Міністерство освіти і науки України

Одеський національний політехнічний університет

Iнститут компьютерних систем

Кафедра компьютерних інтеллектуальних систем і мереж

Розрахунково-графічна робота

З дисципліни: “Організація баз данних”

Предметна область «СТО»

Зробив

Ст. Борщов М.I.

Групи АМ-182

Перевiрили:

Головачова О.В.

Одеса 2020

**Вступ**

У наше століття, століття цифрових технологій ЕОМ грають найважливішу роль. Зараз в кожній організації - будь то державні установи або приватні фірми все комп'ютеризовано, це обумовлено дуже високою обчислювальною потужністю. Обчислення навіть найскладніших процесів і поставлених завдань виконується в найкоротші терміни, а фактор часу найчастіше грає найважливішу роль в більшості поставлених завдань. Обчислювальна потужність і обсяг пам'яті ЕОМ в останні роки стали неймовірно великі, а ціни на них істотно знизилися, це і сприяло масової комп'ютеризації всіх галузей діяльності людини. Зараз важко уявити життя без розумної машини, яка спрощує і прискорює величезну кількість поставлених завдань. Корисність комп'ютера зводиться нанівець при відсутності спеціалізованого програмного забезпечення, без якого «залізний помічник» не може нічого виконати. У цій роботі буде докладно описано шлях створення такої важливої в усіх організаціях і областях людства річ, яка зветься база даних. В даному конкретному випадку база даних автосервісу.

Основні ідеї сучасної технологій базуються на концепції, згідно з якою дані повинні бути організовані в бази даних (БД) з метою адекватного уявлення про реальний світ і використанні інформаційних ресурсів користувачів. Ці бази даних використовуються і управляються за допомогою спеціальних програмних комплексів.

База даних - це інформаційна модель, метою якої є впорядковане зберігання інформації.

Система управління базами даних - це популярний інструментальний засіб.

Це дозволяє автоматизувати бізнес-процеси з використанням інформаційних технологій, що дозволяють вирішувати проблеми зберігання та систематизації інформації відповідно до індивідуальних вимог компаній.  
**Опис предметної області**

Дана база даних призначена для організацій, що займаються будь-якими видами послуг з технічного обслуговування автомобілів (Автосервіс). У БД буде зберігатися інформація про:

1) Всі автомобілі, яккі коли-небудь знаходились в автосервісі;

2) Усіх клієнтів, які звертались/звертаються до автомайстерні, персональні знижки;

3) Працівників, їх обов’язки, цнформація щодо заробітної плати, соц.статусу;

4) Коштовне обладнання у майстерні;

5) Облік запчастин та матеріалів на складі.

В БД зберігається інформація про кожного власника, його ім’я, контактні данні, знижки.

Кожний автомобіль, якому оказуються послуги, має бути внесений до бд. Його основні дані, хто є власником, які роботи виконувались над ним, пробіг на час виконання робіт, дата та статус автомобілля на данну годину.

Коштовне обладнання, запчастини та матеріали повинні бути внесені до БД для їх обліку та розуміння того, що є у наявності, а що потрібно замовити. Також у базі данних зберігається інформація про постачальників, їх контактна інформація, адреса, поштовий адрес.

Інформація про працівників сервісу, які працюють або працювали у сервісі, повинна зберігатись у таблиці. Паспортні дані, адреса, обов’яки, соціальний статус, дату прийняття на роботу для рахування стажу.

При роботі автомайстерні ведеться облік виконаних робіт. Майстер вказує що він родить з певним авто та дату виконання роботи.

Повинна бути створена окрема таблиця, у якій вказані усі роботи, які виконуються у сервісі. Їх ціну, необхідні деталі, гарантію на роботу.

Ціна за роботу фіксована, зберігається у відповідній таблиці. При ремонті ціна обговорюється індивідуально, підсумовується ціна деталей і за роботу.

Зарплата рахується погодинно.

Майстерня надає повний спектр послуг:

* Заміна запчастин та росхідників;
* Ремонт / заміна деталей в двигуні / підвісці / коробці передач;
* Обслуговування салону;
* Ремонт кузова і кузовних деталей.
* Для клієнтів будуть існувати знижки.

База даних буде використана для отримання і зберігання інформації:

* Ремонт / обслуговування транспортних засобів;
* Облік обладнання в майстерні;
* Клієнтська база;
* Співробітники сервісу;
* Деталі, що перебувають у складі;
* Список постачальників.

**Реляційна модель**

У базі даних можна виділити наступні основні об'єкти:

1) Авто. Власне автомобіль який приїжджає на ремонт/обслуговування

2) Клієнт. Власник авто

3) Персонал. Люди які працюють на сто. Потрібно вести облік їх роботи та зберігати інформацію стосовно заробітної плати

4) Інструмент та обладнання. Які використовуються в майстерні. Потрібно для податкової

5) Склад. Місце де зберігаються деталі. В БД зберігаєся інформація про кількість, дату надходження, постачальника

6) Обслуговування. Для зберігання інформації ХТО, КОЛИ, ЩО і з яким автомобілем робив.

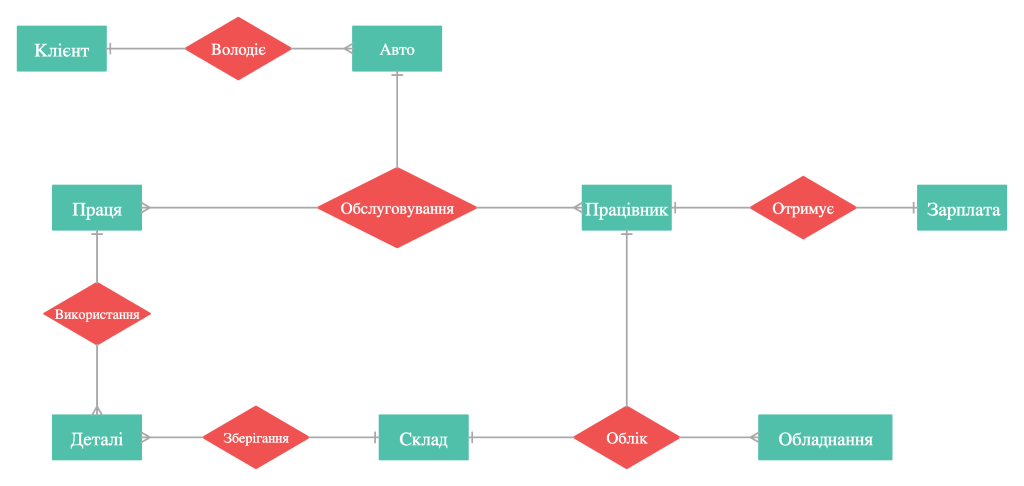
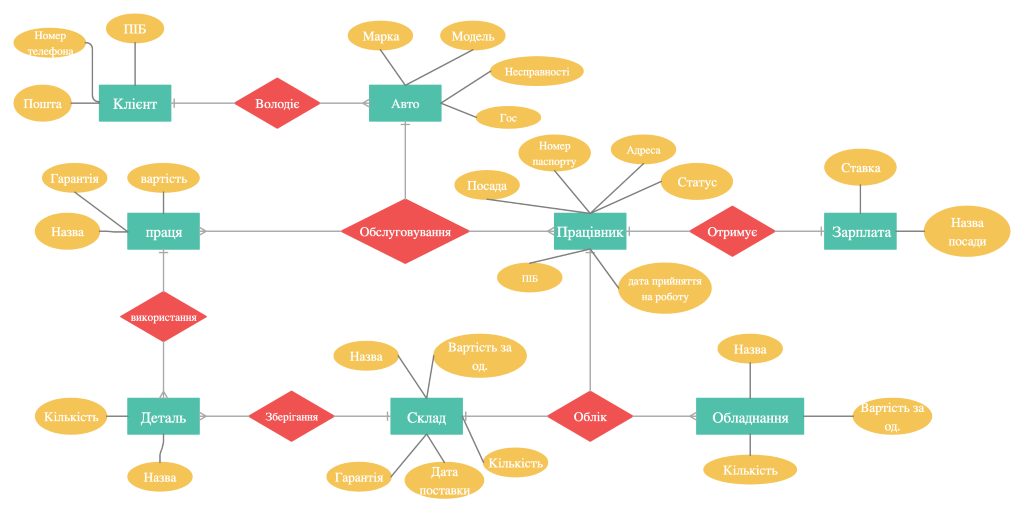
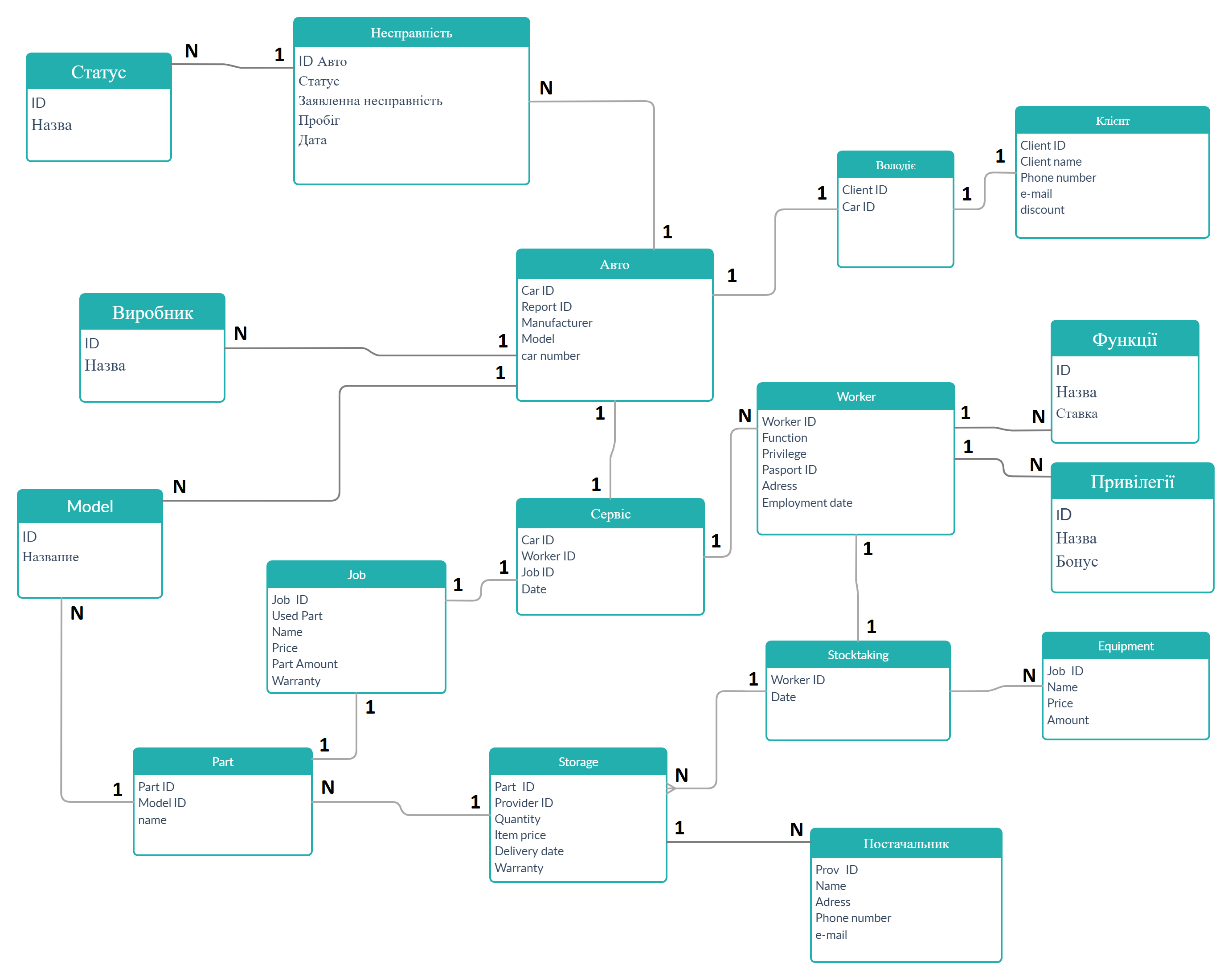


Схема Сутність-зв'язок



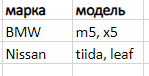
Сутність-зв'язок з атрибутами

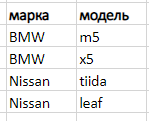


Логічний рівень

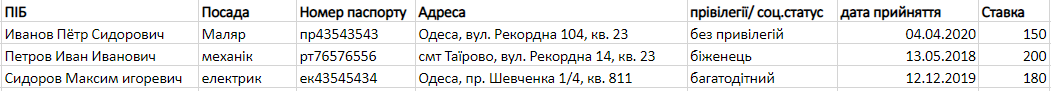
**Привидення до 3 НФ**

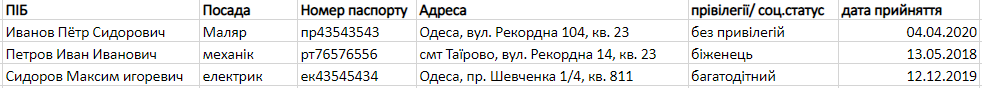
Спочатку приведемо до 1 нф





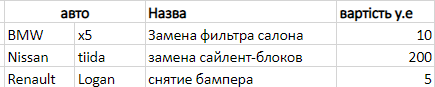
Далі до другої:

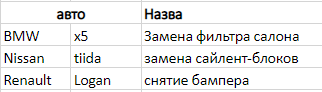


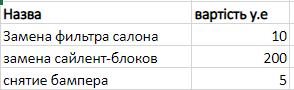




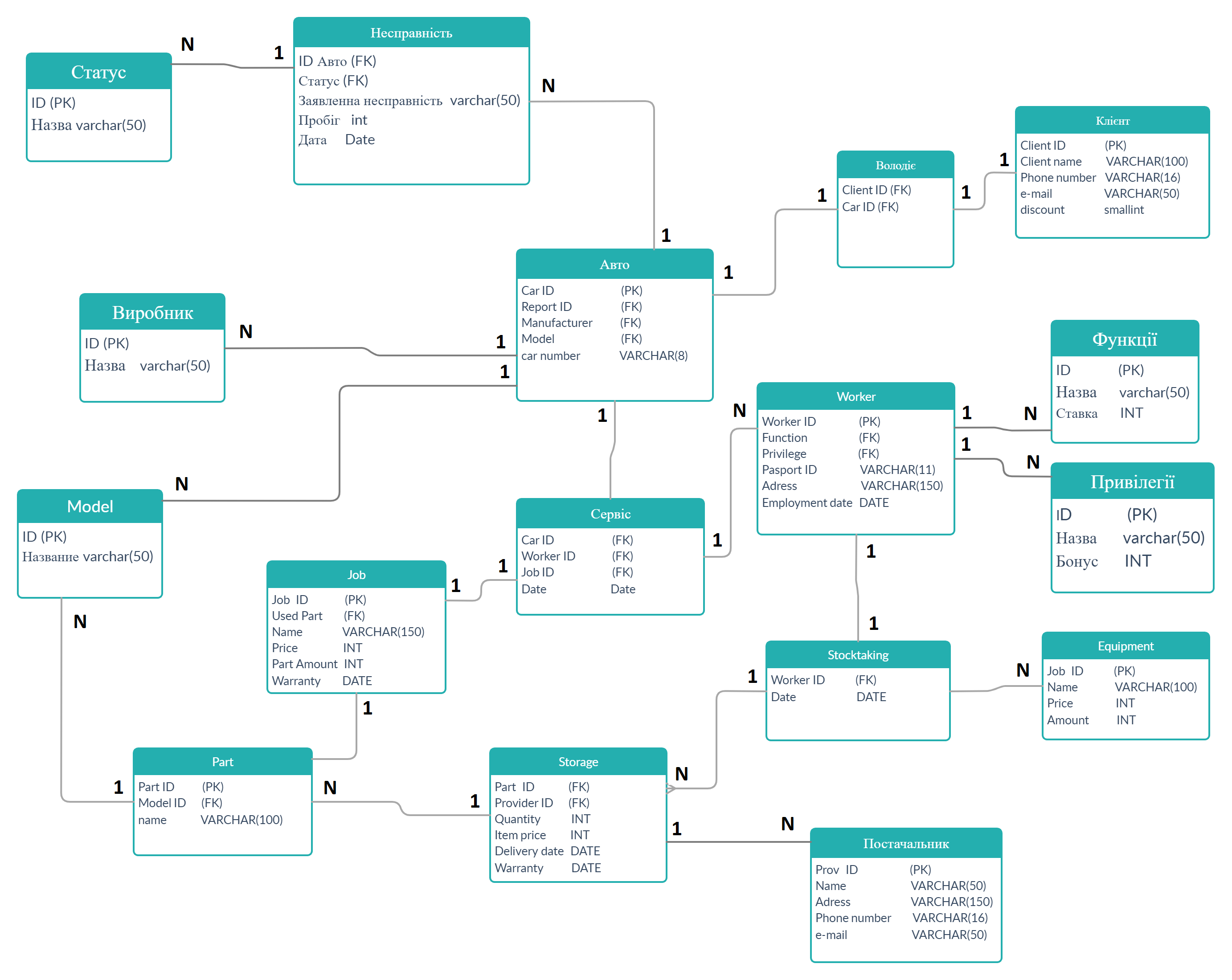
Та до третьої:







**ФІзичний рівень**



Фізичний рівень

При створенні фізичного рівня потрібно задати усі необхідні параметри до усіх атрибутів усіх таблиць.

Для атрибутів, де зберігаються назви, текстові поля або імена використовується VARCHAR до 50ти символів. Виключення: назва запчастини (до 100ти символів), назва послуги та адреси до 150ти символів, поле для номеру паспорту - 11, для номерів телефонів - 16, для номерів авто - 8.

Усі ключі встановюються автоматично. Використовується INT.

Для атрибуту “знижка” у клієнта використовуються SMALLINT.

Для дат - DATE, для усіх чисел INT.

**Створення таблиць для бази даних та їх заповнення**

Результат створення таблиць та занесення данних до них прикріплен у Додатку А.

Створюємо базу данних та обираємо її для роботи:

create database sto\_bd;

use sto\_bd;

Створення таблиць та заповнення їх данними

CREATE TABLE auto (

car\_id int(11) NOT NULL,

car\_brand\_id int(11) NOT NULL,

car\_model\_id int(11) NOT NULL,

plate\_number varchar(8) DEFAULT NULL,

client\_id int(11) NOT NULL

) ;

CREATE TABLE client (

client\_id int(11) NOT NULL,

full\_name varchar(100) DEFAULT NULL,

phone varchar(16) DEFAULT NULL,

email varchar(50) NOT NULL,

discount smallint(4) NOT NULL DEFAULT '0'

) ;

CREATE TABLE defect (

reported\_malfunction varchar(100) DEFAULT NULL,

mileage int(11) DEFAULT NULL,

car\_id int(11) DEFAULT NULL,

date date DEFAULT NULL,

status\_id int(11) DEFAULT NULL

) ;

CREATE TABLE employee (

employee\_id int(11) NOT NULL,

employee\_name varchar(100) NOT NULL,

functions tinyint(4) DEFAULT NULL,

passport varchar(11) DEFAULT NULL,

adress varchar(150) DEFAULT NULL,

privilege tinyint(4) NOT NULL DEFAULT '0',

employment\_date date NOT NULL

) ;

CREATE TABLE equipment (

equipment\_id int(11) NOT NULL,

name varchar(100) DEFAULT NULL,

amount int(11) NOT NULL DEFAULT '0',

price int(11) NOT NULL DEFAULT '0'

) ;

CREATE TABLE functions (

functions\_id tinyint(4) NOT NULL,

name varchar(50) DEFAULT NULL,

rate smallint(6) DEFAULT NULL

) ;

CREATE TABLE job (

job\_id int(11) NOT NULL,

name varchar(150) DEFAULT NULL,

price int(11) DEFAULT NULL,

used\_part\_id int(11) DEFAULT NULL,

part\_amount int(11) DEFAULT NULL,

warranty tinyint(4) DEFAULT NULL

) ;

CREATE TABLE manufacturer (

manufactures\_id int(11) NOT NULL,

name varchar(50) DEFAULT NULL

) ;

CREATE TABLE model (

model\_id int(11) NOT NULL,

name varchar(50) NOT NULL

) ;

CREATE TABLE part (

part\_id int(11) NOT NULL,

name varchar(100) DEFAULT NULL,

model\_id int(11) DEFAULT NULL

) ;

CREATE TABLE privilege (

privilege\_id tinyint(11) NOT NULL,

name varchar(50) NOT NULL,

salary\_increase tinyint(4) NOT NULL DEFAULT '0'

) ;

CREATE TABLE provider (

provider\_id int(11) NOT NULL,

name varchar(50) DEFAULT NULL,

adress varchar(150) DEFAULT NULL,

phone varchar(16) DEFAULT NULL,

email varchar(50) DEFAULT NULL

) ;

CREATE TABLE service (

car\_id int(11) DEFAULT NULL,

job\_id int(11) DEFAULT NULL,

employee\_id int(11) DEFAULT NULL,

date date DEFAULT NULL

) ;

CREATE TABLE status (

status\_id int(11) NOT NULL,

name varchar(50) DEFAULT NULL

) ;

CREATE TABLE storage (

id\_part int(11) NOT NULL,

amount int(11) NOT NULL,

price int(11) NOT NULL DEFAULT '0',

receipt\_date date NOT NULL,

provider\_id int(11) DEFAULT NULL,

warranty tinyint(4) DEFAULT NULL

) ;

**Задання усіх ключей та офосбливих полів:**

ALTER TABLE auto

ADD PRIMARY KEY ( car\_id ),

ADD KEY car\_model\_id ( car\_model\_id ),

ADD KEY client\_id ( client\_id ),

ADD KEY car\_brand\_id ( car\_brand\_id ) USING BTREE;

ALTER TABLE client

ADD PRIMARY KEY ( client\_id );

ALTER TABLE defect

ADD KEY car\_id ( car\_id ),

ADD KEY status\_id ( status\_id );

ALTER TABLE employee

ADD PRIMARY KEY ( employee\_id ),

ADD KEY functions ( functions ),

ADD KEY privilege ( privilege );

ALTER TABLE equipment

ADD PRIMARY KEY ( equipment\_id );

ALTER TABLE functions

ADD PRIMARY KEY ( functions\_id );

ALTER TABLE job

ADD PRIMARY KEY ( job\_id ),

ADD KEY used\_part\_id ( used\_part\_id );

ALTER TABLE manufacturer

ADD PRIMARY KEY ( manufactures\_id );

ALTER TABLE model

ADD PRIMARY KEY ( model\_id );

ALTER TABLE part

ADD PRIMARY KEY ( part\_id ),

ADD KEY model\_id ( model\_id );

ALTER TABLE privilege

ADD PRIMARY KEY ( privilege\_id );

ALTER TABLE provider

ADD PRIMARY KEY ( provider\_id );

ALTER TABLE service

ADD KEY car\_id ( car\_id , job\_id , employee\_id ),

ADD KEY job\_id ( job\_id ),

ADD KEY employee\_id ( employee\_id );

ALTER TABLE status

ADD PRIMARY KEY ( status\_id );

ALTER TABLE storage

ADD KEY id\_part ( id\_part ),

ADD KEY provider\_id ( provider\_id );

ALTER TABLE auto

MODIFY car\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

ALTER TABLE client

MODIFY client\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=21;

ALTER TABLE employee

MODIFY employee\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=21;

ALTER TABLE equipment

MODIFY equipment\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=7;

ALTER TABLE functions

MODIFY functions\_id tinyint(4) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=7;

ALTER TABLE job

MODIFY job\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=41;

ALTER TABLE manufacturer

MODIFY manufactures\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

ALTER TABLE model

MODIFY model\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=17;

ALTER TABLE part

MODIFY part\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

ALTER TABLE privilege

MODIFY privilege\_id tinyint(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=9;

ALTER TABLE provider

MODIFY provider\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

ALTER TABLE status

MODIFY status\_id int(11) NOT NULL AUTO\_INCREMENT, AUTO\_INCREMENT=11;

ALTER TABLE auto

ADD CONSTRAINT auto\_ibfk\_1 FOREIGN KEY ( client\_id ) REFERENCES client ( client\_id ),

ADD CONSTRAINT auto\_ibfk\_2 FOREIGN KEY ( car\_model\_id ) REFERENCES model ( model\_id ),

ADD CONSTRAINT auto\_ibfk\_3 FOREIGN KEY ( car\_brand\_id ) REFERENCES manufacturer ( manufactures\_id );

ALTER TABLE defect

ADD CONSTRAINT defect\_ibfk\_1 FOREIGN KEY ( status\_id ) REFERENCES status ( status\_id ),

ADD CONSTRAINT defect\_ibfk\_2 FOREIGN KEY ( car\_id ) REFERENCES auto ( car\_id );

ALTER TABLE employee

ADD CONSTRAINT employee\_ibfk\_1 FOREIGN KEY ( privilege ) REFERENCES privilege ( privilege\_id ),

ADD CONSTRAINT employee\_ibfk\_2 FOREIGN KEY ( functions ) REFERENCES functions ( functions\_id );

ALTER TABLE job

ADD CONSTRAINT job\_ibfk\_1 FOREIGN KEY ( used\_part\_id ) REFERENCES part ( part\_id );

ALTER TABLE part

ADD CONSTRAINT part\_ibfk\_1 FOREIGN KEY ( model\_id ) REFERENCES model ( model\_id );

ALTER TABLE service

ADD CONSTRAINT service\_ibfk\_1 FOREIGN KEY ( job\_id ) REFERENCES job ( job\_id ),

ADD CONSTRAINT service\_ibfk\_2 FOREIGN KEY ( car\_id ) REFERENCES auto ( car\_id ),

ADD CONSTRAINT service\_ibfk\_3 FOREIGN KEY ( employee\_id ) REFERENCES employee ( employee\_id );

ALTER TABLE storage

ADD CONSTRAINT storage\_ibfk\_1 FOREIGN KEY ( id\_part ) REFERENCES part ( part\_id ),

ADD CONSTRAINT storage\_ibfk\_2 FOREIGN KEY ( provider\_id ) REFERENCES provider ( provider\_id );

COMMIT;

**Введення даних**

INSERT INTO auto ( car\_id , car\_brand\_id , car\_model\_id , plate\_number , client\_id ) VALUES

(1, 8, 4, 'DF5434GF', 13),

(2, 6, 9, 'HN7698LK', 19),

(3, 3, 3, 'OI8978NN', 13),

(4, 9, 8, 'OI5787JU', 17),

(5, 10, 10, 'UY8667KH', 14),

(6, 2, 2, 'NB7645HG', 20),

(7, 5, 7, 'MN8967NV', 11),

(8, 1, 1, 'LK8978LK', 17),

(9, 4, 5, 'GF7658JY', 12),

(10, 7, 6, 'JH7564LK', 16);

INSERT INTO client ( client\_id , full\_name , phone , email , discount ) VALUES

(11, 'Colin Anderson', '38(075)157-08-85', '[example1@mail.com](mailto:example1@mail.com)', 5),

(12, 'Brandon Arnold', '38(075)418-79-31', '[example2@mail.com](mailto:example2@mail.com)', 0),

(13, 'Connor Edwards', '38(075)895-77-05', '[example3@mail.com](mailto:example3@mail.com)', 0),

(14, 'Ambrose White', '38(082)708-30-97', '[example4@mail.com](mailto:example4@mail.com)', 0),

(15, 'Katelyn Jones', '38(082)344-88-06', '[example5@mail.com](mailto:example5@mail.com)', 0),

(16, 'William Mason', '38(099)662-32-87', '[example6@mail.com](mailto:example6@mail.com)', 0),

(17, 'Ashley Jenkins', '38(099)664-99-49', '[example7@mail.com](mailto:example7@mail.com)', 5),

(18, 'Hunter Thomas', '38(099)470-80-64', '[example8@mail.com](mailto:example8@mail.com)', 0),

(19, 'Richard Garcia', '38(068)151-68-51', '[example9@mail.com](mailto:example9@mail.com)', 0),

(20, 'Arianna Torres', '38(068)594-66-02', 'example10@mail.com', 3);

INSERT INTO defect ( reported\_malfunction , mileage , car\_id , date , status\_id ) VALUES

('Reinstall front window', 54323, 1, '2020-05-01', 2),

('Loose enjine cover', 342657, 1, '2019-12-12', 10),

('Loose brakes', 568643, 5, '2020-04-20', 4),

('Broke left door', 43576, 5, '2020-04-08', 9),

('Reinstall stop light', 78567, 8, '2020-04-21', 1),

('Change right door', 87656, 5, '2020-05-27', 5),

('Repair right front door', 98756, 8, '2020-04-17', 7),

('Uninsall left front door', 76431, 10, '2020-04-22', 10),

('I broke right back passenger door', 76545, 6, '2020-05-12', 6),

('Install new front passenger seat', 754657, 5, '2020-04-02', 8);

INSERT INTO employee ( employee\_id , employee\_name , functions , passport , adress , privilege , employment\_date ) VALUES

(11, 'Porter Wright', 3, 'JH876980', 'Odessa, Levitana st.54. ap.65', 8, '2020-03-03'),

(12, 'Presley Sanchez', 2, 'MK987865', 'Odessa, Glushko st,32. ap.6', 5, '2020-04-16'),

(13, 'Ronan White', 6, 'ER537656', 'Odessa, Bunina st,87. ap.26', 8, '2019-07-01'),

(14, 'Ramon Scott', 3, 'UY479777', 'smt. Tairovo, Radista st,12', 8, '2020-03-11'),

(15, 'Irvin Carter', 1, 'YT456523', 'v. Novi Sangary, Karantynna 43, ap.3', 5, '2019-11-15'),

(16, 'Knox Henderson', 5, 'NB968498', 'Lviv, Europeyskaya 43, ap.76', 2, '2020-03-27'),

(17, 'Nelson Kelly', 2, 'JU567254', 'Odessa, Bunina st,57. ap.66', 5, '2020-03-31'),

(18, 'Brandon Campbell', 5, 'UJ567609', 'smt. Tairovo, Radista st.43, ap.22', 2, '2020-04-15'),

(19, 'Finnley Scott', 4, 'YU836712', 'Odessa, Shevchenka av, 1/4. ap.811', 1, '2019-11-13'),

(20, 'Louis Richardson', 1, 'GR431454', 'Odessa, Glushko st,234. ap.68', 6, '2019-08-01');

INSERT INTO equipment ( equipment\_id , name , amount , price ) VALUES

(1, 'Compressor AC-40', 1, 450),

(2, 'Compressor AC-50', 1, 550),

(3, 'Laptop Asus J751', 1, 750),

(4, 'Computer Imac Pro 2019', 1, 5000),

(5, 'Lifter Compas RE-200', 2, 1200),

(6, 'Linux Server IBM-2000', 1, 3000);

INSERT INTO functions ( functions\_id , name , rate ) VALUES

(1, 'Painter', 15),

(2, 'mechanic', 13),

(3, ' bodybuilder', 15),

(4, 'engine builder', 18),

(5, 'manager', 10),

(6, 'director', 20);

INSERT INTO job ( job\_id , name , price , used\_part\_id , part\_amount , warranty ) VALUES

(31, 'Replase back stop light', 10, 5, 1, 12),

(32, 'Install new engine cover', 25, 3, 1, NULL),

(33, 'Install new rear bumper', 150, 1, 1, 12),

(34, 'Replase passenger door', 50, 4, 1, 12),

(35, 'Replase Front passenger seat', 30, 6, 1, 6),

(36, 'Unplug front window', 15, NULL, 0, 24),

(37, 'Repair Right passenger door', 24, 8, 0, 12),

(38, 'Paint body', 500, NULL, 2, 36),

(39, 'Restore body geometry', 2000, NULL, NULL, 12),

(40, 'Glue front window', 100, 7, 1, 24);

INSERT INTO manufacturer ( manufactures\_id , name ) VALUES

(1, 'Renault'),

(2, 'Nissan'),

(3, 'Lamborghini'),

(4, 'Opel'),

(5, 'Honda'),

(6, 'Doodge'),

(7, 'Mitsubishi'),

(8, 'BMW'),

(9, 'Mercedes'),

(10, 'Smart');

INSERT INTO model ( model\_id , name ) VALUES

(1, 'Kajar'),

(2, 'Tiida'),

(3, 'Huracan'),

(4, 'X5M'),

(5, 'Astra'),

(6, 'Solaris'),

(7, 'Civic'),

(8, 'ML200'),

(9, 'Charger'),

(10, '2101'),

(11, 'AE&T 4A 380B F4D-4'),

(12, 'AC-50'),

(13, 'AE&T 4A 400B F4D-4'),

(14, 'AC-70'),

(15, 'J751'),

(16, 'Alienweir 17 b3');

INSERT INTO part ( part\_id , name , model\_id ) VALUES

(1, 'rear bumper', 1),

(2, 'front lef light', 2),

(3, 'engine cover', 9),

(4, 'left passenger door', 10),

(5, 'back stop light', 5),

(6, 'front passenger seat', 3),

(7, 'front window', 8),

(8, 'right front passenger door', 6),

(9, 'right back passenger door', 5),

(10, 'left passenger door', 1);

INSERT INTO privilege ( privilege\_id , name , salary\_increase ) VALUES

(1, 'Large family', 10),

(2, 'Disabled person of the first group', 5),

(3, 'Disabled person of the second group', 8),

(4, 'Disabled person of the third group', 10),

(5, 'Student', -5),

(6, 'Best worker', 5),

(7, ' refugee', 3),

(8, 'Without privileges', 0);

INSERT INTO provider ( provider\_id , name , adress , phone , email ) VALUES

(1, 'AD Avanta', 'Odessa, Glushko st,30. of.1', '38(075)157-08-85', '[example1@mail.com](mailto:example1@mail.com)'),

(2, 'Forward Autoparts', 'Odessa, Glushko st,30. of.2', '38(075)418-79-31', '[example2@mail.com](mailto:example2@mail.com)'),

(3, 'Alluniparts', 'Odessa, Glushko st,30. of.3', '38(078)395-55-99', '[example3@mail.com](mailto:example3@mail.com)'),

(4, 'AUTOLANS', 'Odessa, Glushko st,30. of.4', '38(078)885-55-55', '[example4@mail.com](mailto:example4@mail.com)'),

(5, 'MYSTORE1', 'Odessa, Glushko st,30. of.5', '38(085)859-15-57', '[example5@mail.com](mailto:example5@mail.com)'),

(6, 'JpAuto', 'Odessa, Glushko st,31. of.1', '38(085)254-26-88', '[example6@mail.com](mailto:example6@mail.com)'),

(7, 'ROSSKO', 'Odessa, Glushko st,31. of.2', '38(085)576-45-08', '[example7@mail.com](mailto:example7@mail.com)'),

(8, 'AUTO-RAY', 'Odessa, Glushko st,31. of.3', '38(085)137-77-06', '[example8@mail.com](mailto:example8@mail.com)'),

(9, 'AVTOGOROD.PRO', 'Odessa, Glushko st,31. of.4', '38(085)525-07-25', '[example9@mail.com](mailto:example9@mail.com)'),

(10, ' AMX24', 'Odessa, Glushko st,31. of.5', '38(085)771-37-47', '[example10@mail.com](mailto:example10@mail.com)');

INSERT INTO service ( car\_id , job\_id , employee\_id , date ) VALUES

(2, 31, 15, '2020-06-02'),

(1, 34, 18, '2019-11-11'),

(10, 36, 15, '2019-12-10'),

(3, 40, 17, '2020-04-07'),

(9, 31, 14, '2020-03-18'),

(8, 33, 14, '2020-04-06'),

(4, 40, 13, '2020-04-07'),

(5, 37, 13, '2020-04-08'),

(7, 33, 20, '2020-05-12'),

(6, 36, 17, '2020-04-10'),

(2, 31, 15, '2020-06-02'),

(1, 34, 18, '2019-11-11'),

(10, 36, 15, '2019-12-10'),

(3, 40, 17, '2020-04-07'),

(9, 31, 14, '2020-03-18'),

(8, 33, 14, '2020-04-06'),

(4, 40, 13, '2020-04-07'),

(5, 37, 13, '2020-04-08'),

(7, 33, 20, '2020-05-12'),

(6, 36, 17, '2020-04-10');

INSERT INTO status ( status\_id , name ) VALUES

(1, 'In process'),

(2, 'Accepted'),

(3, 'On diagnosis'),

(4, ' Awaiting Details'),

(5, 'Decliend'),

(6, 'Done, waiting for payment'),

(7, 'Done, paid'),

(8, 'As agreed'),

(9, 'Car is with the owner'),

(10, 'Done, issued to owner');

INSERT INTO storage ( id\_part , amount , price , receipt\_date , provider\_id , warranty ) VALUES

(1, 10, 15, '2019-11-16', 5, 12),

(2, 8, 25, '2020-03-25', 4, 24),

(3, 21, 10, '2019-09-05', 8, 6),

(4, 5, 20, '2020-02-10', 1, 24),

(5, 10, 20, '2019-11-16', 9, 24),

(6, 3, 22, '2019-10-31', 2, 24),

(7, 5, 100, '2020-03-11', 1, 36),

(8, 0, 20, '2020-04-15', 7, 24),

(9, 11, 25, '2020-02-12', 10, 24),

(10, 10, 25, '2019-10-02', 1, 24);

**Отримання інформації**

Результати усіх запитів наведені у Додатку Б.

Код:

1) SELECT P.name, MAX(S.amount) FROM storage S INNER JOIN part P ON P.part\_id = S.id\_part;

2) SELECT P.name, MIN( S.amount) FROM storage S INNER JOIN part P ON P.part\_id = S.id\_part;

3) SELECT P.name AS partname, M.name AS modelname, S.amount AS amount FROM storage S, model M, part P WHERE M.model\_id = P.model\_id AND P.part\_id = S.id\_part ;

4) SELECT P.name AS partname, M.name AS modelname, S.amount AS amount FROM storage S, model M, part P WHERE M.model\_id = P.model\_id AND P.part\_id = S.id\_part ;

5) SELECT SUM(price) FROM job WHERE 1;

6) SELECT AVG(price) FROM job WHERE 1;

7) SELECT name FROM manufacturer WHERE name LIKE '%MW';

8) SELECT name FROM manufacturer WHERE name LIKE '%ss%';

9) SELECT \* FROM job WHERE price BETWEEN 50 AND 450;

10) SELECT \* FROM functions WHERE name REGEXP("[GJ]");

11) SELECT \* FROM service ORDER BY date DESC;

12) SELECT COUNT(S.id\_part), P.name FROM storage S LEFT JOIN part P ON P.part\_id = S.id\_part GROUP BY S.id\_part;

13) SELECT \* FROM provider WHERE email LIKE '%ample%' ;

14) SELECT MIN(discount) FROM client;

15) SELECT MAX(discount) FROM client;

16) UPDATE employee SET employee\_name = 'Burble Margl' WHERE employee\_name = ‘Nelson Kelly’;

17) DELETE FROM equipment WHERE equipment\_id=3;

18) DELETE FROM equipment WHERE price=450;

19) UPDATE equipment SET name=’OUT OF STOCK’ WHERE price=’1200’;

20) UPDATE function SET rate=5000 WHERE name=’director’;

**Висновки**

Виконуючи цю роботу я навчився усім крокам створення бази даних. Описувати предметну область. Створювати реляційну модель та схему «сутність-зв’язок». Виконувати логічний рівень проектування бази даних та опису переходу до 3НФ, формування таблиці для кожної Сутності і кожного Зв'язку, задання для кожної таблиці ключів (первинних та зовнішніх), визначити тип кожного зв'язку. Створювати таблиці та атрибути у них, задавати тип данних та обмеження. Реалізовувати структуру БД та її вивід. Заповнювати таблиці даними та виконувати запити (прості і вкладені).

База даних для автомайстерні успішно створена та функціонує, тож завдання по розробці виконано успішно та вчасно.

**Список використаних джерел**

1) <https://el.opu.ua/course/view.php?id=611>

(Усi лекції)

2) <https://dev.mysql.com/doc/>

(Документацiя MySQL, для розгляду прикладiв рiзних запитiв)

3) <https://www.w3schools.com/sql/sql_union.asp>

4) <https://www.sqlshack.com/sql-overview-usage-and-examples/>6) <https://study.com/academy/lesson/regular-expressions-in-sql-server-databases-implementation-use.html>

5) <https://www.geeksforgeeks.org/mysql-regular-expressions-regexp/>

6) <https://dev.mysql.com/doc/refman/8.0/en/sql-data-definition-statements.html>

7) <https://dev.mysql.com/doc/refman/8.0/en/non-typed-operators.html>

8)<http://www.mysql.ru/docs/gruber/mg03.html>