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 |
| CompanyName varchar(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 |
| Orderdate datetime null |
| Shippeddate datetime null |
| Freight money null |
| Shipname varchar(40) null |
| Shipaddres varchar(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. Using the SQL Server Management Studio, try to instert a new row into the **Orders** table with the following values:

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

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 |

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.