“УКРАЇНСЬКИЙ КАТОЛИЦЬКИЙ УНІВЕРСИТЕТ”



**Звіт**

про виконання лабораторної роботи № 6

з курсу «Організація баз даних»

на тему «Побудова ER-діаграм в ER/Studio та генерування SQL-сценаріїв»

Варіант № 8

Виконав:

Студент групи ПКН15/Б

Карпінський Максим

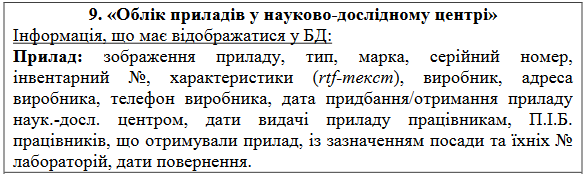
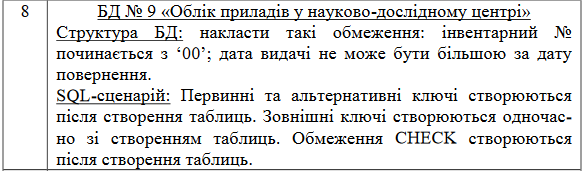
Перевірив:

Павельчак А.Г.

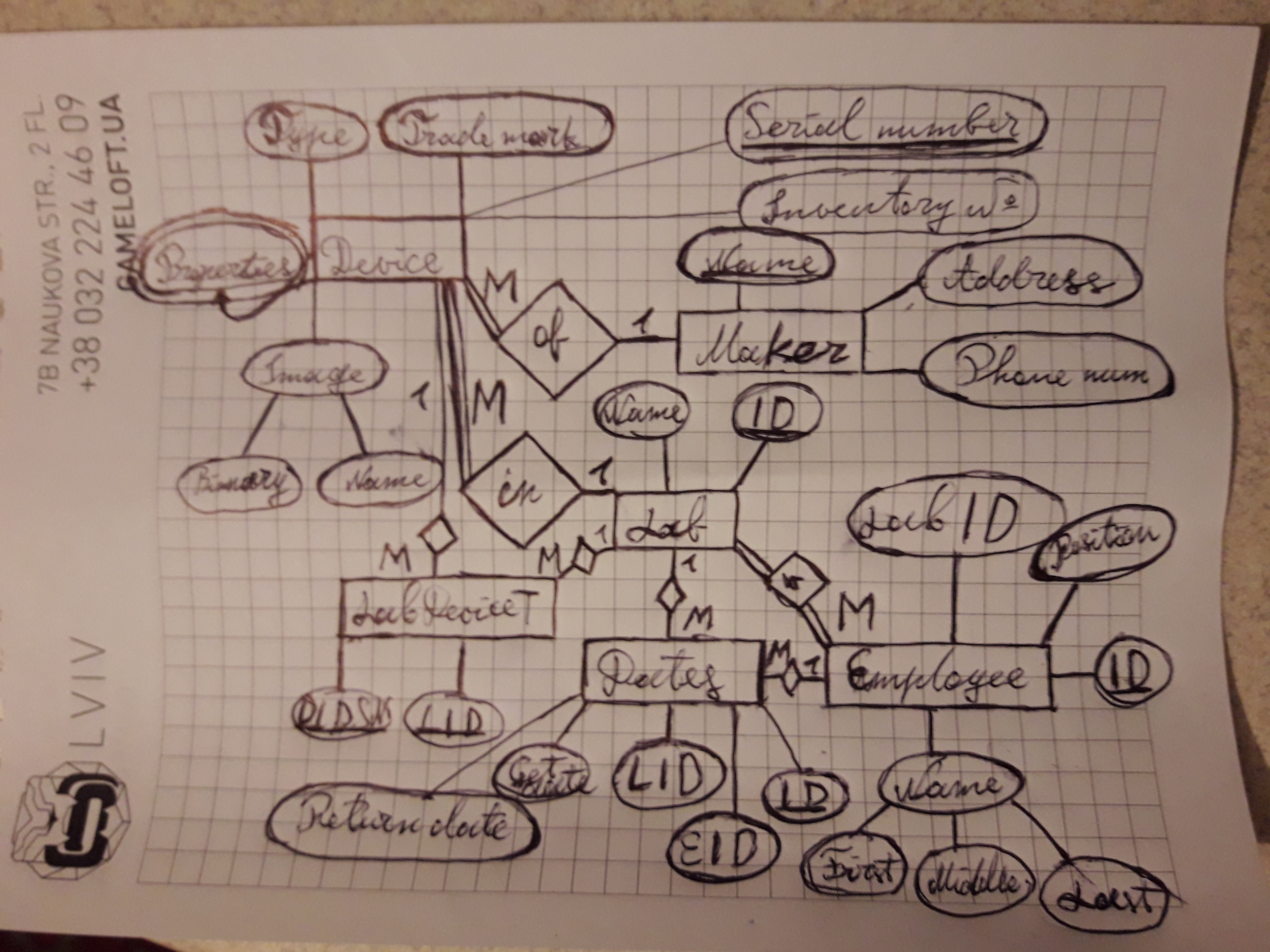
Львів - 2017

**Мета роботи:** навчитись проектувати ER-дмодель даних в нотації Пітера Чена, засобами CASE-засобу ER/Studio розробляти логічні ER-діаграми та генерувати фізичні, вивчити загальні принципи користування ER/Studio.

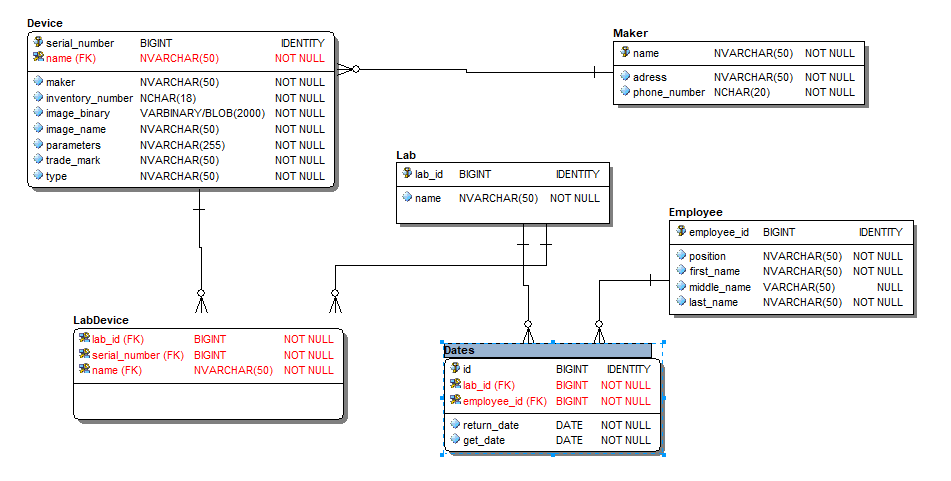
**Завдання до лабораторної роботи**



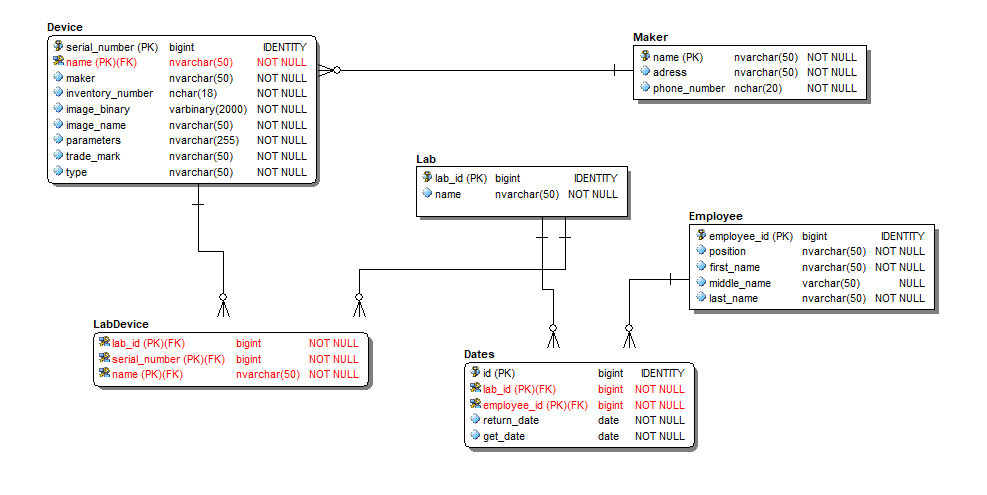
**ER-модель даних в нотації Пітера Чена**



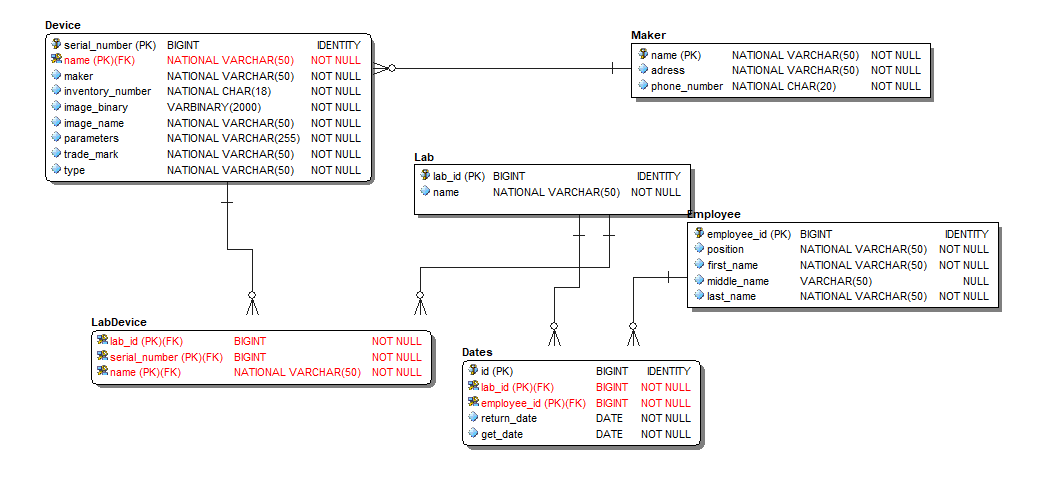
**Логічна ER-діаграма БД**



**Фізична ER-діаграма БД ( MS SQL Server)**



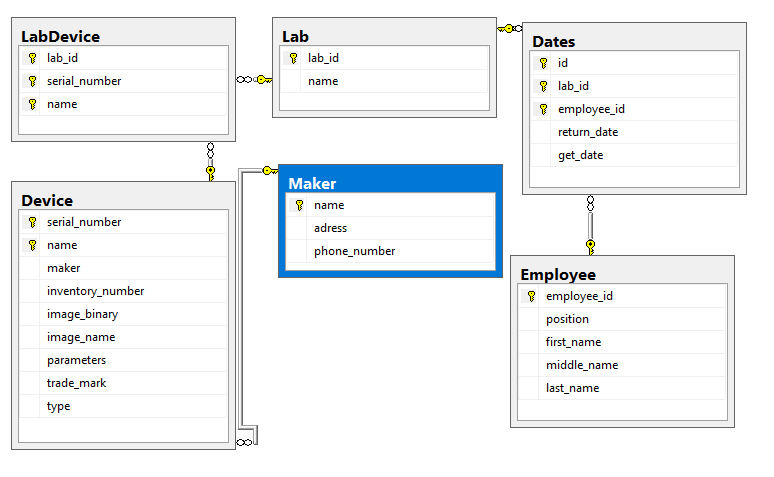
**Фізична ER-діаграма БД (MySQL)**



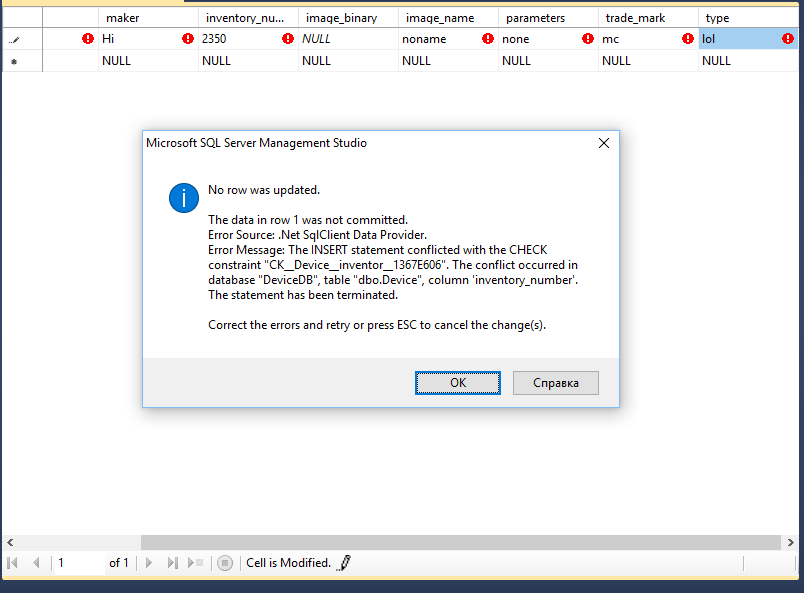
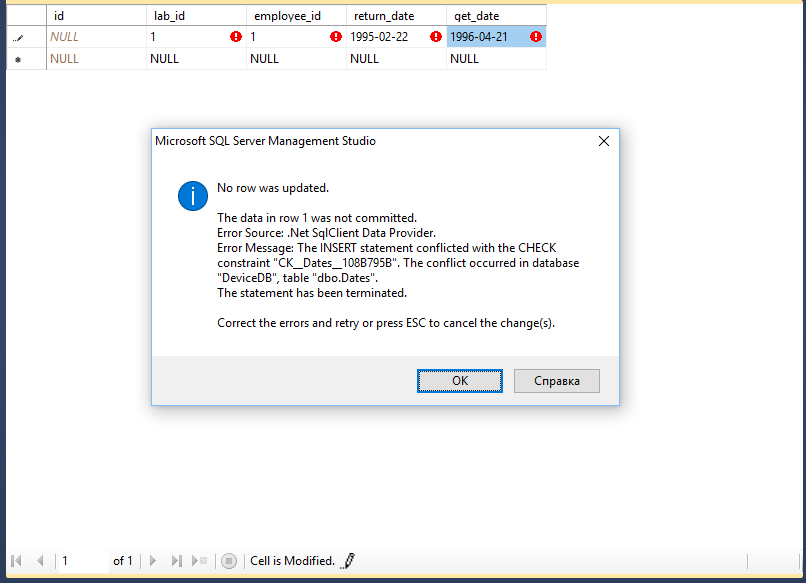
**Згенерований скрипт на MS SQL Server**

1. /\*
2. \* ER/Studio Data Architect SQL Code Generation
3. \* Project : Model1.DM1
4. \*
5. \* Date Created : Saturday, June 10, 2017 00:37:29
6. \* Target DBMS : Microsoft SQL Server 2014
7. \*/
8. USE master
9. go
10. CREATE DATABASE DeviceDB
11. go
12. USE DeviceDB
13. go
14. /\*
15. \* TABLE: Dates
16. \*/
17. CREATE TABLE Dates(
18. id bigint IDENTITY(1,1),
19. lab\_id bigint NOT NULL,
20. employee\_id bigint NOT NULL,
21. return\_date date NOT NULL,
22. get\_date date NOT NULL,
23. CONSTRAINT PK8 PRIMARY KEY NONCLUSTERED (id, lab\_id, employee\_id)
24. )
25. go
26. ALTER TABLE Dates
27. ADD CHECK (get\_date < return\_date);
28. IF OBJECT\_ID('Dates') IS NOT NULL
29. PRINT '<<< CREATED TABLE Dates >>>'
30. ELSE
31. PRINT '<<< FAILED CREATING TABLE Dates >>>'
32. go
33. /\*
34. \* TABLE: Device
35. \*/
36. CREATE TABLE Device(
37. serial\_number bigint IDENTITY(1,1),
38. name nvarchar(50) NOT NULL,
39. maker nvarchar(50) NOT NULL,
40. inventory\_number nchar(18) NOT NULL,
41. image\_binary varbinary(2000) NOT NULL,
42. image\_name nvarchar(50) NOT NULL,
43. parameters nvarchar(255) NOT NULL,
44. trade\_mark nvarchar(50) NOT NULL,
45. type nvarchar(50) NOT NULL,
46. CONSTRAINT PK4 PRIMARY KEY NONCLUSTERED (serial\_number, name)
47. )
48. go
49. ALTER TABLE Device
50. ADD CHECK (inventory\_number LIKE '00%');
51. IF OBJECT\_ID('Device') IS NOT NULL
52. PRINT '<<< CREATED TABLE Device >>>'
53. ELSE
54. PRINT '<<< FAILED CREATING TABLE Device >>>'
55. go
56. /\*
57. \* TABLE: Employee
58. \*/
59. CREATE TABLE Employee(
60. employee\_id bigint IDENTITY(1,1),
61. position nvarchar(50) NOT NULL,
62. first\_name nvarchar(50) NOT NULL,
63. middle\_name varchar(50) NULL,
64. last\_name nvarchar(50) NOT NULL,
65. CONSTRAINT PK9 PRIMARY KEY NONCLUSTERED (employee\_id)
66. )
67. go
68. IF OBJECT\_ID('Employee') IS NOT NULL
69. PRINT '<<< CREATED TABLE Employee >>>'
70. ELSE
71. PRINT '<<< FAILED CREATING TABLE Employee >>>'
72. go
73. /\*
74. \* TABLE: Lab
75. \*/
76. CREATE TABLE Lab(
77. lab\_id bigint IDENTITY(1,1),
78. name nvarchar(50) NOT NULL,
79. CONSTRAINT PK7 PRIMARY KEY NONCLUSTERED (lab\_id)
80. )
81. go
82. IF OBJECT\_ID('Lab') IS NOT NULL
83. PRINT '<<< CREATED TABLE Lab >>>'
84. ELSE
85. PRINT '<<< FAILED CREATING TABLE Lab >>>'
86. go
87. /\*
88. \* TABLE: LabDevice
89. \*/
90. CREATE TABLE LabDevice(
91. lab\_id bigint NOT NULL,
92. serial\_number bigint NOT NULL,
93. name nvarchar(50) NOT NULL,
94. CONSTRAINT PK6 PRIMARY KEY NONCLUSTERED (lab\_id, serial\_number, name)
95. )
96. go
97. IF OBJECT\_ID('LabDevice') IS NOT NULL
98. PRINT '<<< CREATED TABLE LabDevice >>>'
99. ELSE
100. PRINT '<<< FAILED CREATING TABLE LabDevice >>>'
101. go
102. /\*
103. \* TABLE: Maker
104. \*/
105. CREATE TABLE Maker(
106. name nvarchar(50) NOT NULL,
107. adress nvarchar(50) NOT NULL,
108. phone\_number nchar(20) NOT NULL,
109. CONSTRAINT PK5 PRIMARY KEY NONCLUSTERED (name)
110. )
111. go
112. IF OBJECT\_ID('Maker') IS NOT NULL
113. PRINT '<<< CREATED TABLE Maker >>>'
114. ELSE
115. PRINT '<<< FAILED CREATING TABLE Maker >>>'
116. go
117. /\*
118. \* TABLE: Dates
119. \*/
120. ALTER TABLE Dates ADD CONSTRAINT RefLab18
121. FOREIGN KEY (lab\_id)
122. REFERENCES Lab(lab\_id)
123. go
124. ALTER TABLE Dates ADD CONSTRAINT RefEmployee19
125. FOREIGN KEY (employee\_id)
126. REFERENCES Employee(employee\_id)
127. go
128. /\*
129. \* TABLE: Device
130. \*/
131. ALTER TABLE Device ADD CONSTRAINT RefMaker4
132. FOREIGN KEY (name)
133. REFERENCES Maker(name)
134. go
135. /\*
136. \* TABLE: LabDevice
137. \*/
138. ALTER TABLE LabDevice ADD CONSTRAINT RefLab16
139. FOREIGN KEY (lab\_id)
140. REFERENCES Lab(lab\_id)
141. go
142. ALTER TABLE LabDevice ADD CONSTRAINT RefDevice17
143. FOREIGN KEY (serial\_number, name)
144. REFERENCES Device(serial\_number, name)
145. go

**ER-діаграма розгорнутої БД на одному зі серверів (MS SQL Server)**



**Перевірки Check**



**Висновок:** Під час виконання практичного завдання я навчився проектувати ER-модель даних в нотації Пітера Чена, засобами CASE-засобу ER/Studio(Embarcadero) розробляти логічні ER-діаграми та генерувати фізичні, вивчив загальні принципи користування ER/Studio. Зрозумів принципи роботи і гнучкість у використанні ER/Studio.