**SQL Server Assignments**

**Create A New DATABASE CALLED ASSIGNMENTDB and DO THE ASSIGNMENTS IN Assignmentdb**

1. Create the tables **Customers** and **Orders** with the following columns. ( do not declare the corresponding primary and foreign keys )

|  |
| --- |
| **Customers** |
| Customerid char(5) not null |
| CompanyNamevarchar(40) not null |
| contactName char(30) null |
| Address varchar(60) null |
| City char(15) null |
| Phone char(24) null |
| Fax char(24) null |

|  |
| --- |
| **Orders** |
| OrderId integer not null |
| customerId char(5) not null |
| Orderdatedatetime null |
| Shippeddatedatetime null |
| Freight money null |
| Shipnamevarchar(40) null |
| Shipaddresvarchar(60) null |
| Quantity integer null |

1. Using the ALTER TABLE statement, add a new column named **shipregion** to the **Orders** table. The fields should be nullable and contain integers.
2. Using the ALTER TABLE statement, change the data type of the column **shipregion**from INTEGER to CHARACTER with length 8. The fields may contain null values.
3. Delete the formerly created column **shipregion.**
4. try to insert a new row into the **Orders** table with the following values:

( 10, ‘ord01’, getdate(), getdate(), 100.0, ‘Windstar’, ‘Ocean’ ,1)

Find out the reason why this is not possible.

1. Using the ALTER TABLE statement, add the current system date and time as the default value to the **orderdate** column of the **Orders** table.
2. Rename the city column of the **Customers** table. The new name is **Town.**
3. Create the following Tables and insert the shown data ( This table will be used in the subsequent Lab sessions )

**Department**

|  |  |  |
| --- | --- | --- |
| **Dept\_no** | **Dept\_name** | **location** |
| d1 | Research | Dallas |
| d2 | Accounting | Seattle |
| d3 | Marketing | Dallas |

**Employee**

|  |  |  |  |
| --- | --- | --- | --- |
| **emp\_no** | **emp\_fname** | **emp\_lname** | **dept\_no** |
| 25348 | Matthew | Smith | d3 |
| 10102 | Ann | Jones | d3 |
| 18316 | John | Barrimore | d1 |
| 29346 | James | James | d2 |

**Project**

|  |  |  |
| --- | --- | --- |
| **project\_no** | **project\_name** | **Budget** |
| p1 | Apollo | 120000 |
| p2 | Gemini | 95000 |
| p3 | Mercury | 185600 |

**Works\_on**

|  |  |  |  |
| --- | --- | --- | --- |
| **emp\_no** | **project\_no** | **Job** | **enter\_date** |
| 10102 | p1 | Analyst | 1997.10.1 |
| 10102 | p3 | manager | 1999.1.1 |
| 25348 | p2 | Clerk | 1998.2.15 |
| 18316 | p2 | NULL | 1998.6.1 |
| 29346 | p2 | NULL | 1997.12.15 |
| 2581 | p3 | Analyst | 1998.10.15 |
| 9031 | p1 | Manager | 198.4.15 |
| 28559 | p1 | NULL | 198.8.1 |
| 28559 | p2 | Clerk | 1992.2.1 |
| 9031 | p3 | Clerk | 1997.11.15 |
| 29346 | p1 | Clerk | 1998.1.4 |

**DML Exercise**

Modifying a Table’s Contents

1. Insert the data of a new employee called Julia Long, whose employee number is 1111. Her department number is not known yet.
2. Create a new table called **emp\_d1\_d2** with all employees who work for department d1 or d1, and load the corresponding rows from the **employee** table.
3. Modify the job of all the employees in project p1 who are managers. They have to work as clerks from now on.
4. The budgets of all projects are no longer determined. Assign all budgets the NULL value.
5. Increase the budget of the project where the manager has the employee number 10102. The increase is 10%.
6. Change the enter date for the projects for those employees who work in project p1 and belong to the department *Sales.* The new date is 12.12.1998.

Data Integrity Constraints

1. Alter the **Customers** and **Orders** tables, enhancing their definition with all primary and foreign keys constraints.
2. Using the ALTER TABLE statement, create an integrity constraint that limits the possible values of the quantity column in the **Orders** table to values between 1 and 30.
3. Create a table called **Regions** with the columns **City** and **Country** using the new data type for the later.
4. Add check constraint for city names as CA(for California), WA( for Washington), OR( for Oregon), and NM( for New Mexico). The default value is CA.
5. Delete the primary key of the **Customers** table. Why isn’t that working?
6. Delete the integrity constraint defined in Q2.

Simple Queries

1. Get all row of the **works\_on** table.
2. Get the employee numbers for all clerks
3. Get the employee numbers for employees working in project p2, and having employee numbers smaller than 10000. Solve this problem with two different but equivalent SELECT statements.
4. Get the employee numbers for all employees who didn’t enter their project in 1998.
5. Get the employee numbers for all employees who have a leading job( i.e., Analyst or Manager) in project p1
6. Get the enter dates for all employess in project p2 whose jobs have not been determined yet.
7. Get the employee numbers and last names of all employees whose first names contain two letter t’s.
8. Get the employee numbers and first names of all employees whose last names have a letter *o* or *a* as the second character and end with the letters *es.*
9. Get the employee numbers of all employees whose departments are located in Seattle.
10. Find the last and first names of all employess who entered their projects on 04.01.1998
11. Group all departments using their locations.
12. Find the biggest employee number.
13. Get the jobs that are done by more than two employees.
14. Find the employee numbers of all employees who are clerks or work for department d3.

Complex Queries

1. Create:
   1. Equijoin
   2. Natural join
   3. Cartesian product

forhe**project** and **works\_on** tables.

1. Get the employee numbers and job titles of all employees working on project Gemini
2. Get the first and last names of all employees that work for departments *Research* or *Acounting.*
3. Get the enter dates of all clerks that belong to the department d1.
4. Get the names of projects on which two or more clerks are working.
5. Get the first and last names of the employees that are manager and that work on project Mercury.
6. Get the first and last names of all employee who entered the project at the same time as at least one other employee.
7. Get the employee numbers of the employees living in the same location and belonging to the same department as one another.
8. Get the employee numbers of all employees belonging to the Marketing department.

Find two equivalent solutions using:

* 1. the JOIN operator
  2. The correlated subquery.

**Index , Views and Stored procedures**

1. Create a nonclusterd index for the **enter\_date** column of the **works\_on** table. Sixty percent of each index leaf page should be filled.
2. Create a unique composite index for the **l\_name** and **f\_name** columns of the **employee** table.
3. Create a view that comprises the data of all employees that work for the department d1.
4. For the **project** table, create a view that can be used by employees who are allowed to view all data of this table except the **budget** column.
5. Create a vew that comprises the first and last names of all employees who entered heir projects in the second half of the year 1988.
6. Solve the previous exercise so that the original columns **f\_name** and **l\_name**have new names in the view: **first** and **last**, respectively.
7. use the view in Exercise 3 to display full details of all employees whose last names begin with the letter M.
8. Create a view which comprises full details of all projects on which the employee named smith works .
9. Using the ALTER VIEW statement, modify the condition in the view in Exercise-3. The modified view should comprise the data of all employees that work either for the department d1 or d2, or both.
10. Using the view from Exercise 4, insert details of a new project with project no ‘p2’ and name ‘moon’
11. Create a view( with the WITH CHECK OPTION clause) that comprises the first and last names of all employees whose employee number is less than 10,000. After that, use he view to insert data for a new employee named Kohn with the employee number 22123, who works for the department d3.
12. Create a view(with the WITH CHECK OPTION clause) with full etails from the **works\_on** table for all employees that entered their projects during the years 1998 and 1999. After that, modify the entering date of the employee with the employee number 19346. The new date is 06/01/1997.
13. Solve the above excersise without the WITH CHECK OPTION clause and find the differences in relation to the modification of the data .
14. Create a stored procedure to insert data into department and Employee table.