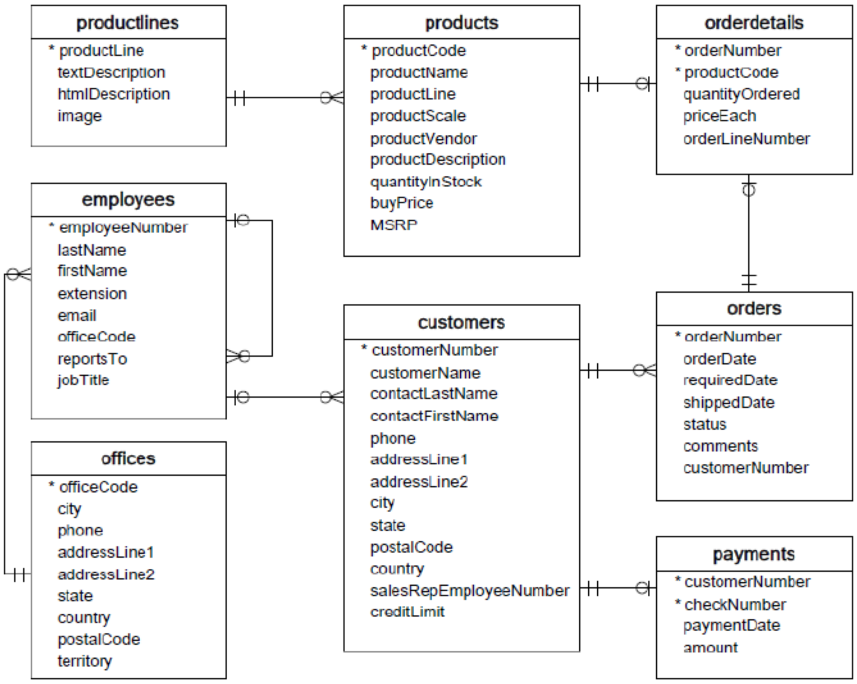
DBMS Practical Assignment

Dr. J. V. Smart

Table of Contents

The Classic Models Sample Database



The Classic Models Sample Database

CMOFFICES

Column Name	Data Type and Size	Constraints	Description
officecode	VARCHAR2(10)	Primary key	Office code
city	VARCHAR2(50)	Not null	Office city
phone	VARCHAR2(50)	Not null	Office phone number
addressline1	VARCHAR2(50)	Not null	Office address (line 1)
addressline2	VARCHAR2(50)		Office address (line 2)

Column Name	Data Type and Size	Constraints	Description
state	VARCHAR2(50)		Office in which state?
country	VARCHAR2(50)	Not null	Country of the office
postalcode	VARCHAR2(15)	Not null	Postal code of the office
territory	VARCHAR2(10)	Not null	Sales territory of the office

CMEMPLOYEES

Column Name	Data Type and Size	Constraints	Description
employeenumber	NUMBER(10,0)	Primary key	Employee number
lastname	VARCHAR2(50)	Not null	Last name of the employee
firstname	VARCHAR2(50)	Not null	First name of the employee
extension	VARCHAR2(10)	Not null	office intercom phone extension no. of the employee
email	VARCHAR2(100)	Not null	email of the employee
officecode	VARCHAR2(10)	Not null, Foreign key into CMOFFICES table	Office code of the office in which the employee works
reportsto	NUMBER(10,0)	Foreign key into CMEMPLOYEES(employeenumber)	The immediate superior of the employee
jobtitle	VARCHAR2(50)	Not null	Job title (designation) of the employee

CMCUSTOMERS

Column Name	Data Type and Size	Constraints	Description
customernumber	NUMBER(10,0)	Primary key	Customer number
customername	VARCHAR2(50)	Not null	Customer name
contactlastname	VARCHAR2(50)	Not null	Last name of the contact person of the customer
contactfirstname	VARCHAR2(50)	Not null	First name of the contact person of the customer
phone	VARCHAR2(50)	Not null	Phone number of the customer
addressline1	VARCHAR2(50)	Not null	Customer address (line 1)
addressline2	VARCHAR2(50)		Customer address (line 2)
city	VARCHAR2(50)	Not null	City of the customer
state	VARCHAR2(50)		State of the customer
postalcode	VARCHAR2(15)		Postal code of the customer
country	VARCHAR2(50)	Not null	Country of the customer

Column Name	Data Type and Size	Constraints	Description
salesrepemployeenumber	NUMBER(10,0)	Foreign key into CMEMPLOYEES(employeenumber)	Sales rep assigned to the customer
creditlimit	FLOAT(126)		The amount of credit allowed to the customer

CMPRODUCTLINE

Column Name	Data Type and Size	Constraints	Description
productline	VARCHAR2(50)	Primary key	Product line (category)
textdescription	VARCHAR2(4000)		Product description in textual form
htmldescription	CLOB		Product description in HTML form
image	BLOB		Image in binary form

CMPRODUCTS

Column Name	Data Type and Size	Constraints	Description
productcode	VARCHAR2(15)	Primary key	Product code
productname	VARCHAR2(70)	Not null	Name of the product
productline	VARCHAR2(50)	Not null, Foreign key into CMPRODUCTLINES table	Product line (category) of the product)
productscale	VARCHAR2(10)	Not null	Scale ratio of the product
productvendor	VARCHAR2(50)	Not null	Vendor of the product
productdescription	CLOB	Not null	Description of the product

Column Name	Data Type and Size	Constraints	Description
quantityinstock	NUMBER(5,0)	Not null	Stock of the product
buyprice	FLOAT(126)	Not null	
msrp	FLOAT(126)	Not null	Manufacturer's Suggested Retail Price of the product

CMORDERS

Column Name	Data Type and Size	Constraints	Description
ordernumber	NUMBER(10,0)	Primary key	Order number
orderdate	DATE	Not null	Order date
requireddate	DATE	Not null	Date when delivery required
shippeddate	DATE		Date of order shipment
status	VARCHAR2(15)	Not null	Status of the order
comments	CLOB		Comments
customernumber	NUMBER(10,0)	Not null, Foreign key into CMCUSTOMERS table	Customer number of the customer who placed the order

CMORDERDETAILS

Column Name	Data Type and Size	Constraints	Description
ordernumber	NUMBER(10,0)	Primary key, Foreign key into CMORDERS table	Order number
productcode	VARCHAR2(15)	Primary key, Foreign key into CMPRODUCTS table	Product code
quantityordered	NUMBER(10,0)	Not null	Quantity ordered
priceeach	FLOAT(126)	Not null	Price of each item

Column Name	Data Type and Size	Constraints	Description
orderlinenumber	NUMBER(5,0)	Not null	Line number of this item in the order

CMPAYMENTS

Column Name	Data Type and Size	Constraints	Description
customernumber	NUMBER(10,0)	Primary key, Foreign key into CMCUSTOMERS table	Customer number
checknumber	VARCHAR2(50)	Primary key	Check (cheque) number
paymentdate	DATE Not null		Payment date
amount	FLOAT(126) Not null		Payment amount

IMPORTANT NOTE: Create the above tables and insert the sample data using the supplied script.

IMPORTANT NOTE: Save all solutions in a file `PID-dbms-assignment.txt` (where `PID` is your PID) in the following format

```
PID:

Name:

1.

CREATE TABLE ...

2.

INSERT INTO ...
```

Notes

1. Write only the SQL or PL/SQL commands. Do not write the output
2. It is not necessary to solve the problems in any particular order. But in the assignment, write the solutions in order
3. Do not skip any number. If you have not solved a problem at the moment, write its number and leave a blank line
4. The assignment must be submitted in written or soft copy form
5. The assignment must be submitted before your internal practical examination

6. More problems may be added to this assignment

SQL Problems

1. Create the following table:

DEPARTMENTS

Column Name	Data Type and Size	Constraints	Description
departmentno	NUMBER(3)	Primary key	Department number
name	VARCHAR(10)	Unique, Not null	Department name
location	VARCHAR(5)	Not null	Location of the department

2. Insert the following data in the Departments table:

departmentno	name	location
1	Finance	Mumbai
2	Purchase	Ahmedabad
3	Sales	Ahmedabad
4	HR	Vadodara

3. Create the following table:

EMPLOYEES

Column Name	Data Type and Size	Constraints	Description
employeeno	NUMBER(3)	Primary key	Employee number
name	VARCHAR(10)	Not null	Employee name
departmentno	NUMBER(3)	Foreign key into DEPARTMENTS table	department number in which the employee works
designation	VARCHAR(10)		Designation of the employee
joiningdate	DATE	Not null, Must be on or after `01-01-2010` , Default value `01-01-2010`	Employee's joining date

Column Name	Data Type and Size	Constraints	Description
salary	NUMBER(10, 2)	Must be greater than 0	Salary of the employee

4. Insert the following data in the Employees table using substitution variables (`&`):

employeeno	name	departmentno	designation	joiningdate	salary
1	Arnab	1	Manager	23-10-2019	<null>
2	Ayushi	3	Sales Officer	11-11-2011	27000
3	Arohi	2	Purchase Officer	05-12-2015	24000
4	Agneya	3	Clerk	21-12-2010	16000
5	Akshat	1	Clerk	02-04-2017	13000
6	Asmita	4	<null>	14-12-2021	<null>

5. Display the table structure of the `EMPLOYEES` table using SQL *PLUS command

6. Display all the rows in the `DEPARTMENTS` table

7. Display all the rows in the `EMPLOYEES` table

8. Insert a row in the `EMPLOYEES` table with employee number 7 using default value for the joining date

9. Change the name of employee number 7 to `qwerty` and salary to 121212

10. Delete the employee with employee number 7

11. Display all the employees whose salary is greater than or equal to ₹20,000

12. Display all the employees who work in department number 3 and whose salary is greater than or equal to ₹20,000

13. Display all the employees who work in department number 3 or draw salary greater than or equal to ₹20,000

14. Display all the employees who are not clerks

15. Display all the employees in the order of employee number

16. Display all the employees in the order of department number with secondary ordering on employee name

17. Display all the employees in the order of department number with secondary ordering on salary (highest first)

18. Create a table `temp7 (no NUMBER(3), name VARCHAR(10))`

19. Add a column `c1 VARCHAR(5)` to the table `temp7`

20. Modify the data type of the column `c1` in the table `temp7` to `NUMBER(5)`

21. Delete the table `temp7`

22. List all the tables you have created

23. Delete the table `EMPLOYEES`

24. Create the table `EMPLOYEES` again without the constraints

25. Add all the constraints to the `EMPLOYEES` table

26. Insert the data in the `EMPLOYEES` table as given in this assignment again

27. Display employee name and designation separated by a `,` (comma) as a single column

28. Display all the employees whose salary is undefined (Null)

29. Display all the employees whose salary is not undefined (Null)

30. Display all employees whose salary is between ₹15,000 and ₹25,000

31. Display all employees whose department number is either 1 or 3

32. Display all employees whose department number is either 1, 3, 6 or 8

33. Display all employees whose name begins with `A`

34. Display all employees whose name begins with `A` and ends in `i`

35. Display all employees whose name begins with `A` and has `n` as the third character

36. Display all employees whose name is 5 characters long

37. Display all employees who joined on or after `01-01-2021`

    Answer the following questions using the `Classic Models` sample database.

    ```
    set linesize 95
    column lastname format a15
    column firstname format a15
    column email format a32
    column jobtitle format a20
    column city format a15
    column phone format a15
    column addressline1 format a25
    column addressline2 format a15
    column postalcode format a10
    column country format a15
    column state format a15
    set pagesize 1000
    ```

38. Display all the offices in the USA

39. Display all the offices from cities whose names start with `S`

40. Display the order number and order date (in `DD-MM-YYYY` format) for all the orders in the order of order number

41. Display the total number of orders

42. For each order, display the order number and product code of each product in that order in the order of order number (descending) and product code (ascending)

43. For each order, display the order number, order date and product code of each product in that order in the order of order number (descending) and product code (ascending)

44. Display the order number, product code and amount for each product in the order number 10282 (amount = quantity ordered × price each)

45. Display the order number, order date, product code and amount for each product in the order number 10282 (amount = quantity ordered × price each)

46. Display the order number, order date, customer name, product code and amount for each product in the order number 10282 (amount = quantity ordered × price each)

47. Display the number of products in each order

48. Display the maximum msrp for a product

49. Display the product code and product name for the product with the maximum msrp

50. Display the product code, product name and product line text description for the product with the maximum msrp

51. Display the highest single payment made by a customer

52. Display the customer number of the customer who has made the highest single payment

53. Display the customer number and customer name of the customer who has made the highest single payment

54. Display the highest total payment made by a customer

55. Display the customer number of the customer who has made the highest total payment

56. Display the customer number and customer name of the customer who has made the highest total payment

57. Display the total of all payments made by customers

PL/SQL Problems

1. Write a PL/SQL block that displays the sum of two numeric variables whose values

are provided using substitution variables (&)

2. Write a PL/SQL block that declares two numeric variables `no1` and `no2` whose values are provided using substitution variables (&). It should display `Greater` if `no1 > no2` and `Less than or equal` if `no1 <= no2`
3. Write a PL/SQL block that declares two numeric variables `no1` and `no2` whose values are provided using substitution variables (&). It should display `Greater` if `no1 > no2` , `Less` if `no1 < no2` and `Equal` if `no1 = no2`
4. Write a PL/SQL block that declares two numeric variables `no1` and `no2` whose values are provided using substitution variables (&). It should display all the numbers between `no1` and `no2` , including `no1` and `no2`
5. Insert department number `5` in the `DEPARTMENTS` table from a PL/SQL block
6. Display the information of department number `5` from a PL/SQL block
7. Modify the location of department number `5` from a PL/SQL block
8. Delete department number `5` from a PL/SQL block
9. Write a PL/SQL block that displays all the information from the `DEPARTMENTS` table
10. Write a PL/SQL block that displays all the information from the `DEPARTMENTS` table in a tabular format
11. Write a PL/SQL block. Define a function `increment` that takes two parameters `amount` and `percentage` of type number. The function should increment the `amount` by the given `percentage` and return the result (e.g. if `amount` is `1200` and `percentage` is `5` , it should return `1260`). Declare appropriate variables, obtain their values using substitution variables (&) and call the function from the body of the PL/SQL block. Display the value returned by the function.
12. Write a PL/SQL block. Define a procedure `new_department` that accepts `departmentno` and `dpartmentname` as parameters. It should insert a row in the `DEPARTMENTS` table using the values passed and `Surat` as location. Declare appropriate variables, obtain their values using substitution variables (&) and call the procedure from the body of the PL/SQL block.