

**Итоговый проект по курсу
«Системы управления базами данных»
Тема: «Разработка базы данных аптеки»**

Выполнила
Севрюк Александра

- Перед началом разработки базы данных был проведен **анализ предметной области** предприятия аптечной сети.
- В результате проведенного анализа было выделено **семь объектов**
 - оптовые заказы,
 - товары,
 - продажи,
 - поставки,
 - контрагенты,
 - страны,
 - места хранения,которые соответствуют **7 таблицам** в разрабатываемой базе данных.
- В итоге было определено, что **целью** является автоматизация процессов сбора, обработки, хранения и анализа информации о продажах, поставках, контрагентах и местах хранения товаров, реализуемых в аптеке.

Описание таблиц

Таблица Products (Товары). В данной таблице будет храниться информация о товарах аптеки.

поле	тип поля	описание
ProductID	integer	уникальный идентификатор товара
VendoreCode	integer	артикул товара
ProductName	varchar(50)	наименование товара
Units	varchar(10)	единицы измерения товара
Price	decimal	цена товара
VacationConditions	varchar(20)	условия отпуска товара

Таблица Warehouses (Места хранения). В данной таблице будет храниться информация о местах хранения (складах) товаров аптеки.

поле	тип поля	описание
WarehouseID	integer	уникальный идентификатор места хранения
WarehouseName	varchar(50)	наименование места хранения
Location	varchar(50)	расположение места хранения
FullNameFinRestPerson	varchar(50)	ФИО материально ответственного лица
Phone	char(15)	телефон места хранения

Таблица Sales (Продажи). В данной таблице будет храниться информация о продажах товаров аптеки.

поле	тип поля	описание
WaybillID	integer	код накладной продажи
SaleDate	date	дата продажи
Counderparty	varchar(50)	контрагент
Amount	integer	количество товара продано
Price	decimal	сумма продажи
WarehouseID	integer	код места хранения
ProductID	integer	код товара

Таблица Deliveries (Поставки). В данной таблице будет храниться информация о поставках товаров в аптеку.

поле	тип поля	описание
WaybillID	integer	код накладной поставки
DeliveryDate	date	дата поставки
Counderparty	varchar(50)	контрагент
Amount	integer	количество товара в поставке
Price	decimal	цена поставки
WarehouseID	integer	код места хранения
ProductID	integer	код товара

Таблица Countries (Страны). В данной таблице будет храниться информация о странах, в которых функционируют контрагенты аптеки.

поле	тип поля	описание
CountryID	integer	уникальный идентификатор страны
CountryName	varchar(50)	название страны
Rating	integer	рейтинг страны

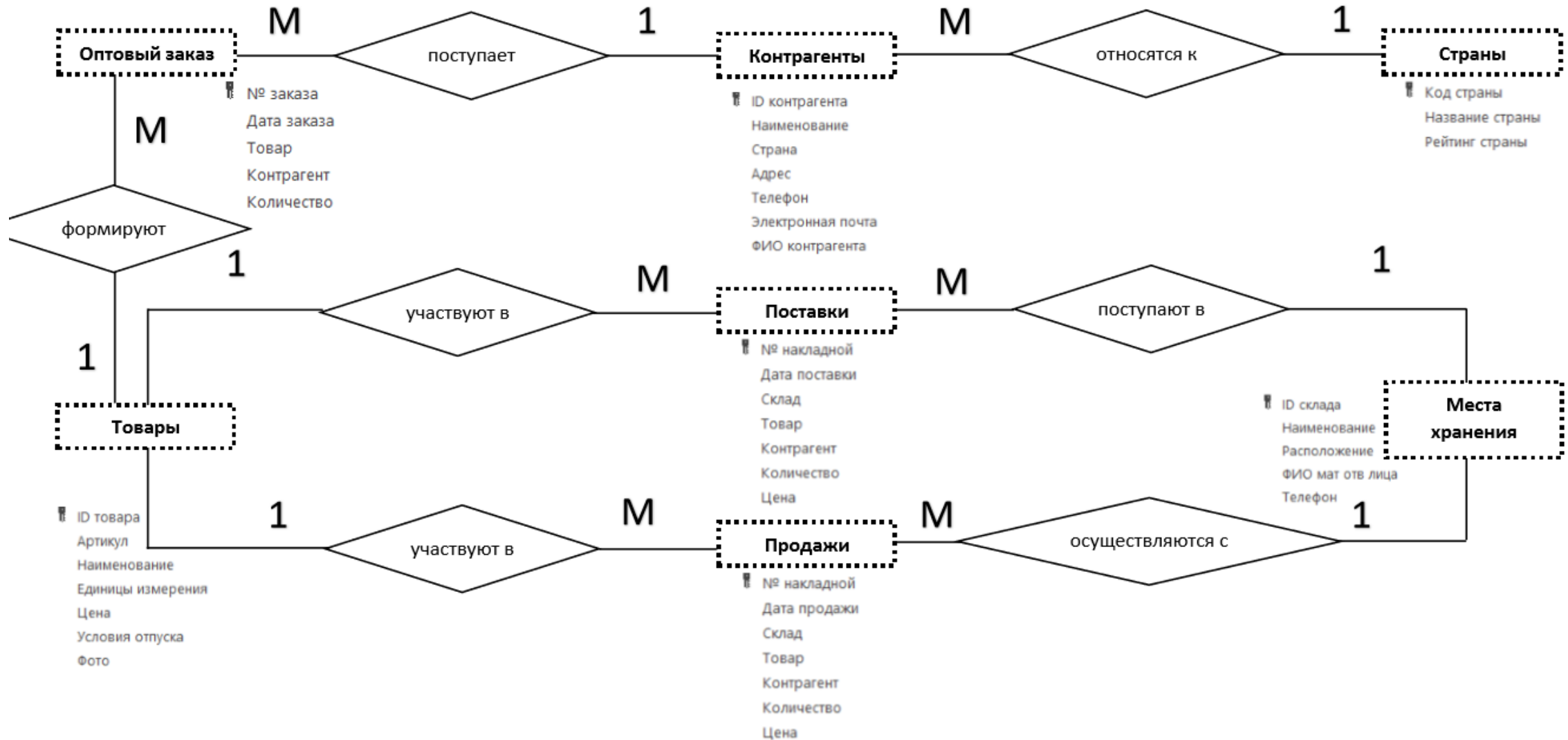
Таблица Counterparties (Контрагенты). В данной таблице будет храниться информация о контрагентах аптеки.

поле	тип поля	описание
CounterpartyID	integer	уникальный идентификатор контрагента
CounterpartyName	varchar(50)	наименование контрагента
Adress	varchar(50)	адрес контрагента
Phone	char(15)	телефон контрагента
Email	varchar(30)	электронная почта контрагента
FullName	varchar(50)	ФИО контрагента
CountryID	integer	страна контрагента

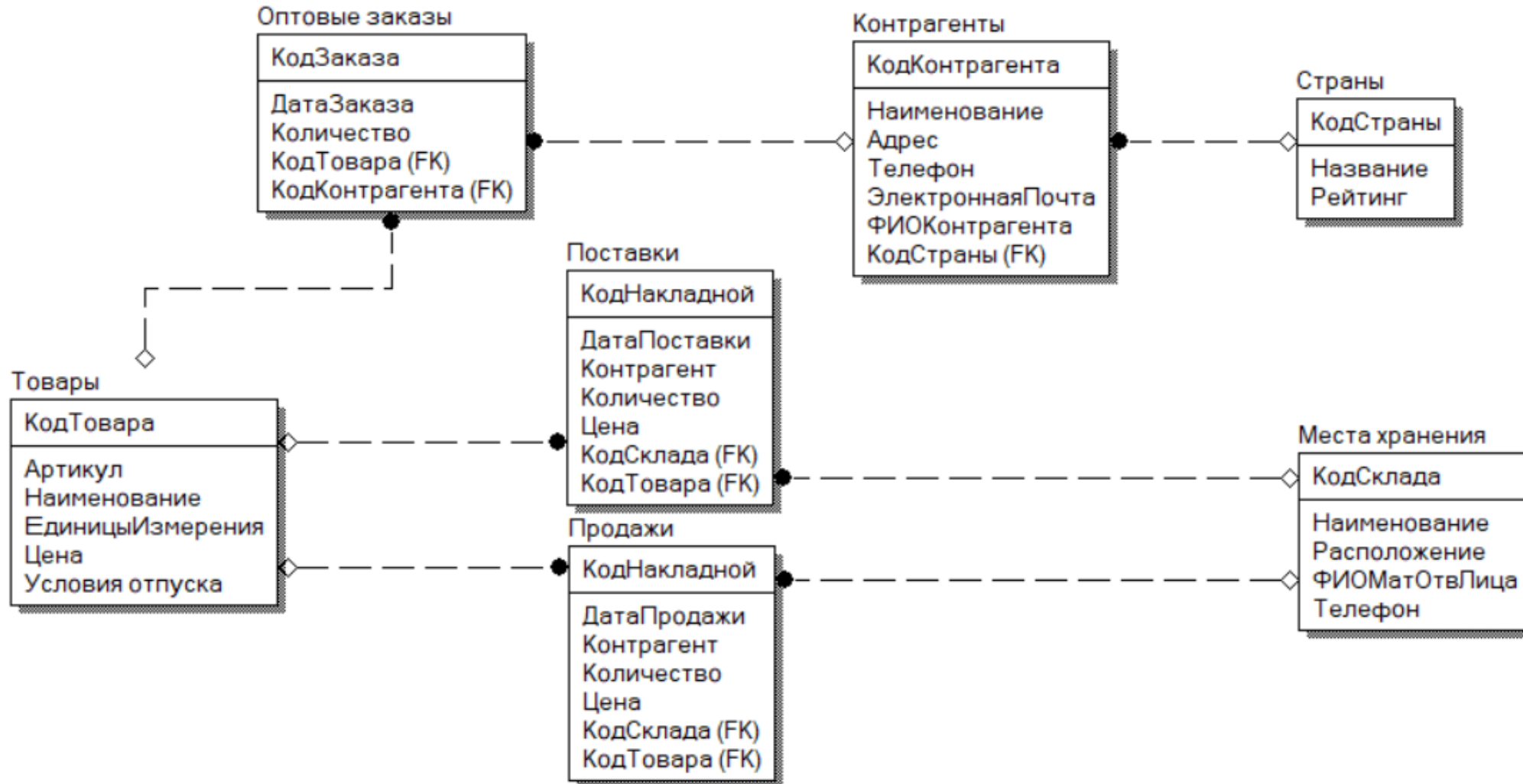
Таблица WholesaleOrders (Оптовые заказы). В данной таблице будет храниться информация о оптовых заказах аптеки у её контрагентов.

поле	тип поля	описание
OrderID	integer	уникальный идентификатор оптового заказа
OrderDate	date	дата оптового заказа
Amount	integer	количество товара в заказе
ProductID	integer	код товара
CounterpartyID	integer	код контрагента

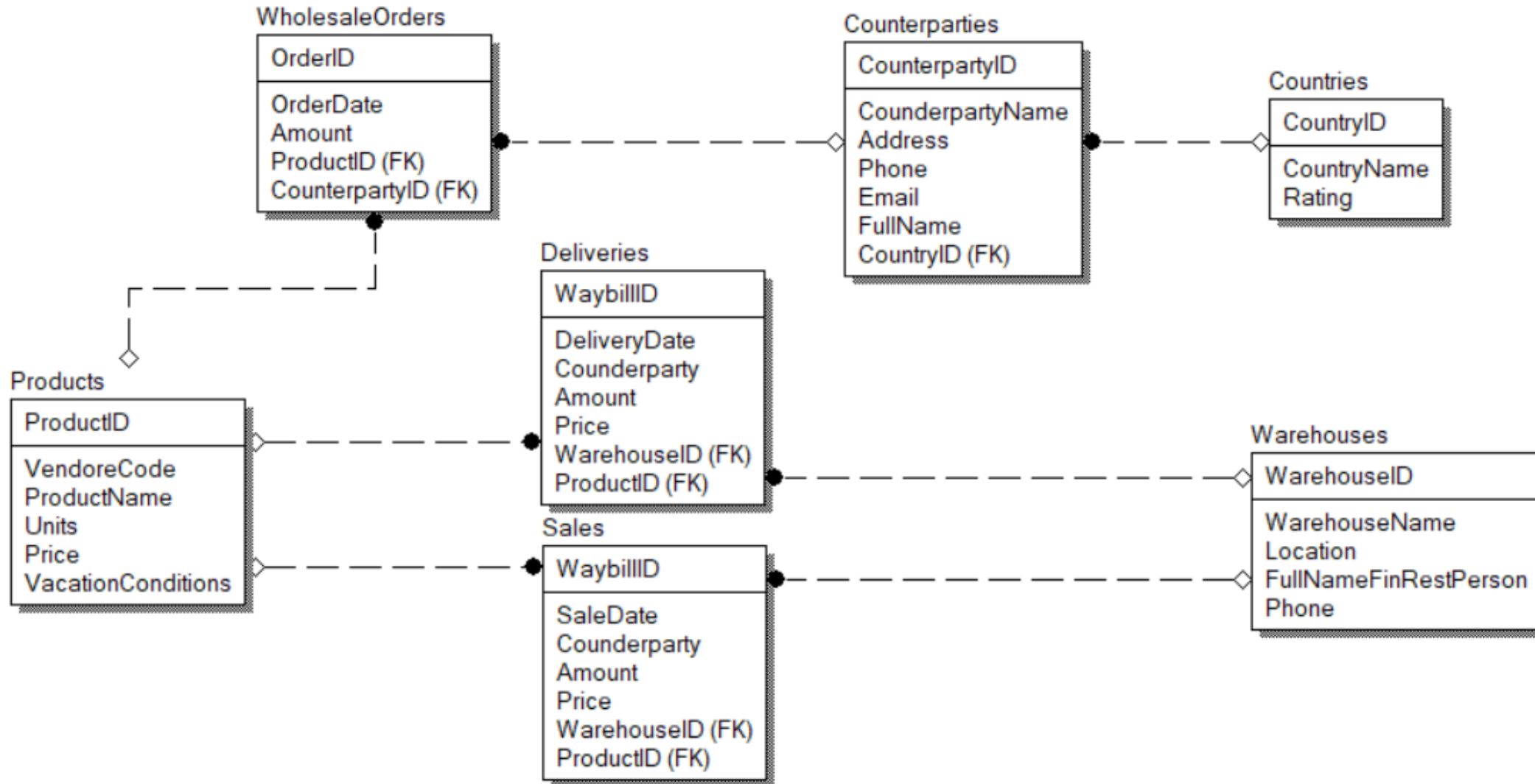
- Была разработана концептуальная (инфологическая) схема данных в нотации Питера Чена



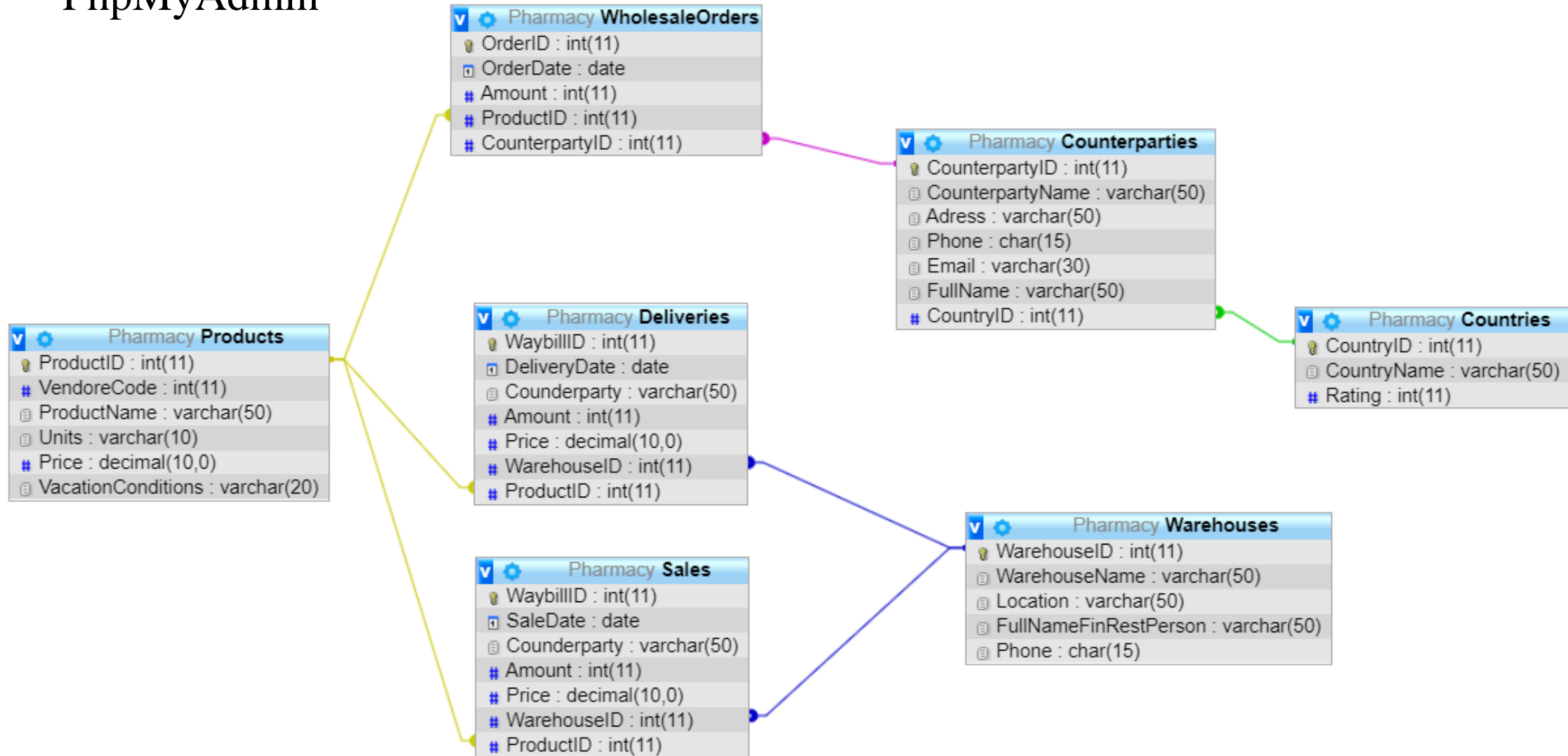
- В ErWin Data Modeler была нарисована **логическая** схема БД.



- Затем она была преобразована в **физическую** схему БД.



- Была создана **схема данных (диаграмма)** со включением всех созданных таблиц в PhpMyAdmin



- Были разработаны запросы для создания всех таблиц

```
CREATE TABLE IF NOT EXISTS `Products`(  
    `ProductID` int NOT NULL AUTO_INCREMENT,  
    `VendoreCode` int NOT NULL,  
    `ProductName` varchar(50) NOT NULL,  
    `Units` varchar(10) NOT NULL,  
    `Price` decimal(10,2) NOT NULL,  
    `VacationConditions` varchar(20) NOT NULL,  
    PRIMARY KEY (`ProductID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Warehouses`(  
    `WarehouseID` int NOT NULL AUTO_INCREMENT,  
    `WarehouseName` varchar(50) NOT NULL,  
    `Location` varchar(50) NOT NULL,  
    `FullNameFinRestPerson` varchar(50) NOT NULL,  
    `Phone` char(15) NOT NULL,  
    PRIMARY KEY (`WarehouseID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Countries`(  
    `CountryID` int NOT NULL AUTO_INCREMENT,  
    `CountryName` varchar(50) NOT NULL,  
    `Rating` int NOT NULL,  
    PRIMARY KEY (`CountryID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Sales`(  
    `WaybillID` int NOT NULL AUTO_INCREMENT,  
    `SaleDate` date NOT NULL,  
    `Counderparty` varchar (50) NOT NULL,  
    `Amount` int NOT NULL,  
    `Price` decimal(10,2) NOT NULL,  
    `WarehouseID` int NOT NULL,  
    `ProductID` int NOT NULL,  
    PRIMARY KEY (`WaybillID`),  
    KEY `Warehouses` (`WarehouseID`),  
    KEY `Products` (`ProductID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Counterparties`(  
    `CounterpartyID` int NOT NULL AUTO_INCREMENT,  
    `CounterpartyName` varchar(50) NOT NULL,  
    `Adress` varchar(50) NOT NULL,  
    `Phone` char(15) NOT NULL,  
    `Email` varchar(30) NOT NULL,  
    `FullName` varchar(50) NOT NULL,  
    `CountryID` int NOT NULL,  
    PRIMARY KEY (`CounterpartyID`),  
    KEY `Countries` (`CountryID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `Deliveries`(  
    `WaybillID` int NOT NULL AUTO_INCREMENT,  
    `DeliveryDate` date NOT NULL,  
    `Counderparty` varchar (50) NOT NULL,  
    `Amount` int NOT NULL,  
    `Price` decimal(10,2) NOT NULL,  
    `WarehouseID` int NOT NULL,  
    `ProductID` int NOT NULL,  
    PRIMARY KEY (`WaybillID`),  
    KEY `Warehouses` (`WarehouseID`),  
    KEY `Products` (`ProductID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE IF NOT EXISTS `WholesaleOrders`(  
    `OrderID` int NOT NULL AUTO_INCREMENT,  
    `OrderDate` date NOT NULL,  
    `Amount` int NOT NULL,  
    `ProductID` int NOT NULL,  
    `CounterpartyID` int NOT NULL,  
    PRIMARY KEY (`OrderID`),  
    KEY `Products` (`ProductID`),  
    KEY `Counterparties` (`CounterpartyID`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Были разработаны ограничения внешнего ключа

```
ALTER TABLE `Counterparties`  
    ADD CONSTRAINT `Counterparties_ibfk_1` FOREIGN KEY (`CountryID`)  
REFERENCES `Countries` (`CountryID`) ON DELETE CASCADE ON UPDATE CASCADE;
```

```
ALTER TABLE `WholesaleOrders`  
    ADD CONSTRAINT `WholesaleOrders_ibfk_1` FOREIGN KEY (`ProductID`)  
REFERENCES `Products` (`ProductID`) ON DELETE CASCADE ON UPDATE CASCADE,  
    ADD CONSTRAINT `WholesaleOrders_ibfk_2` FOREIGN KEY (`CounterpartyID`)  
REFERENCES `Counterparties` (`CounterpartyID`) ON DELETE CASCADE ON UPDATE  
CASCADE;
```

```
ALTER TABLE `Deliveries`  
    ADD CONSTRAINT `Deliveries_ibfk_1` FOREIGN KEY (`WarehouseID`)  
REFERENCES `Warehouses` (`WarehouseID`) ON DELETE CASCADE ON UPDATE  
CASCADE,  
    ADD CONSTRAINT `Deliveries_ibfk_2` FOREIGN KEY (`ProductID`)  
REFERENCES `Products` (`ProductID`) ON DELETE CASCADE ON UPDATE CASCADE;
```

```
ALTER TABLE `Sales`  
    ADD CONSTRAINT `Sales_ibfk_1` FOREIGN KEY (`WarehouseID`) REFERENCES  
`Warehouses` (`WarehouseID`) ON DELETE CASCADE ON UPDATE CASCADE,  
    ADD CONSTRAINT `Sales_ibfk_2` FOREIGN KEY (`ProductID`) REFERENCES  
`Products` (`ProductID`) ON DELETE CASCADE ON UPDATE CASCADE;
```

- Были разработаны запросы для заполнения данными таблиц.

```
INSERT INTO Products (`ProductID`, `VendoreCode`, `ProductName`, `Units`, `Price`,
`VacationConditions`) VALUES
('1', '123', 'Парацетамол', 'Упаковка', '5.96', 'Без рецепта'),
('2', '124', 'Анальгин', 'Упаковка', '1.24', 'Без рецепта'),
('3', '125', 'Зинерит', 'Упаковка', '32.64', 'По рецепту'),
('4', '126', 'Бепантен', 'Тюбик', '15.42', 'Без рецепта'),
('5', '127', 'Лизак', 'Упаковка', '4.81', 'Без рецепта'),
('6', '128', 'Антигриппин', 'Упаковка', '12.35', 'Без рецепта');
```

ProductID	VendoreCode	ProductName	Units	Price	VacationConditions
1	123	Парацетамол	Упаковка	5.96	Без рецепта
2	124	Анальгин	Упаковка	1.24	Без рецепта
3	125	Зинерит	Упаковка	32.64	По рецепту
4	126	Бепантен	Тюбик	15.42	Без рецепта
5	127	Лизак	Упаковка	4.81	Без рецепта
6	128	Антигриппин	Упаковка	12.35	Без рецепта

```
INSERT INTO WholesaleOrders (`OrderID`, `OrderDate`, `Amount`, `ProductID`, `CounterpartyID`) VALUES
('1', '2020-11-01', '20', '1', '1'),
('2', '2020-11-17', '50', '2', '3'),
('3', '2020-11-03', '45', '5', '6'),
('4', '2020-11-04', '15', '4', '2'),
('5', '2020-11-05', '70', '6', '5'),
('6', '2020-11-06', '65', '4', '3'),
('7', '2020-11-07', '80', '3', '4'),
('8', '2020-11-08', '20', '1', '2'),
('9', '2020-11-08', '25', '3', '6'),
('10', '2020-11-09', '30', '4', '5');
```

OrderID	OrderDate	Amount	ProductID	CounterpartyID
1	2020-11-01	20	1	1
2	2020-11-17	50	2	3
3	2020-11-03	45	5	6
4	2020-11-04	15	4	2
5	2020-11-05	70	6	5
6	2020-11-06	65	4	3
7	2020-11-07	80	3	4
8	2020-11-08	20	1	2
9	2020-11-08	25	3	6
10	2020-11-09	30	4	5

- Были разработаны запросы для заполнения данными таблиц.

INSERT INTO `Sales`

(`WaybillID`, `SaleDate`, `COUNTERPARTY`, `Amount`, `Price`, `WarehouseID`, `ProductID`)
VALUES

('1', '2020-11-15', '1', '17', '5.96', '1', '1'),
('2', '2020-11-15', '2', '12', '1.24', '2', '2'),
('3', '2020-11-16', '3', '20', '32.64', '3', '3'),
('4', '2020-11-16', '4', '10', '15.42', '4', '4'),
('5', '2020-11-16', '5', '8', '4.81', '5', '5'),
('6', '2020-11-17', '6', '16', '12.35', '6', '6'),
('7', '2020-11-17', '3', '25', '1.24', '1', '2'),
('8', '2020-11-18', '4', '5', '32.64', '2', '3'),
('9', '2020-11-19', '5', '21', '15.42', '3', '4'),
('10', '2020-11-20', '6', '30', '4.81', '4', '5');

WaybillID	SaleDate	COUNTERPARTY	Amount	Price	WarehouseID	ProductID
1	2020-11-15	1	17	5.96	1	1
2	2020-11-15	2	12	1.24	2	2
3	2020-11-16	3	20	32.64	3	3
4	2020-11-16	4	10	15.42	4	4
5	2020-11-16	5	8	4.81	5	5
6	2020-11-17	6	16	12.35	6	6
7	2020-11-17	3	25	1.24	1	2
8	2020-11-18	4	5	32.64	2	3
9	2020-11-19	5	21	15.42	3	4
10	2020-11-20	6	30	4.81	4	5

INSERT INTO `Deliveries`

(`WaybillID`, `DeliveryDate`, `COUNTERPARTY`, `Amount`, `Price`, `WarehouseID`, `ProductID`)
VALUES

('1', '2020-11-10', '1', '50', '1.24', '1', '2'),
('2', '2020-11-10', '3', '45', '15.42', '2', '4'),
('3', '2020-11-11', '2', '35', '5.96', '3', '1'),
('4', '2020-11-11', '4', '25', '32.64', '4', '3'),
('5', '2020-11-12', '5', '20', '12.35', '5', '6'),
('6', '2020-11-13', '6', '65', '4.81', '6', '5'),
('7', '2020-11-13', '1', '64', '32.64', '2', '3'),
('8', '2020-11-13', '2', '40', '5.96', '4', '1'),
('9', '2020-11-14', '3', '45', '15.42', '5', '4'),
('10', '2020-11-14', '2', '30', '1.24', '1', '2');

WaybillID	DeliveryDate	COUNTERPARTY	Amount	Price	WarehouseID	ProductID
1	2020-11-10	1	50	1.24	1	2
2	2020-11-10	3	45	15.42	2	4
3	2020-11-11	2	35	5.96	3	1
4	2020-11-11	4	25	32.64	4	3
5	2020-11-12	5	20	12.35	5	6
6	2020-11-13	6	65	4.81	6	5
7	2020-11-13	1	64	32.64	2	3
8	2020-11-13	2	40	5.96	4	1
9	2020-11-14	3	45	15.42	5	4
10	2020-11-14	2	30	1.24	1	2

- Были разработаны запросы для заполнения данными таблиц.

```
INSERT INTO `Counterparties` (`CounterpartyID`, `CounterpartyName`, `Adress`, `Phone`,
`Email`, `FullName`, `CountryID`) VALUES
('1', 'ООО "ФармВитекс"', 'г. Минск, ул. Тимирязева, 54', '+375291104294',
'FarmViteks@mail.ru', 'Иванов Петр Владимирович', '1'),
('2', 'ООО "Белфармация"', 'г. Брест, ул. Солнечная, 65', '+375291234567',
'belfarmcia@mail.ru', 'Борисова Ангелина Дмитриевна', '1'),
('3', 'ООО "Фармекс"', 'г. Дрезден, ул. Несецкая, 45', '+789265894657',
'farmeks@mail.ru', 'Федоров Владимир Вечеславович', '5'),
('4', 'ООО "Медикфармация"', 'г. Москва, ул. Цветочная, 9', '+79063514862',
'medicfarmacia@mail.ru', 'Ширяев Владимир Анатольевич', '2'),
('5', 'ООО "ФармМед"', 'г. Варшава, ул. Польская, 15', '+5695126515',
'farmmed@mail.ru', 'Титова Варвара Александровна', '4'),
('6', 'ООО "ЧехМедПрепарат"', 'г. Прага, ул. Чешская, 7', '+75912364856',
'zechmedpreparat@mail.ru', 'Августовская Ирина Владимировна', '4');
```

```
INSERT INTO `Countries` (`CountryID`, `CountryName`, `Rating`) VALUES
('1', 'Беларусь', '8'),
('2', 'Россия', '7'),
('3', 'Польша', '9'),
('4', 'Чехия', '9'),
('5', 'Германия', '10');
```

CountryID	CountryName	Rating
1	Беларусь	8
2	Россия	7
3	Польша	9
4	Чехия	9
5	Германия	10

CounterpartyID	CounterpartyName	Adress	Phone	Email	FullName	CountryID
1	ООО "ФармВитекс"	г. Минск, ул. Тимирязева, 54	+375291104294	FarmViteks@mail.ru	Иванов Петр Владимирович	1
2	ООО "Белфармация"	г. Брест, ул. Солнечная, 65	+375291234567	belfarmcia@mail.ru	Борисова Ангелина Дмитриевна	1
3	ООО "Фармекс"	г. Дрезден, ул. Несецкая, 45	+789265894657	farmeks@mail.ru	Федоров Владимир Вечеславович	5
4	ООО "Медикфармация"	г. Москва, ул. Цветочная, 9	+79063514862	medicfarmacia@mail.ru	Ширяев Владимир Анатольевич	2
5	ООО "ФармМед"	г. Варшава, ул. Польская, 15	+5695126515	farmmed@mail.ru	Титова Варвара Александровна	4
6	ООО "ЧехМедПрепарат"	г. Прага, ул. Чешская, 7	+75912364856	zechmedpreparat@mail.ru	Августовская Ирина Владимировна	4

- Были разработаны запросы для заполнения данными таблиц.

```
INSERT INTO `Warehouses`  
(`WarehouseID`,`WarehouseName`,`Location`,`FullNameFinRestPerson`,`Phone`)  
VALUES  
('1','Склад 123','г. Минск, ул. Луговая, 9','Петров Иван Владимирович',  
'+375293196359'),  
('2','Склад 234','г. Брест, ул. Полевая, 76','Цветкова Светлана Георгиевна',  
'+375448963571'),  
('3','Склад 345','г. Витебск, ул. Янки Купало, 34','Васечкин Григорий Васильевич',  
'+375336489125'),  
('4','Склад 456','г. Гомель, ул. Красной книги, 68','Тихановская Валентина  
Федоровна','+375291163595'),  
('5','Склад 567','г. Гродно, ул. Лунная, 156','Тихановская Валентина Федоровна',  
'+375294569182'),  
('6','Склад 678','г. Могилев, ул. Хоружей, 36','Журавская Анастасия Игоревна',  
'+375448961537');
```

WarehouseID	WarehouseName	Location	FullNameFinRestPerson	Phone
1	Склад 123	г. Минск, ул. Луговая, 9	Петров Иван Владимирович	<u>+375293196359</u>
2	Склад 234	г. Брест, ул. Полевая, 76	Цветкова Светлана Георгиевна	<u>+375448963571</u>
3	Склад 345	г. Витебск, ул. Янки Купало, 34	Васечкин Григорий Васильевич	<u>+375336489125</u>
4	Склад 456	г. Гомель, ул. Красной книги, 68	Тихановская Валентина Федоровна	<u>+375291163595</u>
5	Склад 567	г. Гродно, ул. Лунная, 156	Тихановская Валентина Федоровна	<u>+375294569182</u>
6	Склад 678	г. Могилев, ул. Хоружей, 36	Журавская Анастасия Игоревна	<u>+375448961537</u>

Запросы к разработанной базе данных

Исходная таблица:

ProductID	VendoreCode	ProductName	Units	Price	VacationConditions
1	123	Парацетамол	Упаковка	5.96	Без рецепта
2	124	Анальгин	Упаковка	1.24	Без рецепта
3	125	Зинерит	Упаковка	32.64	По рецепту
4	126	Бепантен	Тюбик	15.42	Без рецепта
5	127	Лизак	Упаковка	4.81	Без рецепта
6	128	Антигриппин	Упаковка	12.35	Без рецепта

1. Запрос на сортировку товаров по условию продажи "без рецепта" и по уменьшению цены.

Запрос:

```
SELECT * FROM `Products` WHERE  
`VacationConditions` = 'Без рецепта' ORDER BY  
`Price` DESC
```

Результат выполнения:

ProductID	VendoreCode	ProductName	Units	Price ▼ 1	VacationConditions
4	126	Бепантен	Тюбик	15.42	Без рецепта
6	128	Антигриппин	Упаковка	12.35	Без рецепта
1	123	Парацетамол	Упаковка	5.96	Без рецепта
5	127	Лизак	Упаковка	4.81	Без рецепта
2	124	Анальгин	Упаковка	1.24	Без рецепта

Запросы к разработанной базе данных

2. Запрос, позволяющий вывести все поля таблицы Продажи кроме поля Склад, и вывести строки, относящиеся к товару "Лизак", количество которого продали больше 20 штук. Затем отсортировать данные по возрастанию дат продаж и по убыванию количества проданного товара. Это может быть необходимо для составления статистики по заболеваемости (в какие даты покупают больше товара).

Исходная таблица:

WaybillID	SaleDate	Counderparty	Amount	Price	WarehouseID	ProductID
1	2020-11-15	1	17	5.96	1	1
2	2020-11-15	2	12	1.24	2	2
3	2020-11-16	3	20	32.64	3	3
4	2020-11-16	4	10	15.42	4	4
5	2020-11-16	5	8	4.81	5	5
6	2020-11-17	6	16	12.35	6	6
7	2020-11-17	3	25	1.24	1	2
8	2020-11-18	4	5	32.64	2	3
9	2020-11-19	5	21	15.42	3	4
10	2020-11-20	6	30	4.81	4	5

Запрос:

```
SELECT `WaybillID`, `SaleDate`, `Amount`, `Price`,  
`ProductID` FROM `Sales` WHERE `ProductID` =  
(SELECT `ProductID` FROM `Products` WHERE  
`ProductName` = 'Лизак') AND `Amount` > 20  
ORDER BY `SaleDate` ASC, `Amount` DESC
```

Результат выполнения:

WaybillID	SaleDate	Amount	Price	ProductID
10	2020-11-20	30	4.81	5

Запросы к разработанной базе данных

3. Запрос, который выводит все даты поставок товара в порядке убывания (на тот случай, если необходимо проверить, в какие дни обычно осуществляются поставки товара).

Исходная таблица:

WaybillID	SaleDate	Counderparty	Amount	Price	WarehouseID	ProductID
1	2020-11-15	1	17	5.96	1	1
2	2020-11-15	2	12	1.24	2	2
3	2020-11-16	3	20	32.64	3	3
4	2020-11-16	4	10	15.42	4	4
5	2020-11-16	5	8	4.81	5	5
6	2020-11-17	6	16	12.35	6	6
7	2020-11-17	3	25	1.24	1	2
8	2020-11-18	4	5	32.64	2	3
9	2020-11-19	5	21	15.42	3	4
10	2020-11-20	6	30	4.81	4	5

Запрос:

```
SELECT `DeliveryDate` AS `Даты  
поставок товара` FROM `Deliveries`  
GROUP BY `DeliveryDate` ORDER BY  
`Даты поставок товара` DESC
```

Результат выполнения:

Даты поставок товара ▾ 1
2020-11-14
2020-11-13
2020-11-12
2020-11-11
2020-11-10

Запросы к разработанной базе данных

4. Запрос, группирующий данные из таблиц Товары и Продажи по названию товара, он выводит его суммарное количество, общую цену и итоговую стоимость проданного товара (запрос показывает с какого товара мы больше заработали), сортировка производится по столбцу общая стоимость проданного товара по убыванию.

Исходные таблицы:

ProductID	VendoreCode	ProductName	Units	Price	VacationConditions
1	123	Парацетамол	Упаковка	5.96	Без рецепта
2	124	Анальгин	Упаковка	1.24	Без рецепта
3	125	Зинерит	Упаковка	32.64	По рецепту
4	126	Бепантен	Тюбик	15.42	Без рецепта
5	127	Лизак	Упаковка	4.81	Без рецепта
6	128	Антигриппин	Упаковка	12.35	Без рецепта

WaybillID	SaleDate	Counderparty	Amount	Price	WarehouseID	ProductID
1	2020-11-15	1	17	5.96	1	1
2	2020-11-15	2	12	1.24	2	2
3	2020-11-16	3	20	32.64	3	3
4	2020-11-16	4	10	15.42	4	4
5	2020-11-16	5	8	4.81	5	5
6	2020-11-17	6	16	12.35	6	6
7	2020-11-17	3	25	1.24	1	2
8	2020-11-18	4	5	32.64	2	3
9	2020-11-19	5	21	15.42	3	4
10	2020-11-20	6	30	4.81	4	5

Запрос:

```
SELECT P.`ProductName` AS `Название проданного товара`,
SUM(S.`Amount`) AS `Суммарное количество проданного товара`,
SUM(S.`Price`) AS `Общая цена проданного товара`,
SUM(S.`Price` * S.`Amount`) AS `Общая стоимость проданного
товара` FROM `Sales` S, `Products` P WHERE S.`ProductID` =
P.`ProductID` GROUP BY P.`ProductName` ORDER BY `Общая
стоимость проданного товара` DESC
```

Результат выполнения:

Название проданного товара	Суммарное количество проданного товара	Общая цена проданного товара	Общая стоимость проданного товара ▼ 1
Зинерит	25	65.28	816.00
Бепантен	31	30.84	478.02
Антигриппин	16	12.35	197.60
Лизак	38	9.62	182.78
Парацетамол	17	5.96	101.32
Анальгин	37	2.48	45.88

Запросы к разработанной базе данных

5. Запрос заменяет коды Товара и Контрагента в таблице Оптовые заказы на названия и имена, это облегчит понимание и чтение пользователю. В таблицу добавляется столбец с Ценой и расчетный столбец Стоимость, выполняется сортировка по убыванию по столбцам Стоимость и Дата Заказа.

Исходная таблица:

OrderID	OrderDate	Amount	ProductID	CounterpartyID
1	2020-11-01	20	1	1
2	2020-11-17	50	2	3
3	2020-11-03	45	5	6
4	2020-11-04	15	4	2
5	2020-11-05	70	6	5
6	2020-11-06	65	4	3
7	2020-11-07	80	3	4
8	2020-11-08	20	1	2
9	2020-11-08	25	3	6
10	2020-11-09	30	4	5

Запрос:

```
SELECT W.`OrderID`, W.`OrderDate`, W.`Amount`, P.`Price`,  
W.`Amount` * P.`Price` AS `Cost`, P.`ProductName`,  
C.`CounterpartyName` FROM `WholesaleOrders` W, `Products` P,  
`Counterparties` C WHERE W.`ProductID` = P.`ProductID` AND  
W.`CounterpartyID` = C.`CounterpartyID` ORDER BY `Cost` DESC,  
`OrderDate` DESC
```

Результат выполнения:

OrderID	OrderDate ▾ 2	Amount	Price	Cost ▾ 1	ProductName	CounterpartyName
7	2020-11-07	80	32.64	2611.20	Зинерит	ООО "Медикфармация"
6	2020-11-06	65	15.42	1002.30	Бепантен	ООО "Фармекс"
5	2020-11-05	70	12.35	864.50	Антигриппин	ООО "ФармМед"
9	2020-11-08	25	32.64	816.00	Зинерит	ООО "ЧехМедПрепарат"
10	2020-11-09	30	15.42	462.60	Бепантен	ООО "ФармМед"
4	2020-11-04	15	15.42	231.30	Бепантен	ООО "Белфармация"
3	2020-11-03	45	4.81	216.45	Лизак	ООО "ЧехМедПрепарат"
8	2020-11-08	20	5.96	119.20	Парацетамол	ООО "Белфармация"
1	2020-11-01	20	5.96	119.20	Парацетамол	ООО "ФармВитекс"
2	2020-11-17	50	1.24	62.00	Анальгин	ООО "Фармекс"

Запросы к разработанной базе данных

6. Данный запрос позволяет нам найти номера оптовых заказов из конкретной страны, в данном случае из Германии.

Исходные таблицы:

OrderID	OrderDate	Amount	ProductID	CounterpartyID
1	2020-11-01	20	1	1
2	2020-11-17	50	2	3
3	2020-11-03	45	5	6
4	2020-11-04	15	4	2
5	2020-11-05	70	6	5
6	2020-11-06	65	4	3
7	2020-11-07	80	3	4
8	2020-11-08	20	1	2
9	2020-11-08	25	3	6
10	2020-11-09	30	4	5

CounterpartyID	CounterpartyName	Adress	Phone	Email	FullName	CountryID
1	ООО "ФармВитекс"	г. Минск, ул. Тимирязева, 54	+375291104294	FarmViteks@mail.ru	Иванов Петр Владимирович	1
2	ООО "Белфармация"	г. Брест, ул. Солнечная, 65	+375291234567	belfarmcia@mail.ru	Борисова Ангелина Дмитриевна	1
3	ООО "Фармекс"	г. Дрезден, ул. Несецкая, 45	+789265894657	farmeks@mail.ru	Федоров Владимир Вечеславович	5
4	ООО "Медикфармация"	г. Москва, ул. Цветочная, 9	+79063514862	medicfarmacia@mail.ru	Ширяев Владимир Анатольевич	2
5	ООО "ФармМед"	г. Варшава, ул. Польская, 15	+5695126515	farmmed@mail.ru	Титова Варвара Александровна	4
6	ООО "ЧехМедПрепарат"	г. Прага, ул. Чешская, 7	+75912364856	zechmedpreparat@mail.ru	Августовская Ирина Владимировна	4

CountryID	CountryName	Rating
1	Беларусь	8
2	Россия	7
3	Польша	9
4	Чехия	9
5	Германия	10

Результат выполнения:

Код заказов из Германии
2
6

Запрос:

```
SELECT `OrderID` AS `Код заказов из Германии` FROM `WholesaleOrders` WHERE `CounterpartyID` =  
(SELECT C1.`CounterpartyID` FROM `Counterparties` C1 WHERE `CountryID` = (SELECT  
C.`CountryID` FROM `Countries` C WHERE C.`CountryName` = 'Германия'))
```

Запросы к разработанной базе данных

7. Запрос, который объединяет две таблицы, в нашем случае будем объединять таблицы Контрагенты и Страны по равенству CountryID, в результирующую таблицу выводим столбцы Код контрагента, Наименование Контрагента, Адрес, Страна, Рейтинг страны. Запрос создается с целью посмотреть, какой контрагент относится к какой стране и какой рейтинг у данной страны.

Исходные таблицы:

CounterpartyID	CounterpartyName	Adress	Phone	Email	FullName	CountryID
1	ООО "ФармВитекс"	г. Минск, ул. Тимирязева, 54	+375291104294	FarmViteks@mail.ru	Иванов Петр Владимирович	1
2	ООО "Белфармация"	г. Брест, ул. Солнечная, 65	+375291234567	belfarmcia@mail.ru	Борисова Ангелина Дмитриевна	1
3	ООО "Фармекс"	г. Дрезден, ул. Несецкая, 45	+789265894657	farmeks@mail.ru	Федоров Владимир Вечеславович	5
4	ООО "Медикфармация"	г. Москва, ул. Цветочная, 9	+79063514862	medicfarmacia@mail.ru	Ширяев Владимир Анатольевич	2
5	ООО "ФармМед"	г. Варшава, ул. Польская, 15	+5695126515	farmmed@mail.ru	Титова Варвара Александровна	4
6	ООО "ЧехМедПрепарат"	г. Прага, ул. Чешская, 7	+75912364856	zechmedpreparat@mail.ru	Августовская Ирина Владимировна	4

CountryID	CountryName	Rating
1	Беларусь	8
2	Россия	7
3	Польша	9
4	Чехия	9
5	Германия	10

Запрос:

```
SELECT C2.`CounterpartyID`, C2.`CounterpartyName`, C2.`Adress`, C1.`CountryName`, C1.`Rating` FROM `Countries` C1 INNER JOIN `Counterparties` C2 ON C1.`CountryID` = C2.`CountryID`
```

Результат выполнения:

CounterpartyID	CounterpartyName	Adress	CountryName	Rating
1	ООО "ФармВитекс"	г. Минск, ул. Тимирязева, 54	Беларусь	8
2	ООО "Белфармация"	г. Брест, ул. Солнечная, 65	Беларусь	8
4	ООО "Медикфармация"	г. Москва, ул. Цветочная, 9	Россия	7
5	ООО "ФармМед"	г. Варшава, ул. Польская, 15	Чехия	9
6	ООО "ЧехМедПрепарат"	г. Прага, ул. Чешская, 7	Чехия	9
3	ООО "Фармекс"	г. Дрезден, ул. Несецкая, 45	Германия	10

Спасибо за внимание!