**1.Составить общее текстовое описание БД и решаемых ею задач;**

База данных создана для учета товара-оборота интернет магазина по продаже техники. В базе присутствуют таблицы:

**2. минимальное количество таблиц - 10;**

catalogs, products, PickUpPoint, contractor, discounts, earnings,movement,Receipt, Receipt\_movement, Selling, Selling\_movement, Return, Return\_movement, write-off, writeoff\_movement.

**3. скрипты создания структуры БД (с первичными ключами, индексами, внешними ключами);**

DROP DATABASE OnlineStore;

CREATE DATABASE OnlineStore;

Use OnlineStore;

DROP Table if EXISTS catalogs;

CREATE TABLE catalogs (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) CHARACTER SET utf8 DEFAULT NULL COMMENT 'Название раздела',

`level\_nesting` bigint(20) unsigned DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Разделы интернет-магазина';

DROP Table if EXISTS products;

CREATE TABLE products (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`artikul` varchar(12) CHARACTER SET utf8 DEFAULT NULL COMMENT 'Артикул',

`name` varchar(255) CHARACTER SET utf8 DEFAULT NULL COMMENT 'Название',

`description` text COLLATE utf8\_unicode\_ci COMMENT 'Описание',

`price` decimal(11,2) DEFAULT NULL COMMENT 'Цена',

`catalog\_id` bigint(20) unsigned DEFAULT NULL,

CONSTRAINT `fk\_catalog\_id` FOREIGN KEY (`catalog\_id`) REFERENCES `catalogs` (`id`),

`quantity` int(10) unsigned DEFAULT '0' COMMENT 'Остаток',

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

KEY `index\_artikul` (`artikul`),

KEY `index\_name` (`name`),

KEY `index\_of\_catalog\_id` (`catalog\_id`)

) ENGINE=InnoDB AUTO\_INCREMENT=457 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Товарные позиции';

DROP Table if EXISTS discounts;

CREATE TABLE `discounts` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`discount` float unsigned DEFAULT NULL COMMENT 'Процент скидки',

`started\_at` datetime DEFAULT NULL,

`finished\_at` datetime DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Скидки';

DROP Table if EXISTS contractor;

CREATE TABLE `contractor` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(50) CHARACTER SET utf8 DEFAULT NULL COMMENT 'Наименование',

`adress` varchar(50) CHARACTER SET utf8 DEFAULT NULL COMMENT 'Почтовый Адрес',

`email` varchar(120) DEFAULT NULL,

`phone` bigint(20) DEFAULT NULL,

status enum('client','partner'),

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

UNIQUE KEY `email` (`email`),

KEY `users\_phone\_idx` (`phone`)

) ENGINE=InnoDB AUTO\_INCREMENT=101 DEFAULT CHARSET=latin1 COMMENT='Контрагенты';

-- ALTER TABLE contractor add status enum('client','partner');

DROP Table if EXISTS earnings;

CREATE TABLE `earnings` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`data` datetime DEFAULT NULL,

`contractor\_id` bigint(20) unsigned NOT NULL,

`amount` MEDIUMINT DEFAULT '0' COMMENT 'Сумма',

PRIMARY KEY (`id`),

-- UNIQUE KEY `id` (`id`)

CONSTRAINT `fk\_contrCash\_id` FOREIGN KEY (`contractor\_id`) REFERENCES `contractor` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Прибыль';

DROP Table if EXISTS PickUpPoint;

CREATE TABLE `PickUpPoint` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(50) Not NULL,

`adress` varchar(120) Not NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Пункт получения';

-- поступление товаров

DROP Table if EXISTS Receipt;

CREATE TABLE `Receipt` (

`id` bigint(10) unsigned NOT NULL AUTO\_INCREMENT,

`number` bigint(20) unsigned DEFAULT NULL,

`data` datetime DEFAULT CURRENT\_TIMESTAMP,

`contractor\_id` bigint(20) unsigned NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

KEY `index\_of\_contractor\_id` (`contractor\_id`),

CONSTRAINT `fk\_contractor\_id` FOREIGN KEY (`contractor\_id`) REFERENCES `contractor` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Поступление товаров';

DROP Table if EXISTS Receipt\_movement;

CREATE TABLE `Receipt\_movement` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

-- typeOfDocument enum('Admission','Selling','write-off','audit','Return'),

`document\_id` bigint(10) unsigned NOT NULL,

`product\_id` bigint(20) unsigned NOT NULL,

`quantity` int(10) unsigned DEFAULT '1' COMMENT 'Количество',

`price` int(10) unsigned DEFAULT '0' COMMENT 'Цена',

`amount` int(10) unsigned DEFAULT '0' COMMENT 'Сумма',

-- `updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

CONSTRAINT `fk\_product\_id` FOREIGN KEY (`product\_id`) REFERENCES `products` (`id`),

CONSTRAINT `fk\_Admission\_id` FOREIGN KEY (`document\_id`) REFERENCES `Receipt` (`id`)

-- CONSTRAINT `fk\_selling\_id` FOREIGN KEY (`document\_id`) REFERENCES `Selling` (`id`)

-- CONSTRAINT `fk\_Return\_id` FOREIGN KEY (`document\_id`) REFERENCES `Return` (`id`),

-- CONSTRAINT `fk\_write-off\_id` FOREIGN KEY (`document\_id`) REFERENCES `write-off` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Товары документа постпления';

DROP Table if EXISTS Selling;

CREATE TABLE `Selling` (

`id` bigint(10) unsigned NOT NULL AUTO\_INCREMENT,

`number` bigint(20) unsigned DEFAULT NULL,

`data` datetime DEFAULT CURRENT\_TIMESTAMP,

`contractor\_id` bigint(20) unsigned NOT NULL,

PickUpPoint\_id bigint(20) unsigned DEFAULT NULL,

discount\_id bigint(20) unsigned DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

KEY `index\_of\_contractor\_id` (`contractor\_id`),

CONSTRAINT `fk\_contr\_id` FOREIGN KEY (`contractor\_id`) REFERENCES `contractor` (`id`),

CONSTRAINT `fk\_PickUpPoint\_id` FOREIGN KEY (`PickUpPoint\_id`) REFERENCES `PickUpPoint` (`id`),

CONSTRAINT `fk\_discount\_id` FOREIGN KEY (`discount\_id`) REFERENCES discounts (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Реализация товаров';

DROP Table if EXISTS Selling\_movement;

CREATE TABLE `Selling\_movement` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

-- typeOfDocument enum('Admission','Selling','write-off','audit','Return'),

`document\_id` bigint(10) unsigned NOT NULL,

`product\_id` bigint(20) unsigned NOT NULL,

`quantity` int(10) unsigned DEFAULT '1' COMMENT 'Количество',

`price` int(10) unsigned DEFAULT '0' COMMENT 'Цена',

`amount` int(10) unsigned DEFAULT '0' COMMENT 'Сумма',

-- `updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

CONSTRAINT `fk\_product2\_id` FOREIGN KEY (`product\_id`) REFERENCES `products` (`id`),

-- CONSTRAINT `fk\_Admission\_id` FOREIGN KEY (`document\_id`) REFERENCES `Receipt` (`id`)

CONSTRAINT `fk\_selling\_id` FOREIGN KEY (`document\_id`) REFERENCES `Selling` (`id`)

-- CONSTRAINT `fk\_Return\_id` FOREIGN KEY (`document\_id`) REFERENCES `Return` (`id`),

-- CONSTRAINT `fk\_write-off\_id` FOREIGN KEY (`document\_id`) REFERENCES `write-off` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Товары документа продажи';

DROP Table if EXISTS `Return`;

CREATE TABLE `Return` (

`id` bigint(10) unsigned NOT NULL AUTO\_INCREMENT,

`number` bigint(20) unsigned DEFAULT NULL,

`data` datetime DEFAULT CURRENT\_TIMESTAMP,

`contractor\_id` bigint(20) unsigned NOT NULL,

PickUpPoint\_id bigint(20) unsigned DEFAULT NULL,

cause varchar(150) DEFAULT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

KEY `index\_of\_contractor\_id` (`contractor\_id`),

CONSTRAINT `fk\_contrReturn\_id` FOREIGN KEY (`contractor\_id`) REFERENCES `contractor` (`id`),

CONSTRAINT `fk\_PickUpPointReturn\_id` FOREIGN KEY (`PickUpPoint\_id`) REFERENCES `PickUpPoint` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Возврат от покупателя';

DROP Table if EXISTS Return\_movement;

CREATE TABLE `Return\_movement` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

-- typeOfDocument enum('Admission','Selling','write-off','audit','Return'),

`document\_id` bigint(10) unsigned NOT NULL,

`product\_id` bigint(20) unsigned NOT NULL,

`quantity` int(10) unsigned DEFAULT '1' COMMENT 'Количество',

`price` int(10) unsigned DEFAULT '0' COMMENT 'Цена',

`amount` int(10) unsigned DEFAULT '0' COMMENT 'Сумма',

-- `updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

CONSTRAINT `fk\_product3\_id` FOREIGN KEY (`product\_id`) REFERENCES `products` (`id`),

-- CONSTRAINT `fk\_Admission\_id` FOREIGN KEY (`document\_id`) REFERENCES `Receipt` (`id`)

-- CONSTRAINT `fk\_selling\_id` FOREIGN KEY (`document\_id`) REFERENCES `Selling` (`id`)

CONSTRAINT `fk\_Return\_id` FOREIGN KEY (`document\_id`) REFERENCES `Return` (`id`)

-- CONSTRAINT `fk\_write-off\_id` FOREIGN KEY (`document\_id`) REFERENCES `write-off` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Товары возврата';

DROP Table if EXISTS `write-off`;

CREATE TABLE `write-off` (

`id` bigint(10) unsigned NOT NULL AUTO\_INCREMENT,

`number` bigint(20) unsigned DEFAULT NULL,

`data` datetime DEFAULT CURRENT\_TIMESTAMP,

cause varchar(150) DEFAULT NULL,

-- `contractor\_id` bigint(20) unsigned NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Списание товаров';

DROP Table if EXISTS writeoff\_movement;

CREATE TABLE `writeoff\_movement` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

-- typeOfDocument enum('Admission','Selling','write-off','audit','Return'),

`document\_id` bigint(10) unsigned NOT NULL,

`product\_id` bigint(20) unsigned NOT NULL,

`quantity` int(10) unsigned DEFAULT '1' COMMENT 'Количество',

`price` int(10) unsigned DEFAULT '0' COMMENT 'Цена',

`amount` int(10) unsigned DEFAULT '0' COMMENT 'Сумма',

-- `updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

CONSTRAINT `fk\_product4\_id` FOREIGN KEY (`product\_id`) REFERENCES `products` (`id`),

-- CONSTRAINT `fk\_Admission\_id` FOREIGN KEY (`document\_id`) REFERENCES `Receipt` (`id`)

-- CONSTRAINT `fk\_selling\_id` FOREIGN KEY (`document\_id`) REFERENCES `Selling` (`id`)

-- CONSTRAINT `fk\_Return\_id` FOREIGN KEY (`document\_id`) REFERENCES `Return` (`id`),

CONSTRAINT `fk\_write-off\_id` FOREIGN KEY (`document\_id`) REFERENCES `write-off` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Товары списания';

DROP Table if EXISTS movement;

CREATE TABLE `movement` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

typeOfDocument enum('Admission','Selling','write-off','audit','Return'),

`document\_id` bigint(10) unsigned NOT NULL,

`product\_id` bigint(20) unsigned NOT NULL,

`quantity` int(10) unsigned DEFAULT '1' COMMENT 'Количество',

`price` int(10) unsigned DEFAULT '0' COMMENT 'Цена',

`amount` int(10) unsigned DEFAULT '0' COMMENT 'Сумма',

-- `updated\_at` datetime DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (`id`),

UNIQUE KEY `id` (`id`),

CONSTRAINT `fk\_productv\_id` FOREIGN KEY (`product\_id`) REFERENCES `products` (`id`)

-- CONSTRAINT `fk\_Admission\_id` FOREIGN KEY (`document\_id`) REFERENCES `Receipt` (`id`),

-- CONSTRAINT `fk\_selling\_id` FOREIGN KEY (`document\_id`) REFERENCES `Selling` (`id`)

-- CONSTRAINT `fk\_Return\_id` FOREIGN KEY (`document\_id`) REFERENCES `Return` (`id`),

-- CONSTRAINT `fk\_write-off\_id` FOREIGN KEY (`document\_id`) REFERENCES `write-off` (`id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci COMMENT='Движение товара';

DROP TRIGGER if EXISTS Control\_movent\_Receipt;

DELIMITER //

CREATE TRIGGER Control\_movent\_Receipt

AFTER DELETE

ON Receipt FOR EACH ROW

BEGIN

DELETE FROM Receipt\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_movent\_Selling//

CREATE TRIGGER Control\_movent\_Selling

AFTER DELETE

ON Selling FOR EACH ROW

BEGIN

DELETE FROM Selling\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_movent\_Return//

CREATE TRIGGER Control\_movent\_Return

AFTER DELETE

ON `Return` FOR EACH ROW

BEGIN

DELETE FROM Return\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_movent\_writeoff//

CREATE TRIGGER Control\_movent\_writeoff

AFTER DELETE

ON `write-off` FOR EACH ROW

BEGIN

DELETE FROM writeoff\_movement where document\_id = OLD.id;

END//

-- DROP TRIGGER if EXISTS Control\_Stock//

-- CREATE TRIGGER Control\_Stock

-- BEFORE INSERT ON Receipt\_movement FOR EACH ROW

-- BEGIN

-- call Change\_count\_products(NEW.product\_id, NEW.document\_id);

-- END//

DROP TRIGGER if EXISTS Control\_earnings\_movement//

CREATE TRIGGER Control\_earnings\_movement

AFTER INSERT ON movement FOR EACH ROW

BEGIN

if NEW.typeOfDocument = 'Selling' then

INSERT INTO earnings (`data`,contractor\_id,amount)

VALUES

(NOW(),(select contractor\_id from Selling where id = NEW.document\_id),NEW.amount);

end if;

END//

DROP PROCEDURE IF EXISTS Change\_count\_products;

DELIMITER //

CREATE PROCEDURE Change\_count\_products (in productID bigint,in quantityM int(10))

BEGIN

UPDATE products

set quantity = quantity + quantityM where id = productID;

-- SELECT quantityM;

END//

DROP TRIGGER if EXISTS Control\_earnings\_Return//

CREATE TRIGGER Control\_earnings\_Return

AFTER INSERT ON movement FOR EACH ROW

BEGIN

DECLARE var1 int;

if NEW.typeOfDocument = 'Return' then

select NEW.amount into var1;

INSERT INTO earnings (`data`,contractor\_id,amount)

VALUES

(NOW(),(select contractor\_id from `Return` where id = NEW.document\_id),(var1\*(-1)));

end if;

-- Change\_count\_products(New.product\_id,New.quantity);

END//

DROP PROCEDURE IF EXISTS product\_balances//

-- DELIMITER //

CREATE PROCEDURE product\_balances ()

BEGIN

select name, quantity from products where quantity > 0;

END//

DROP PROCEDURE IF EXISTS product\_Receipt//

-- DELIMITER //

CREATE PROCEDURE product\_Receipt ()

BEGIN

update products

join Receipt\_movement

on Receipt\_movement.product\_id = products.id

set products.quantity= products.quantity + Receipt\_movement.quantity;

END//

DROP PROCEDURE IF EXISTS product\_Selling//

-- DELIMITER //

CREATE PROCEDURE product\_Selling ()

BEGIN

update products

join Selling\_movement

on Selling\_movement.product\_id = products.id

set products.quantity= products.quantity - Selling\_movement.quantity;

END//

DROP PROCEDURE IF EXISTS product\_Return//

-- DELIMITER //

CREATE PROCEDURE product\_Return ()

BEGIN

update products

join Return\_movement

on Return\_movement.product\_id = products.id

set products.quantity= products.quantity + Return\_movement.quantity;

END//

DROP PROCEDURE IF EXISTS product\_Selling//

-- DELIMITER //

CREATE PROCEDURE product\_Selling ()

BEGIN

update products

join writeoff\_movement

on writeoff\_movement.product\_id = products.id

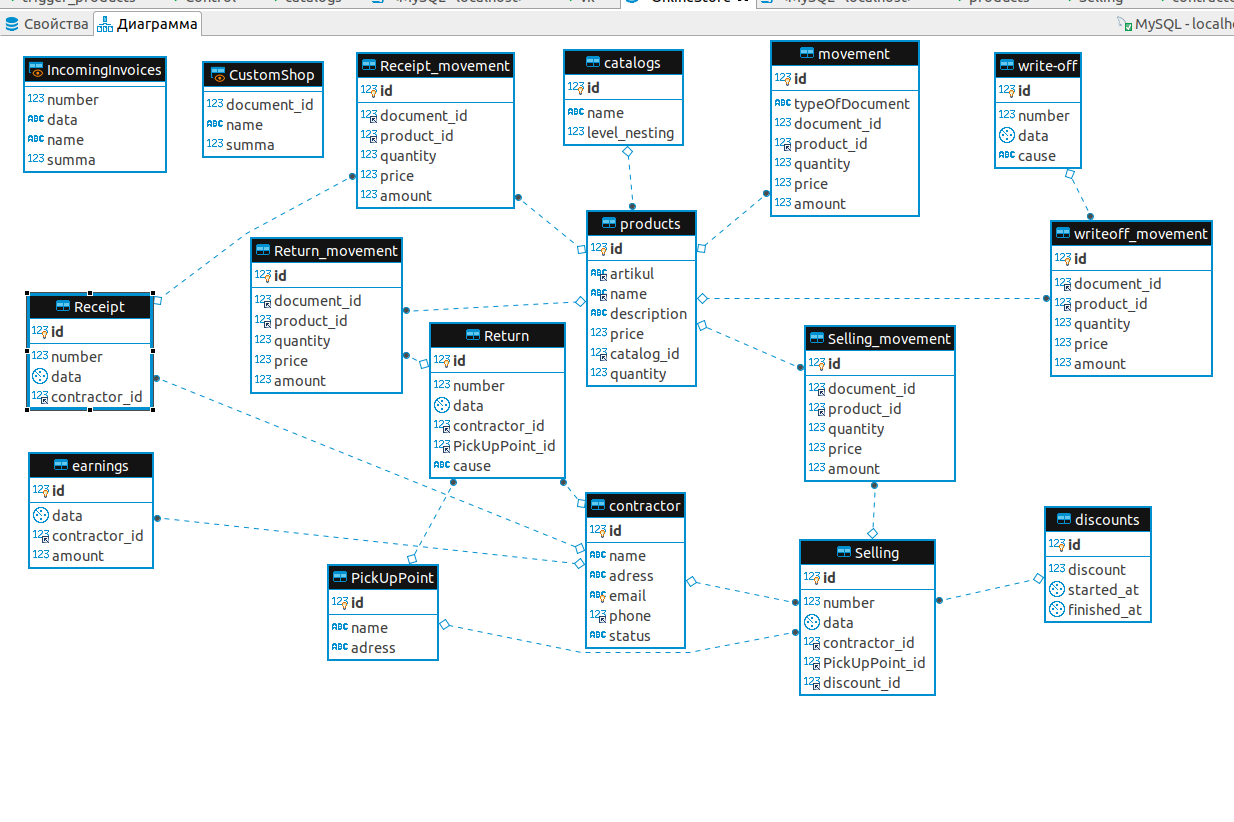
set products.quantity= products.quantity - writeoff\_movement.quantity;

END//

DELIMITER ;

DELIMITER ;

**4.** **создать ERDiagram для БД;**



**5. скрипты наполнения БД данными;**

INSERT INTO catalogs (name,level\_nesting)

VALUES

('Кухонная техника',0),

('Техника для дома',0),

('Цифровые устройства',0),

('Компьютеры и оргтехника',0),

('Климатическая техника',0),

('Автотовары',0),

('Холодильники',1),

('Электрические плиты',1),

('Газовые плиты',1),

('Микроволновые печи',1),

('Чайники электрические',1),

('Кофеварки',1),

('Мультиварки',1),

('Блендеры',1),

('Миксеры',1),

('Мясорубки',1),

('Хлебопечки',1),

('Кастрюли',1),

('Посудомоечные машины',1),

('Стиральные машины',2),

('Пылесосы',2),

('Утюги',2),

('Швейные машины',2),

('Отпариватели',2),

('Сушильные барабаны для белья',2),

('Аксессуары для стиральных машин',2),

('Смартфоны',3),

('Мобильные телефоны',3),

('Чехлы для телефонов',3),

('Планшеты',3),

('Фотоаппараты компактные',3),

('Видео камеры Наушники и гарнитуры',3),

('Внешние аккумуляторы',3),

('Электронные книги',3),

('Карты памяти microSD(HC)',3),

('MP3 плееры',3),

('Ноутбуки',4),

('Настольные компьютеры',4),

('Принтеры',4),

('Комплекты клавиатура мышь',4),

('Внешние жесткие диски',4),

('Роутеры и сетевое оборудование',4),

('Мониторы',4),

('Флеш карты',4),

('Кресла',4),

('Кондиционеры',5),

('Водонагреватели',5),

('Увлажнители воздуха',5),

('Обогреватели',5),

('Вентиляторы',5),

('Очистители и мойки воздуха',5),

('Метеостанции',5),

('Автомагнитолы',6),

('Видеорегистраторы',6),

('Автоакустика',6),

('Сабвуферы для авто',6),

('Навигаторы',6),

('УсилителиРадар-детекторыКомплекты подключения автозвука',6);

DELETE FROM products;

INSERT INTO products (name,price,catalog\_id,quantity)

VALUES

-- ('FFFFFFF',0,(select id from catalogs where name = 'Утюги')),

('Холодильник INDESIT DF 4180 W',35000,(select id from catalogs where name = 'Холодильники'),0),

('Холодильник HAIER C2 F 536 CMSG',27000,(select id from catalogs where name = 'Холодильники'),0),

('Холодильник LERAN SDF 107 WHITE',15000,(select id from catalogs where name = 'Холодильники'),0),

('Холодильник BOSCH KGV 36NW1AR',37000,(select id from catalogs where name = 'Холодильники'),0),

('Холодильник БИРЮСА 118',10000,(select id from catalogs where name = 'Холодильники'),0),

('Плита электрическая FLAMA 33606 W',7500,(select id from catalogs where name = 'Электрические плиты'),0),

('Плита электрическая LERAN ECC 1615 W',12500,(select id from catalogs where name = 'Электрические плиты'),0),

('Микроволновая печь BINATONE FMO 2030 W',5000,(select id from catalogs where name = 'Газовые плиты'),0),

('Микроволновая печь LERAN FMO 2035 W',3500,(select id from catalogs where name = 'Газовые плиты'),0),

('Микроволновая печь SAMSUNG ME-88SUW',7500,(select id from catalogs where name = 'Газовые плиты'),0),

('Стиральная машина CANDY CS4 1051D1/2-07',11900,(select id from catalogs where name = 'Стиральные машины'),0),

('Стиральная машина ZANUSSI ZWSE 680V',10990,(select id from catalogs where name = 'Стиральные машины'),0),

('Стиральная машина INDESIT IWUD 4105',11990,(select id from catalogs where name = 'Стиральные машины'),6),

('Пылесос DOFFLER VCC 1418 VG',3990,(select id from catalogs where name = 'Пылесосы'),0),

('Пылесос SAMSUNG SC 4140',3470,(select id from catalogs where name = 'Пылесосы'),0),

('Пылесос THOMAS TWIN Sigma Aquafilter',14990,(select id from catalogs where name = 'Пылесосы'),0),

('Утюг TEFAL FV 4496',1990,(select id from catalogs where name = 'Утюги'),0),

('Утюг REDMOND RI-C262',1990,(select id from catalogs where name = 'Утюги'),0),

('Утюг BINATONE CEI 1011',690,(select id from catalogs where name = 'Утюги'),0),

('Смартфон Honor 10 Lite 32Gb Blue',11990,(select id from catalogs where name = 'Смартфоны'),0),

('Смартфон Apple iPhone 7 32GB Black MN8X2RU/A',28990,(select id from catalogs where name = 'Смартфоны'),0),

('Смартфон Samsung Galaxy A50 64GB (2019) SM-A505 Black',15990,(select id from catalogs where name = 'Смартфоны'),0),

('Мобильный телефон Jinga Simple F200n black',549,(select id from catalogs where name = 'Мобильные телефоны'),0),

('Мобильный телефон Vertex M122 black',2799,(select id from catalogs where name = 'Мобильные телефоны'),0),

('Мобильный телефон INOI 245R Black',1999,(select id from catalogs where name = 'Мобильные телефоны'),0),

('Планшет Lenovo Tab4 10 TB-X304L (10.1) IPS/32Gb/2Gb/LTE/Black ',7900,(select id from catalogs where name = 'Планшеты'),0),

('Планшет Lenovo Tab 4 TB-X304L (10.1) IPS/16Gb/3G/LTE/White /ZA2K0082RU/',9500,(select id from catalogs where name = 'Планшеты'),0),

('Планшет Samsung GALAXY Tab A 10.1 /SM-T585NZKASER/ LTE 16Gb Black',17990,(select id from catalogs where name = 'Планшеты'),0),

('Ноутбук HP 15-db0041ur /4HC63EA/ AMD E2 ',21500,(select id from catalogs where name = 'Ноутбуки'),0),

('Ноутбук Lenovo 330-15IKB /81DE029GRU/ intel i3 ',25000,(select id from catalogs where name = 'Ноутбуки'),0),

('Ноутбук HP 15-db0377ur /5GZ88EA/ Ryzen 3 2200U/8Gb/1Tb/15.6/Vega ',39900,(select id from catalogs where name = 'Ноутбуки'),0),

('Внешний жесткий диск Western Digital (WDBMTM0020BBK) 2Tb USB ',7500,(select id from catalogs where name = 'Внешние жесткие диски'),0),

('Внешний жесткий диск Seagate STSHX-M101TCBM 1Tb USB 3.0 Maxtor ',5800,(select id from catalogs where name = 'Внешние жесткие диски'),0),

('Внешний жесткий диск Western Digital (WDBMTM0010BBK) 1Tb USB ',3500,(select id from catalogs where name = 'Внешние жесткие диски'),0),

('Системный блок AMCV i021N Game PRO intel i7/16Gb/1Tb/GTX1060 ',56990,(select id from catalogs where name = 'Настольные компьютеры'),0),

('Моноблок HP 22-c0094ur /5KN68EA/ intel J5005/4Gb/500Gb/21.5FHD/KB+mouse/Win10 ',35900,(select id from catalogs where name = 'Настольные компьютеры'),0),

('Системный блок HP 460-a212ur /5KN71EA/ intel J3060/4Gb/500Gb/Intel HD/Win 10',18990,(select id from catalogs where name = 'Настольные компьютеры'),0),

('Клавиатура+мышь Logitech MK270 Wireless (920-004518)',1700,(select id from catalogs where name = 'Комплекты клавиатура мышь'),0),

('Клавиатура+мышь Qumo Solaris',1850,(select id from catalogs where name = 'Комплекты клавиатура мышь'),0),

('Клавиатура+мышь CANYON CNS-HSETW4RU slim wireless',1599,(select id from catalogs where name = 'Комплекты клавиатура мышь'),0),

('Кондиционер KOMANCHI KAC-K2/07H/N1',9500,(select id from catalogs where name = 'Кондиционеры'),0),

('Кондиционер LERAN AC-900',10000,(select id from catalogs where name = 'Кондиционеры'),0),

('Кондиционер BALLU BPAC-09CM',12500,(select id from catalogs where name = 'Кондиционеры'),0),

('Водонагреватель ZANUSSI ZWH/S 80 Splendore XP 2.0',3600,(select id from catalogs where name = 'Водонагреватели'),0),

('Водонагреватель LERAN REWH-10015 enamel',7500,(select id from catalogs where name = 'Водонагреватели'),0),

('Водонагреватель POLARIS SIGMA IDF 50V',6990,(select id from catalogs where name = 'Водонагреватели'),0),

('Увлажнитель воздуха LERAN USH-900T',3750,(select id from catalogs where name = 'Увлажнители воздуха'),0),

('Увлажнитель воздуха TIMBERK THU UL 15E (BL)',4580,(select id from catalogs where name = 'Увлажнители воздуха'),0),

('Увлажнитель воздуха LERAN USH-500',1990,(select id from catalogs where name = 'Увлажнители воздуха'),0),

('Автомагнитола PROLOGY MPV-400',5550,(select id from catalogs where name = 'Автомагнитолы'),0),

('Автомагнитола PIONEER FH-X730BT',6600,(select id from catalogs where name = 'Автомагнитолы'),0),

('Автомагнитола SOUNDMAX SM-CCR3064F',8390,(select id from catalogs where name = 'Автомагнитолы'),0),

('Видеорегистраторы LEXAND LR-65',4990,(select id from catalogs where name = 'Видеорегистраторы'),0),

('Видеорегистраторы Navitel MR250 NV',5500,(select id from catalogs where name = 'Видеорегистраторы'),0),

('Видеорегистраторы КАРКАМ D5',3750,(select id from catalogs where name = 'Видеорегистраторы'),0),

('Колонки PIONEER TS-650C',3790,(select id from catalogs where name = 'Автоакустика'),0),

('Колонки SOUNDMAX SM-CF 502',790,(select id from catalogs where name = 'Автоакустика'),0),

('Колонки JBL CS752',1590,(select id from catalogs where name = 'Автоакустика'),0)

;

DELETE FROM PickUpPoint;

INSERT INTO PickUpPoint (name,adress)

VALUES

('Доставка до клиента','адрес в карточке клиента'),

('ТД Гулливер','ул. Пролетарская, 247'),

('ТД Восход','ул. Карагандинская, 58/1'),

('ТРЦ Армада','ул. Шарлыкское ш. 1/2');

DELETE FROM discounts;

INSERT INTO discounts (discount,started\_at,finished\_at)

VALUES

(5,'20170101', '20170501'),

(12,'20180101', '20180501'),

(10,'20190201', '20190228');

DELETE FROM contractor;

INSERT INTO contractor (name,adress,email,phone,status)

VALUES

('Vestel','ул. Профсоюзная д.27', 'Vestel@ya.ru','4972584545','partner'),

('Brizze','ул. Шевченко д.123', 'Brizze@ya.ru','4972545645','partner'),

('Екб Транс','ул. Хмельницского д.27', 'Tranc@ya.ru','4998784545','partner'),

('MT','ул. "Энгелься д.157', 'MT@ya.ru','4321584545','partner'),

('Сидоров Борис','ул. Победы д.270', 'SB777@ya.ru','4972584545','client'),

('Польчак Виталий','ул. Пролетарская д.11', 'PolVit@ya.ru','4123584654','client'),

('Васильченко Василий','ул. Гришина д.98', 'VV77@ya.ru','4972584545','client'),

('Абрамов Константин','ул. Невельская д.12', 'AbKons@ya.ru','4707599577','client');

DELETE FROM Receipt;

INSERT INTO Receipt (`number`,`data`,contractor\_id)

VALUES

(0001,'20190101', (select id from contractor where name = 'Vestel')),

(0002,'20190821', (select id from contractor where name = 'Brizze')),

(0003,'20190901', (select id from contractor where name = 'MT'));

DELETE FROM Selling;

INSERT INTO Selling (`number`,`data`,contractor\_id,PickUpPoint\_id)

VALUES

(0001,'20190101', (select id from contractor where name = 'Сидоров Борис'),(select id from contractor where name = 'Доставка до клиента')),

(0002,'20190821', (select id from contractor where name = 'Польчак Виталий'),(select id from contractor where name = 'ТД Гулливер')),

(0003,'20190901', (select id from contractor where name = 'Васильченко Василий'),(select id from contractor where name = 'ТД Восход')),

(0004,'20190921', (select id from contractor where name = 'Абрамов Константин'),(select id from contractor where name = 'ТРЦ Армада'))

;

DELETE FROM `Return`;

INSERT INTO `Return` (`number`,`data`,contractor\_id,PickUpPoint\_id,cause)

VALUES

(0001,'20191001', (select id from contractor where name = 'Сидоров Борис'),(select id from contractor where name = 'Доставка до клиента'),'Гарантийный случай');

DELETE FROM `write-off`;

INSERT INTO `write-off` (`number`,`data`,cause)

VALUES

(0001,'20191001','Гарантийный случай');

-- typeOfDocument enum('Admission','Selling','write-off','audit','Return'),

-- product\_balances

DELETE FROM `Receipt\_movement`;

INSERT INTO `Receipt\_movement` (`document\_id`,`product\_id`,quantity,price,amount)

VALUES

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Утюг TEFAL FV 4496'),10,5000,50000),

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Кондиционер KOMANCHI KAC-K2/07H/N1'),7,200,14000),

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Кондиционер LERAN AC-900'),8,5000,40000),

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Холодильник INDESIT DF 4180 W'),10,5000,50000),

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Холодильник HAIER C2 F 536 CMSG'),7,200,14000),

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Холодильник BOSCH KGV 36NW1AR'),8,22500,180000),

((select id from Receipt where `number` = '0001'),(select id from products where `name` = 'Холодильник БИРЮСА 118'),5,15000,75000),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Плита электрическая FLAMA 33606 W'),10,5000,50000),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Плита электрическая LERAN ECC 1615 W'),7,5000,35000),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Микроволновая печь BINATONE FMO 2030 W'),8,5000,40000),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Микроволновая печь LERAN FMO 2035 W'),10,5000,50000),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Микроволновая печь SAMSUNG ME-88SUW'),7,3500,24500),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Стиральная машина CANDY CS4 1051D1/2-07'),8,25000,200000),

((select id from Receipt where `number` = '0002'),(select id from products where `name` = 'Стиральная машина ZANUSSI ZWSE 680V'),5,15000,75000),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Увлажнитель воздуха LERAN USH-900T'),10,500,5000),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Увлажнитель воздуха TIMBERK THU UL 15E (BL)'),7,500,3500),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Увлажнитель воздуха LERAN USH-500'),8,500,4000),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Автомагнитола PROLOGY MPV-400'),10,5000,50000),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Автомагнитола PIONEER FH-X730BT'),7,3500,24500),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Автомагнитола SOUNDMAX SM-CCR3064F'),8,2500,20000),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Видеорегистраторы Navitel MR250 NV'),7,3500,24500),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Видеорегистраторы КАРКАМ D5'),8,2500,20000),

((select id from Receipt where `number` = '0003'),(select id from products where `name` = 'Видеорегистраторы LEXAND LR-65'),5,1500,7500);

call product\_Receipt();

DELETE FROM `Selling\_movement`;

INSERT INTO `Selling\_movement` (`document\_id`,`product\_id`,quantity,price,amount)

VALUES

((select id from Selling where `number` = '0001'),(select id from products where `name` = 'Увлажнитель воздуха LERAN USH-900T'),10,500,5000),

((select id from Selling where `number` = '0001'),(select id from products where `name` = 'Увлажнитель воздуха TIMBERK THU UL 15E (BL)'),7,500,3500),

((select id from Selling where `number` = '0002'),(select id from products where `name` = 'Увлажнитель воздуха LERAN USH-500'),8,500,4000),

((select id from Selling where `number` = '0002'),(select id from products where `name` = 'Автомагнитола PROLOGY MPV-400'),10,5000,50000),

((select id from Selling where `number` = '0002'),(select id from products where `name` = 'Автомагнитола PIONEER FH-X730BT'),7,3500,24500),

((select id from Selling where `number` = '0003'),(select id from products where `name` = 'Автомагнитола SOUNDMAX SM-CCR3064F'),8,2500,20000),

((select id from Selling where `number` = '0003'),(select id from products where `name` = 'Видеорегистраторы Navitel MR250 NV'),7,3500,24500),

((select id from Selling where `number` = '0003'),(select id from products where `name` = 'Видеорегистраторы КАРКАМ D5'),8,2500,20000),

((select id from Selling where `number` = '0003'),(select id from products where `name` = 'Видеорегистраторы LEXAND LR-65'),5,1500,7500),

((select id from Selling where `number` = '0004'),(select id from products where `name` = 'Микроволновая печь BINATONE FMO 2030 W'),7,3500,24500),

((select id from Selling where `number` = '0004'),(select id from products where `name` = 'Стиральная машина CANDY CS4 1051D1/2-07'),8,2500,20000),

((select id from Selling where `number` = '0004'),(select id from products where `name` = 'Увлажнитель воздуха LERAN USH-900T'),7,3500,24500),

((select id from Selling where `number` = '0004'),(select id from products where `name` = 'Кондиционер KOMANCHI KAC-K2/07H/N1'),8,2500,20000),

((select id from Selling where `number` = '0004'),(select id from products where `name` = 'Утюг TEFAL FV 4496'),5,1500,7500),

((select id from Selling where `number` = '0004'),(select id from products where `name` = 'Автомагнитола PIONEER FH-X730BT'),5,3500,27500);

call product\_Selling();

INSERT INTO `Return\_movement` (`document\_id`,`product\_id`,quantity,price,amount)

VALUES

((select id from `Return` where `number` = '0001'),(select id from products where `name` = 'Увлажнитель воздуха LERAN USH-900T'),10,500,5000);

call product\_Return ();

**6**. **скрипты характерных выборок (включающие группировки, JOIN'ы, вложенные таблицы);**

Select m.document\_id as doc, s.contractor\_id as con from Selling\_movement m left join Selling s on m.document\_id = s.id;

Select

m.document\_id,

m.contractor\_id,

sum(m.amount)

from

(Select m.document\_id, s.contractor\_id, m.amount from Selling\_movement m left join Selling s on m.document\_id = s.id) m

GROUP BY m.contractor\_id, m.document\_id;

Select

m.document\_id,

(select name from contractor WHERE id = m.contractor\_id) name,

sum(m.amount) summa

from

(Select m.document\_id, s.contractor\_id, m.amount from Selling\_movement m left join Selling s on m.document\_id = s.id) m

GROUP BY m.contractor\_id, m.document\_id;

**7. представления (минимум 2);**

CREATE VIEW CustomShop as

Select

m.document\_id,

(select name from contractor WHERE id = m.contractor\_id) name,

sum(m.amount) summa

from

(Select m.document\_id, s.contractor\_id, m.amount from Selling\_movement m left join Selling s on m.document\_id = s.id) m

GROUP BY m.contractor\_id, m.document\_id;

CREATE VIEW IncomingInvoices as

Select

m.`number`,

DATE\_FORMAT(m.`data`,'%d.%m.%Y') as `data`,

(select name from contractor WHERE id = m.contractor\_id) Provider,

sum(m.amount) summa

from

(Select m.document\_id, r.contractor\_id, r.`data`,r.`number` ,m.amount from Receipt\_movement m left join Receipt r on m.document\_id = r.id) m

GROUP BY m.contractor\_id, m.document\_id;

**8. хранимые процедуры / триггеры;**

DROP TRIGGER if EXISTS Control\_movent\_Receipt;

DELIMITER //

CREATE TRIGGER Control\_movent\_Receipt

AFTER DELETE

ON Receipt FOR EACH ROW

BEGIN

DELETE FROM Receipt\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_movent\_Selling//

CREATE TRIGGER Control\_movent\_Selling

AFTER DELETE

ON Selling FOR EACH ROW

BEGIN

DELETE FROM Selling\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_movent\_Return//

CREATE TRIGGER Control\_movent\_Return

AFTER DELETE

ON `Return` FOR EACH ROW

BEGIN

DELETE FROM Return\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_movent\_writeoff//

CREATE TRIGGER Control\_movent\_writeoff

AFTER DELETE

ON `write-off` FOR EACH ROW

BEGIN

DELETE FROM writeoff\_movement where document\_id = OLD.id;

END//

DROP TRIGGER if EXISTS Control\_earnings\_movement//

CREATE TRIGGER Control\_earnings\_movement

AFTER INSERT ON movement FOR EACH ROW

BEGIN

if NEW.typeOfDocument = 'Selling' then

INSERT INTO earnings (`data`,contractor\_id,amount)

VALUES

(NOW(),(select contractor\_id from Selling where id = NEW.document\_id),NEW.amount);

end if;

END//

DROP PROCEDURE IF EXISTS Change\_count\_products;

DELIMITER //

CREATE PROCEDURE Change\_count\_products (in productID bigint,in quantityM int(10))

BEGIN

UPDATE products

set quantity = quantity + quantityM where id = productID;

-- SELECT quantityM;

END//

DROP TRIGGER if EXISTS Control\_earnings\_Return//

CREATE TRIGGER Control\_earnings\_Return

AFTER INSERT ON movement FOR EACH ROW

BEGIN

DECLARE var1 int;

if NEW.typeOfDocument = 'Return' then

select NEW.amount into var1;

INSERT INTO earnings (`data`,contractor\_id,amount)

VALUES

(NOW(),(select contractor\_id from `Return` where id = NEW.document\_id),(var1\*(-1)));

end if;

-- Change\_count\_products(New.product\_id,New.quantity);

END//

DROP PROCEDURE IF EXISTS product\_balances//

-- DELIMITER //

CREATE PROCEDURE product\_balances ()

BEGIN

select name, quantity from products where quantity > 0;

END//

DROP PROCEDURE IF EXISTS product\_Receipt//

-- DELIMITER //

CREATE PROCEDURE product\_Receipt ()

BEGIN

update products

join Receipt\_movement

on Receipt\_movement.product\_id = products.id

set products.quantity= products.quantity + Receipt\_movement.quantity;

END//

DROP PROCEDURE IF EXISTS product\_Selling//

-- DELIMITER //

CREATE PROCEDURE product\_Selling ()

BEGIN

update products

join Selling\_movement

on Selling\_movement.product\_id = products.id

set products.quantity= products.quantity - Selling\_movement.quantity;

END//

DROP PROCEDURE IF EXISTS product\_Return//

-- DELIMITER //

CREATE PROCEDURE product\_Return ()

BEGIN

update products

join Return\_movement

on Return\_movement.product\_id = products.id

set products.quantity= products.quantity + Return\_movement.quantity;

END//

DROP PROCEDURE IF EXISTS product\_Selling//

-- DELIMITER //

CREATE PROCEDURE product\_Selling ()

BEGIN

update products

join writeoff\_movement

on writeoff\_movement.product\_id = products.id

set products.quantity= products.quantity - writeoff\_movement.quantity;

END//

DELIMITER ;