SCORE";
ELSEIF NEW.CIBIL_SCORE>750 THEN SET
NEW.CIBIL_SCORE = "NO PENALTY";
ELSEIF NEW.CIBIL_SCORE>0 THEN SET
NEW.CIBIL_SCORE = "PENALTY";


END IF;
END &&
DELIMITER ;
```

- Updating Gender and Age for Customer ID



```
-- Updating Gender and age base on customer_id
Update CUSTOMER_INFO
set GENDER =
   CASE
   WHEN CUSTOMER_ID = "IP43006" THEN "FEMALE"
   WHEN CUSTOMER_ID = "IP43018" THEN "MALE"
   WHEN CUSTOMER_ID = "IP43038" THEN "MALE"
   WHEN CUSTOMER_ID = "IP43508" THEN "FEMALE"
   WHEN CUSTOMER_ID = "IP43577" THEN "FEMALE"
   WHEN CUSTOMER_ID = "IP43589" THEN "FEMALE"
   WHEN CUSTOMER_ID = "IP43593" THEN "FEMALE"
   ELSE GENDER
   END,
   AGE =
   CASE
   WHEN CUSTOMER_ID ="IP43007" THEN 45
   WHEN CUSTOMER_ID ="IP43009" THEN 32
   ELSE AGE
   END;
```

- Joins



```
CREATE TABLE CUSTOMER_HOME_LOAN_DATA AS
SELECT
    CS.LOAN_ID, CS.CUSTOMER_ID,CS.APPLICANT_INCOME,CS.COAPPLICANT_INCOME,CS.PROPERTY_AREA,
CC.INCOME_GRADE,
   CSA.LOAN_AMOUNT,CSA.MONTHLY_INTEREST_PERCENTAGE, CSA.MONTHLY_INTEREST_AMOUNT,CSA.ANNUAL_INTEREST_AMOUNT,
   LS.CIBIL_SCORE,LS.LOAN_STATUS,
  CIA.CUSTOMER_NAME,CIA.GENDER,CIA.AGE,CIA.MARRIED,CIA.EDUCATION,CIA.SELF_EMPLOYED,
  SC.REGION_ID,SC.POSTAL_CODE,SC.SEGMENT,SC.STATE,
   R.REGION
FROM
   CUSTOMER_INCOME_STATUS CS
JOIN
    CUSTOMER_CRITERIA CC
    ON CS.CUSTOMER_ID = CC.CUSTOMER_ID
JOIN
   CUSTOMER_INTEREST_ANALYSIS CSA
    ON CS.LOAN_ID = CSA.LOAN_ID
JOIN
  LOAN_CIBIL_SCORE_DETAILS LS
    ON CS.LOAN_ID = LS.LOAN_ID
JOIN
    CUSTOMER_INFO CI
    ON CS.LOAN_ID = CI.LOAN_ID
JOIN
   C_S_R SC
```