**SQL**

1. Examine the commands used to create DEPARTMENT\_DETAILS and COURSE\_DETAILS table.

CREATE TABLE DEPARTMENT\_DETAILS

(

DEPARTMENT\_ID NUMBER PRIMARY KEY,

DEPARTMENT\_NAME VARCHAR2(50),

HOD VARCHAR2(50));

CREATE TABLE COURSE\_DETAILS

(

COURSEID NUMBER PRIMARY KEY,

COURSE\_NAME VARCHAR2(50),

DEPARTMENT\_ID NUMBER REFERENCES DEPARTMENT\_DETAILS (DEPARTMENT\_ID)

);

DEPARTMENT\_DETAILS

A screenshot of a computer

AI-generated content may be incorrect.

COURSE\_DETAILS

A screenshot of a computer

AI-generated content may be incorrect.

**Generate a list of all department IDs along with any course IDs that may have been assigned to them.**

2.

a) CREATE A TEACHER TABLE AND a STUDENT TABLE  
        WITH COLUMNS TO HAVE THE RESPECTIVE CONSTRAINTS  
\* PRIMARY KEY  
\* NOT NULL  
\* UNIQUE  
\* CHECK  
\* FOREIGN KEY  
b) STORE THE REQUIRED DATA IN BOTH TABLES TO CHECK THE CONSTRAINTS

c) Write a query to display the student's name, course, marks, and Teacher Id with the teacher's name.

**PL/SQL**

1. Imagine a scenario where a company wants to update the salaries of its employees. The company gives a raise to all employees in the "Sales" department. We need a PL/SQL procedure that updates the salary of employees who work in the "Sales" department by a specific percentage.

A screenshot of a computer code

AI-generated content may be incorrect.

Create a PL/SQL procedure named update\_sales\_salary that:

1. Takes the percentage raise as an input parameter.
2. Updates the salary of all employees in the "sales" department by the given percentage.

2. Write a stored procedure to find the max and min salary of employees in any given department. Handle the exceptions.