Exercises: Table Relations

This document defines the **exercise assignments** for the "Databases Basics - MySQL" course @ Software University.

Problem 1. One-To-One Relationship

Create two tables as follows. Use appropriate data types.

persons			
person_id	**first_name**	**salary**	**passport_id**
1	Roberto	43300.00	102
2	Tom	56100.00	103
3	Yana	60200.00	101

passports	
passport	**passport_number**
101	N34FG21B
102	K65LO4R7
103	ZE657QP2

Insert the data from the example above.

Alter table persons and make person_id a primary key. Create a foreign key between persons and passports by using passport_id column.

Submit your queries by using **MySQL run skeleton run queries and check db.**

Problem 2. One-To-Many Relationship

Create two tables as follows. Use appropriate data types.

manufacturers		
manufacturer_id	**name**	**established_on**
1	BMW	07/03/1916
2	Tesla	01/01/2003
3	Lada	01/05/1966

models		
model_id	**name**	**manufacturer_id**
101	X1	1
102	i6	1
103	Model S	2
104	Model X	2
105	Model 3	2
106	Nova	3

Insert the data from the example above. Add primary keys and foreign keys.

Submit your queries by using **MySQL run skeleton run queries and check db.**

Problem 3. Many-To-Many Relationship

Create three tables as follows. Use appropriate data types.

students	
student_id	**name**
1	Mila
2	Toni
3	Ron

exams	
exam_id	**name**
101	Spring MVC
102	Neo4j
103	Oracle 11g

students_exams	
student_id	**exam_id**
1	101
1	102
2	101
3	103
2	102
2	103

Insert the data from the example above.
Add primary keys and foreign keys. Have in mind that table StudentsExams should have a composite primary key.

Submit your queries by using **MySQL run skeleton run queries and check db.**

Problem 4. Self-Referencing

Create a single table as follows. Use appropriate data types.

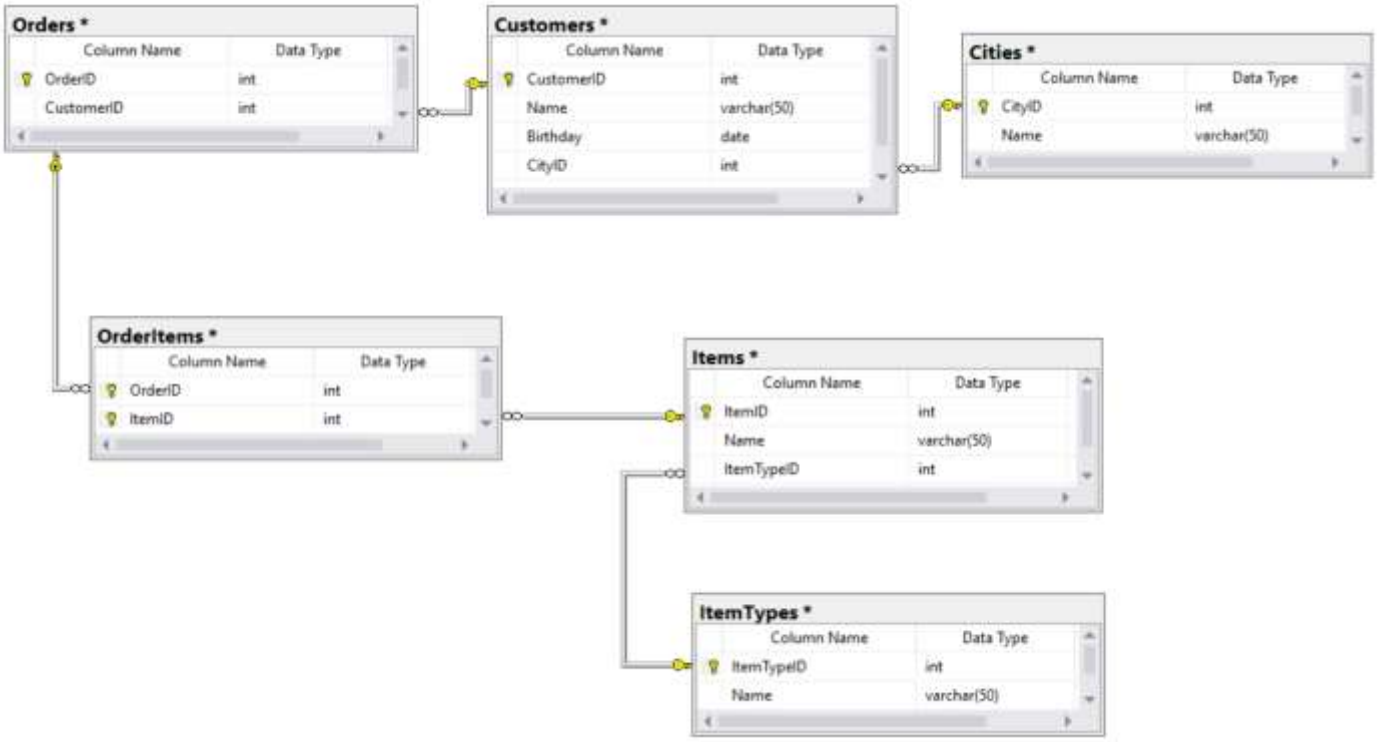
teachers		
teacher_id	**name**	**manager_id**
101	John	
102	Maya	106
103	Silvia	106
104	Ted	105
105	Mark	101
106	Greta	101

Insert the data from the example above. Add primary keys and foreign keys. The foreign key should be between ManagerId and TeacherId.

Submit your queries by using **MySQL run skeleton run queries and check db.**

Problem 5. Online Store Database

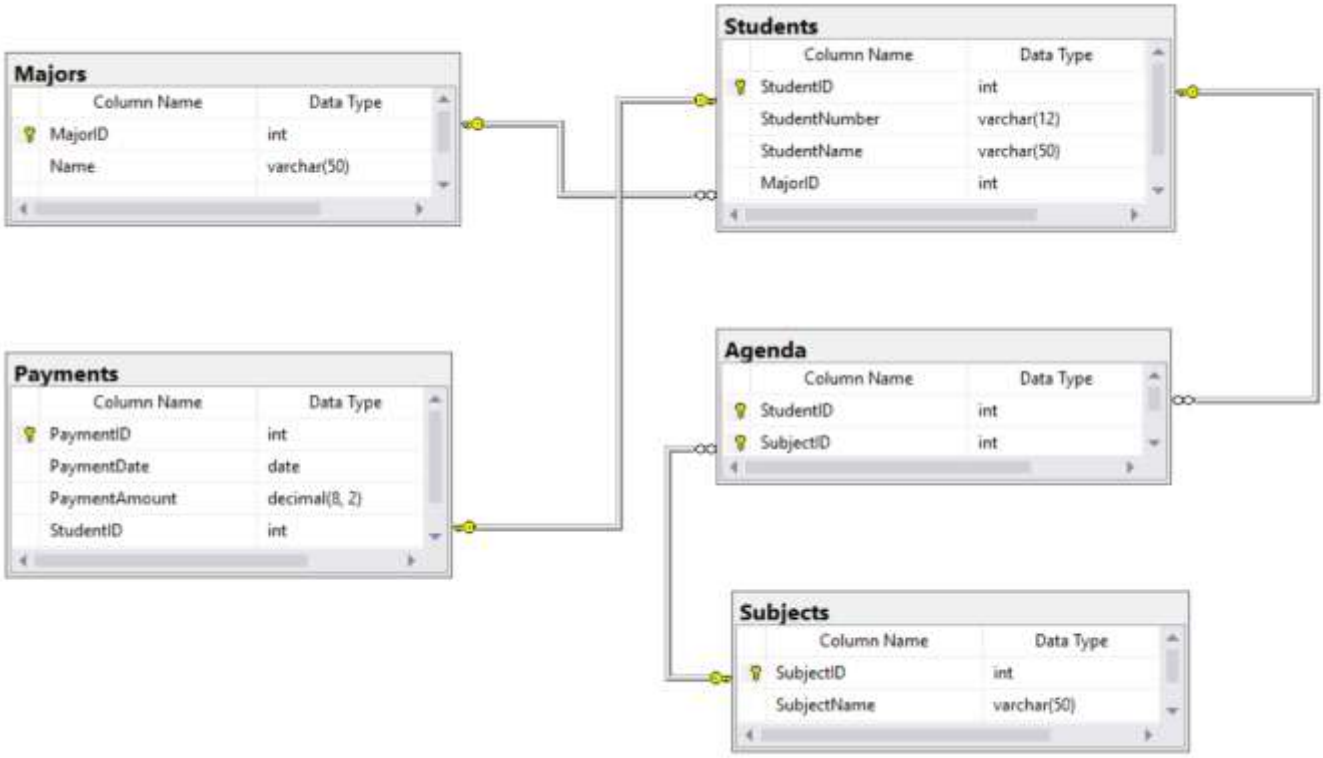
Create a new database and design the following structure:



Submit your queries by using **MySQL run skeleton run queries and check db.**

Problem 6. University Database

Create a new database and design the following structure:



Submit your queries by using **MySQL run skeleton run queries and check db.**

Problem 7. Soft_Uni Design

Create an E/R Diagram of the SoftUni Database.

Problem 8. Geography Design

Create an E/R Diagram of the Geography Database.

Problem 9. Employee Address

Write a query that selects:

- employee_id
- job_title
- address_id
- address_text

Return the first 5 rows sorted by address_id in ascending order.

Example:

employee_id	job_title	address_id	address_text
142	Production Technician	1	108 Lakeside Court
…	…	…	…

Problem 10. Employee Departments

Write a query that selects:

- employee_id
- first_name
- salary
- department_name

Filter only employees with salary higher than 15000. Return the first 5 rows sorted by DepartmentID in ascending order.

Example:

employee_id	first_name	salary	department_name
3	Roberto	43300.00	Engineering
…	…	…	…

Problem 11. Employees Without Project

Write a query that selects:

- employee_id
- first_name

Filter only employees without a project. Return the first 3 rows sorted by employee_id in ascending order.

Example:

employee_id	first_name

2	Kevin
…	…

Problem 12. Employees with Project

Write a query that selects:

- employee_Id
- first_name
- project_name

Filter only employees with a project which has started after 13.08.2002 and it is still ongoing (no end date). Return the first 5 rows sorted by employee_id in ascending order.

Example

employee_id	first_name	project_name
5	Thierry	Road-350
…	…	…

Problem 13. Employee 24

Write a query that selects:

- employee_id
- first_name
- project_name

Filter all the projects of employee with id 24. If the project has started before 2005 the return value should be NULL.

Example

employee_id	first_name	project_name
24	David	NULL
24	David	Road-650
…	…	…

Problem 14. Employee Manager

Write a query that selects:

- employee_id
- first_name
- manager_id
- manager_name

Filter all employees with a manager who has ID equals to 3 or 7. Return the all rows sorted by employee_id in ascending order.

Example

employee_id	first_name	manager_id	manager_name
4	Rob	3	Roberto
…	…	…	…

Problem 15. Highest Peak in Bulgaria

Write a query that selects:

- country_code
- mountan_range
- peak_name
- elevation

Filter all peaks in Bulgaria with elevation over 2835. Return the all rows sorted by elevation in descending order.

Example

country_code	mountan_range	peak_name	elevation
BG	Rila	Musala	2925
…	…	…	…

Problem 16. Count Mountain Ranges

Write a query that selects:

- country_code
- mountan_ranges

Filter the count of the mountain ranges in the USA, Russia and Bulgaria.

Example

country_code	mountan_ranges
BG	6
…	…

Problem 17. Countries with Rivers

Write a query that selects:

- country_name
- river_name

Find the first 5 countries with or without rivers in Africa. Sort them by country_name in ascending order.

Example

country_name	river_name
Algeria	Niger
Angola	Congo
Benin	Niger
Botswana	NULL
Burkina Faso	Niger

Problem 18. *Continents and Currencies

Write a query that selects:

- continent_code
- currency_code

- currency_usage

Find all continents and their most used currency. Filter any currency that is used in only one country. Sort your results by continent_code.

Example

continent_code	currency_code	currency_usage
AF	XOF	8
AS	AUD	2
AS	ILS	2
EU	EUR	26
NA	XCD	8
OC	USD	8

Problem 19. Countries Without any Mountains

Write a query that selects **country_code.** Find all the count of all countries which don't have a mountain.

Example

country_code
231

Problem 20. Highest Peak and Longest River by Country

For each country, find the elevation of **the highest peak** and **the length of the longest river**, sorted by the highest peak elevation (from highest to lowest), then by the longest river length (from longest to smallest), then by country name (alphabetically). Display **NULL** when no data is available in some of the columns. Submit for evaluation the result grid with headers. Limit only the first 5 rows.

country_name	highest_peak_elevation	longest_river_length
China	8848	6300
India	8848	3180
Nepal	8848	2948
Pakistan	8611	3180
Argentina	6962	4880
Chile	6962	NULL
…	…	…

Problem 21. ** Highest Peak Name and Elevation by Country

For each country, find the name and elevation of **the highest peak**, along with its mountain. When no peaks are available in some country, display elevation **0**, "**(no highest peak)**" as peak name and "**(no mountain)**" as mountain name. When multiple peaks in some country have the same elevation, display all of them. Sort the results by country name alphabetically, then by highest peak name alphabetically. Submit for evaluation the result grid with headers. Limit only the first 5 rows.

country	highest_peak_name	highest_peak_elevation	mountain
Afghanistan	(no highest peak)	0	(no mountain)
…	…	…	…
Argentina	Aconcagua	6962	Andes
…	…	…	…

Bulgaria	Musala	2925	Rila
Burkina Faso	(no highest peak)	0	(no mountain)
…	…	…	…
United States	Mount McKinley	6194	Alaska Range
…	…	…	…
Zambia	(no highest peak)	0	(no mountain)
Zimbabwe	(no highest peak)	0	(no mountain)