**ТЕХНИЧЕСКИ УНИВЕРСИТЕТ – СОФИЯ**

**Катедра „Компютърни системи”**

**КУРСОВ ПРОЕКТ**

**ПО БАЗИ ОТ ДАННИ**

**Студент: Васил Стойчев Стойков**

**ФАК. № 121222094** **Група: 43 А**

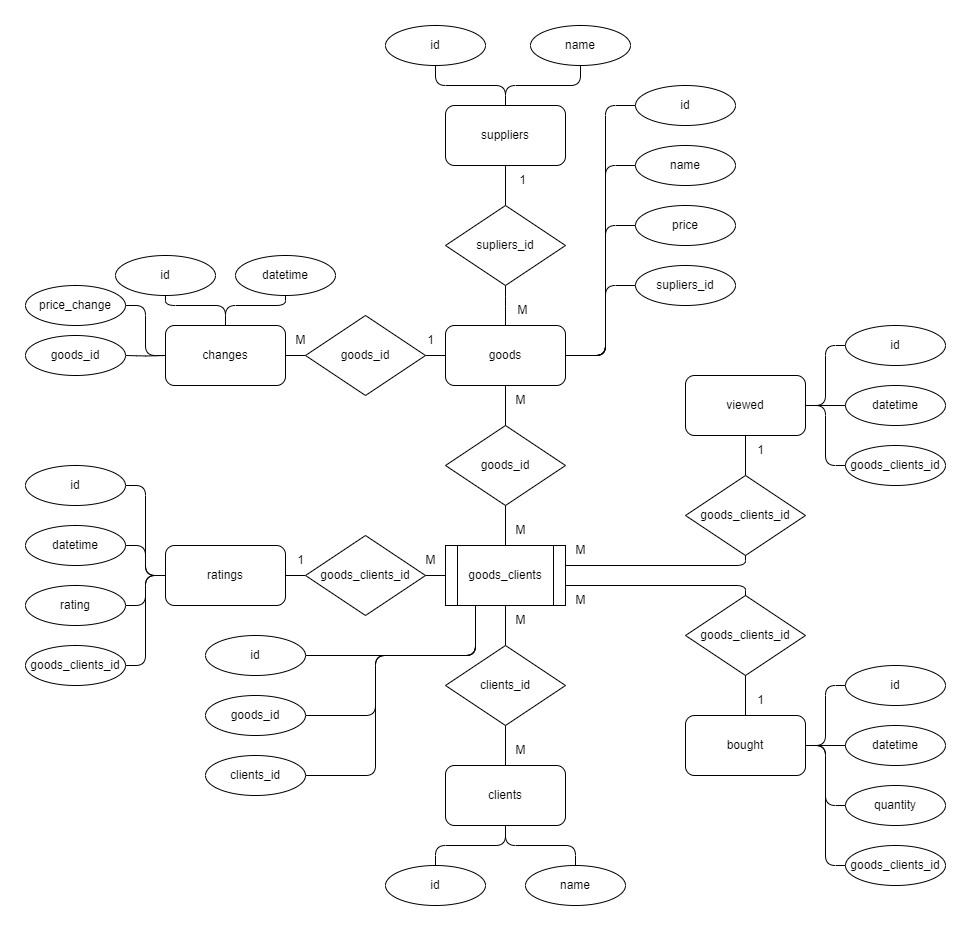
**Тема №4**

Разработете база данни за система за поддържане на доставчици, стоки и цени, които ще се разглеждат от различни клиенти - „дилъри“. Да се пази история кой клиент – коя стока е разгледал и кой клиент – какви стоки е поръчал. Да се направи и рейтинг на стоките. Да може да се извеждат различни статистически данни – напр. най-продавани артикули за конкретен период, промяна в цената на дадена стока на годишна база и т.н.

1. Да се проектира база от данни и да се представи ER диаграма със съответни CREATE TABLE заявки за средата MySQL.
2. Напишете заявка, в която демонстрирате SELECT с ограничаващо условие по избор.
3. Напишете заявка, в която използвате агрегатна функция и GROUP BY по ваш избор.
4. Напишете заявка, в която демонстрирате INNER JOIN по ваш избор.
5. Напишете заявка, в която демонстрирате OUTER JOIN по ваш избор.
6. Напишете заявка, в която демонстрирате вложен SELECT по ваш избор.
7. Напишете заявка, в която демонстрирате едновременно JOIN и агрегатна функция.
8. Създайте тригер по ваш избор.
9. Създайте процедура, в която демонстрирате използване на курсор.

1. **Да се проектира база от данни и да се представи ER диаграма със съответни CREATE TABLE заявки за средата MySQL:**

Според заданието основните обекти, за които трябва да съхраняваме информация са: доставчици (suppliers), стоки (goods) и клиенти (clients). Освен това ние ще трябва да съхраняваме и информация за промените по цените на стоките, рейтинг, който клиентите са дали за стоките и кой клиент коя стока е прегледал или поръчал. Разбира се ние ще съхраняваме информация за това кога са се извършило всяко едно от тези събития, цените на стоките и промяната на цените на стоките. Ще имаме и още 1 допълнителна log таблица отразяваща добавянето на рейтинги, която ще ни е необходима за задача осем. В тази база данни имаме една много ключова таблица показваща връзката на стоките с клиентите като само по себе си това не означава нищо, но идентификационния номер на всеки запис от тази таблица бива посочен от запис от другите три таблици отговарящи за рейтингите, преглежданията и покупките, като това намаля броя на необходимите външни ключове и ни позволява да оптимизираме базата, като един такъв запис бива посочен от повече от една други таблици. За проектирането на базата ще използваме модела ER-диаграма (Entity Relationship Diagram):



Заявките, с които създаваме базата данни и таблиците са:

CREATE DATABASE course\_project\_number\_4;

CREATE TABLE suppliers (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) NOT NULL UNIQUE

);

CREATE TABLE goods (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) NOT NULL UNIQUE,

price FLOAT NOT NULL,

suppliers\_id INT NOT NULL,

FOREIGN KEY (suppliers\_id) REFERENCES suppliers(id)

);

CREATE TABLE changes (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

datetime DATETIME DEFAULT NULL,

price\_change FLOAT NOT NULL,

goods\_id INT NOT NULL,

FOREIGN KEY (goods\_id) REFERENCES goods(id)

);

CREATE TABLE clients (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(255) NOT NULL UNIQUE

);

CREATE TABLE goods\_clients (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

goods\_id INT NOT NULL,

clients\_id INT NOT NULL,

FOREIGN KEY (goods\_id) REFERENCES goods(id),

FOREIGN KEY (clients\_id) REFERENCES clients(id)

);

CREATE TABLE ratings (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

datetime DATETIME DEFAULT NULL,

rating ENUM('1', '2', '3', '4', '5') NOT NULL,

goods\_clients\_id INT NOT NULL,

FOREIGN KEY (goods\_clients\_id) REFERENCES goods\_clients(id)

);

CREATE TABLE bought (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

datetime DATETIME DEFAULT NULL,

quantity INT default 1,

goods\_clients\_id INT NOT NULL,

FOREIGN KEY (goods\_clients\_id) REFERENCES goods\_clients(id)

);

CREATE TABLE viewed (

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

datetime DATETIME DEFAULT NULL,

goods\_clients\_id INT NOT NULL,

FOREIGN KEY (goods\_clients\_id) REFERENCES goods\_clients(id)

);

Добавяме и тестови данни в таблиците:

insert into suppliers (id, name)

values (1, 'Sweet Treats Inc.'),

(2, 'Candy Co.'),

(3, 'Sugar Rush Enterprises'),

(4, 'Choco Delight Corporation'),

(5, 'Gummy Goodness Ltd.'),

(6, 'Lollipop Land Industries'),

(7, 'Caramel Creations Co.'),

(8, 'Jellybean Junction'),

(9, 'Marshmallow Magic Inc.'),

(10, 'Toffee Time Enterprises');

insert into goods (id, name, price, supliers\_id) values

(1, 'Snickers', 2.49, 2),

(2, 'Twix', 2.87, 5),

(3, 'Kit Kat', 6.48, 2),

(4, 'Reese''s', 7.67, 2),

(5, 'M&M''s', 2.9, 6),

(6, 'Skittles', 7.23, 6),

(7, 'Hershey''s', 5.91, 10),

(8, 'Milky Way', 4.12, 5),

(9, 'Nerds', 5.84, 9),

(10, 'Jolly Rancher', 2.84, 8),

(11, 'Butterfinger', 3.84, 2),

(12, 'Tootsie Roll', 4.29, 4),

(13, 'Dots', 1.39, 3),

(14, 'Airheads', 3.6, 3),

(15, 'Laffy Taffy', 6.61, 7),

(16, 'Sour Patch Kids', 5.13, 6),

(17, 'Gummy Bears', 7.43, 10),

(18, 'Candy Corn', 4.49, 1),

(19, 'Swedish Fish', 2.24, 6),

(20, 'Smarties', 8.42, 2),

(21, 'Almond Joy', 2.13, 10),

(22, 'Baby Ruth', 1.61, 8),

(23, 'Caramel Apple Pops', 4.05, 10),

(24, 'Pop Rocks', 7.17, 3),

(25, 'Ring Pop', 1.1, 9),

(26, 'Pixy Stix', 2.95, 9),

(27, 'Warheads', 8.6, 3),

(28, 'Gobstopper', 1.13, 5),

(29, 'Cotton Candy', 8.87, 3),

(30, 'Fun Dip', 1.06, 7),

(31, 'Now and Later', 2.88, 6),

(32, 'Rolo', 6.26, 2),

(33, 'Sugar Daddy', 9.66, 9),

(34, 'Tootsie Pop', 4.91, 2),

(35, 'Whoppers', 1.05, 3),

(36, 'York Peppermint Pattie', 6.79, 9),

(37, '100 Grand', 1.24, 4),

(38, 'Atomic Fireball', 6.21, 8),

(39, 'Bit-O-Honey', 7.1, 3),

(40, 'Charleston Chew', 9.04, 2),

(41, 'Chick-O-Stick', 8.18, 1),

(42, 'Circus Peanuts', 1.85, 3),

(43, 'Cry Baby', 1.82, 8),

(44, 'Dum Dums', 3.84, 8),

(45, 'Goo Goo Cluster', 5.24, 4),

(46, 'Good & Plenty', 6.84, 3),

(47, 'Hot Tamales', 3.85, 2),

(48, 'Jawbreaker', 6.19, 9),

(49, 'Jujubes', 9.66, 2),

(50, 'Mary Janes', 4.8, 8);

insert into changes (id, datetime, price\_change, goods\_id) values (1, '2021-05-23 16:05:43', -0.54, 4),

(2, '2023-12-10 12:43:12', 1.23, 5),

(3, '2022-11-25 05:55:03', -1.68, 46),

(4, '2021-11-07 19:30:30', -1.63, 16),

(5, '2020-02-18 16:56:18', -1.49, 12),

(6, '2021-12-15 12:56:23', 1.96, 15),

(7, '2020-01-24 06:15:30', -0.83, 29),

(8, '2023-08-20 00:43:50', -1.57, 4),

(9, '2023-05-09 22:34:20', -0.34, 45),

(10, '2022-08-20 04:04:46', 0.27, 34),

(11, '2022-10-28 19:12:45', -0.46, 4),

(12, '2024-01-02 08:07:13', 1.64, 16),

(13, '2023-03-13 03:59:21', 0.88, 37),

(14, '2022-08-10 22:15:32', -1.33, 43),

(15, '2021-05-16 20:40:42', 0.3, 1),

(16, '2021-05-11 20:25:12', -1.35, 3),

(17, '2023-07-26 20:53:32', 0.72, 49),

(18, '2024-02-04 20:12:31', 1.49, 15),

(19, '2022-06-12 09:47:10', -0.43, 46),

(20, '2023-11-05 08:17:58', 0.58, 26),

(21, '2023-05-29 07:14:33', 1.77, 9),

(22, '2020-12-12 13:12:51', 0.57, 3),

(23, '2022-08-02 23:18:11', -0.76, 22),

(24, '2020-02-25 15:27:32', -1.74, 33),

(25, '2021-09-06 19:43:15', 0.07, 9),

(26, '2022-01-07 22:11:36', 1.5, 18),

(27, '2022-12-27 09:07:16', 1.73, 6),

(28, '2020-11-03 05:23:01', -0.35, 14),

(29, '2023-02-16 16:30:26', -1.31, 31),

(30, '2021-10-12 17:55:43', 0.56, 9),

(31, '2021-10-24 00:40:48', -0.22, 11),

(32, '2023-06-06 04:57:23', -1.68, 26),

(33, '2023-07-31 07:46:37', -0.41, 20),

(34, '2020-01-08 14:10:30', -1.71, 39),

(35, '2023-02-20 05:06:58', 1.15, 16),

(36, '2020-03-10 13:34:16', -0.22, 28),

(37, '2021-01-05 04:47:53', 0.12, 17),

(38, '2023-06-04 13:03:34', -1.24, 38),

(39, '2021-05-14 02:19:16', 1.37, 2),

(40, '2022-01-29 06:53:05', -0.15, 25),

(41, '2022-08-01 16:37:03', 0.26, 43),

(42, '2021-07-24 03:36:12', -1.09, 38),

(43, '2023-06-23 04:13:46', 1.76, 16),

(44, '2023-03-16 15:16:57', -1.39, 28),

(45, '2021-04-03 05:41:13', -1.84, 32),

(46, '2023-04-14 19:31:31', -0.03, 1),

(47, '2023-03-19 22:21:11', -0.27, 10),

(48, '2020-02-16 04:21:32', -0.47, 27),

(49, '2020-11-25 14:14:44', -1.01, 10),

(50, '2024-03-01 15:30:52', 0.36, 7),

(51, '2021-02-12 02:50:08', -0.13, 4),

(52, '2021-11-15 10:13:12', 1.49, 42),

(53, '2022-12-25 23:43:53', 1.17, 24),

(54, '2022-06-06 09:12:21', 0.77, 10),

(55, '2021-01-20 12:37:49', -1.1, 18),

(56, '2022-02-06 16:24:31', 0.08, 17),

(57, '2020-11-10 01:08:40', 0.58, 12),

(58, '2023-09-23 09:38:06', -1.32, 46),

(59, '2024-01-18 22:43:35', 0.05, 46),

(60, '2023-02-09 03:23:21', 1.73, 46),

(61, '2020-07-11 08:50:22', -0.62, 31),

(62, '2024-03-08 06:51:17', 0.79, 13),

(63, '2022-12-05 12:57:40', 1.89, 48),

(64, '2021-09-29 23:15:32', -1.59, 47),

(65, '2020-06-26 00:22:26', -0.28, 26),

(66, '2023-07-01 10:40:07', 1.46, 12),

(67, '2023-11-26 20:33:52', 0.09, 41),

(68, '2024-01-03 16:40:44', -0.21, 42),

(69, '2022-04-17 08:19:39', 0.82, 39),

(70, '2021-03-11 08:07:50', 1.2, 7),

(71, '2022-03-20 00:07:34', -0.1, 22),

(72, '2022-05-09 06:21:49', -1.72, 8),

(73, '2022-06-10 20:49:19', 0.7, 34),

(74, '2021-01-03 03:03:28', -1.04, 22),

(75, '2020-07-06 06:47:38', -1.53, 42),

(76, '2022-06-27 05:16:17', 0.28, 30),

(77, '2021-12-04 08:30:19', 0.23, 22),

(78, '2023-01-30 02:39:06', -1.66, 49),

(79, '2020-07-03 15:38:59', 0.73, 48),

(80, '2020-08-27 09:44:05', -1.54, 24),

(81, '2023-03-06 14:29:55', 1.29, 49),

(82, '2020-12-24 18:15:09', -0.95, 3),

(83, '2024-03-09 23:43:39', 1.47, 47),

(84, '2021-06-29 19:47:02', -0.83, 47),

(85, '2020-08-17 05:43:50', -1.55, 32),

(86, '2023-03-10 19:20:02', -0.77, 10),

(87, '2023-11-12 01:36:46', -1.68, 20),

(88, '2021-06-16 21:07:05', 1.62, 30),

(89, '2024-01-28 22:46:21', 0.1, 28),

(90, '2022-05-16 12:23:35', -1.46, 38),

(91, '2022-11-23 09:12:57', -0.4, 32),

(92, '2023-01-07 01:09:20', -1.84, 12),

(93, '2021-01-02 14:38:20', -1.89, 24),

(94, '2020-01-30 02:20:09', -1.61, 8),

(95, '2020-02-18 02:48:08', -0.07, 2),

(96, '2023-05-22 12:20:14', 1.38, 18),

(97, '2022-12-10 09:10:29', -1.15, 13),

(98, '2020-02-15 11:52:34', 1.81, 41),

(99, '2020-01-06 22:33:28', 0.26, 28),

(100, '2020-01-23 14:46:32', -1.64, 45),

(101, '2021-03-31 03:10:49', 0.71, 12),

(102, '2021-05-04 07:36:53', 0.75, 35),

(103, '2023-01-19 11:14:19', -1.81, 44),

(104, '2023-06-24 17:51:33', 0.31, 15),

(105, '2023-06-07 19:29:08', -0.85, 25),

(106, '2021-01-28 16:28:56', -0.14, 37),

(107, '2020-06-29 06:56:44', 0.29, 5),

(108, '2022-10-29 21:21:02', -0.0, 9),

(109, '2022-06-04 13:02:25', 1.15, 7),

(110, '2023-11-24 08:06:26', 1.9, 13),

(111, '2021-10-13 22:55:41', -0.59, 7),

(112, '2021-09-28 07:21:00', 1.36, 19),

(113, '2022-03-11 01:23:21', -0.47, 17),

(114, '2023-08-07 00:49:51', 0.43, 45),

(115, '2023-11-05 21:55:02', 0.52, 48),

(116, '2022-03-19 05:46:34', -0.24, 30),

(117, '2021-07-25 06:51:50', -1.81, 44),

(118, '2020-08-29 22:38:11', 1.61, 46),

(119, '2021-06-17 19:26:12', 1.92, 17),

(120, '2022-07-21 12:05:37', -0.39, 48),

(121, '2021-12-19 05:04:52', -1.06, 47),

(122, '2023-12-05 18:24:25', -0.41, 10),

(123, '2022-06-10 00:06:56', -1.75, 43),

(124, '2021-04-02 04:00:57', -0.87, 45),

(125, '2022-04-20 18:54:52', -1.61, 44),

(126, '2020-11-02 04:31:43', -1.38, 19),

(127, '2021-01-22 07:01:43', 1.25, 41),

(128, '2021-01-20 10:38:00', -0.98, 7),

(129, '2022-04-23 05:44:32', -1.38, 37),

(130, '2023-05-22 03:23:55', -1.87, 29),

(131, '2022-05-08 00:20:44', 1.44, 27),

(132, '2020-12-12 09:07:18', 1.56, 17),

(133, '2022-06-20 23:54:29', -1.19, 13),

(134, '2023-10-28 10:56:52', -1.34, 38),

(135, '2021-02-01 02:16:06', 0.79, 13),

(136, '2022-04-27 23:42:10', -1.53, 2),

(137, '2021-07-14 04:13:51', -1.64, 50),

(138, '2023-12-22 11:20:30', 0.94, 39),

(139, '2023-05-02 12:45:02', -0.36, 1),

(140, '2020-10-01 14:44:51', -1.87, 35),

(141, '2021-08-21 17:17:38', -1.72, 39),

(142, '2022-01-08 05:37:18', -1.81, 30),

(143, '2023-02-25 15:50:31', 1.25, 18),

(144, '2023-01-03 15:16:47', -0.64, 38),

(145, '2020-12-11 16:08:21', -1.87, 9),

(146, '2021-12-11 04:04:57', -0.42, 22),

(147, '2020-09-06 22:31:58', 1.02, 7),

(148, '2020-06-14 07:39:22', -1.85, 11),

(149, '2021-03-05 20:44:26', 0.34, 28),

(150, '2021-09-04 08:40:47', 1.49, 3),

(151, '2023-03-08 04:09:40', -0.49, 16),

(152, '2023-03-01 06:32:33', -1.55, 43),

(153, '2020-04-11 17:46:04', -1.23, 43),

(154, '2023-05-03 13:27:44', -1.56, 6),

(155, '2023-07-29 08:56:24', -0.36, 50),

(156, '2023-05-02 18:22:47', -1.59, 2),

(157, '2022-06-20 22:46:17', 0.96, 2),

(158, '2024-03-18 04:35:11', -0.59, 6),

(159, '2021-07-30 21:26:34', 1.59, 46),

(160, '2020-11-02 04:00:45', 1.74, 9),

(161, '2022-08-01 22:58:40', -0.12, 42),

(162, '2020-12-08 17:29:58', 0.91, 2),

(163, '2020-06-03 12:35:22', 1.14, 22),

(164, '2023-01-27 08:21:34', 0.21, 36),

(165, '2022-10-13 22:32:50', 0.71, 29),

(166, '2022-09-07 22:36:57', 0.15, 34),

(167, '2021-01-01 00:55:04', 1.73, 17),

(168, '2021-11-05 07:21:04', -0.49, 39),

(169, '2022-11-10 11:20:51', -0.36, 14),

(170, '2022-04-13 11:21:15', -0.34, 8),

(171, '2024-03-09 18:09:50', -0.65, 18),

(172, '2023-07-15 15:18:54', -0.79, 24),

(173, '2022-09-14 15:13:16', -1.62, 9),

(174, '2021-06-01 04:47:02', 0.04, 28),

(175, '2022-10-14 16:13:43', -0.71, 45),

(176, '2022-09-27 07:46:47', -0.93, 32),

(177, '2020-09-11 11:42:56', -1.27, 6),

(178, '2022-01-23 12:24:45', -1.61, 3),

(179, '2023-05-24 23:44:18', -0.79, 5),

(180, '2020-04-21 04:57:03', -0.15, 9),

(181, '2022-05-11 22:58:41', -0.23, 30),

(182, '2021-04-21 22:09:27', -1.93, 28),

(183, '2021-03-18 10:10:15', 1.18, 26),

(184, '2021-04-09 01:26:06', 1.52, 25),

(185, '2023-12-19 15:39:05', 0.05, 32),

(186, '2020-08-12 21:21:38', -0.25, 21),

(187, '2021-08-04 09:01:49', -1.85, 42),

(188, '2021-04-15 06:07:54', 1.12, 22),

(189, '2020-08-08 22:12:29', 1.02, 38),

(190, '2021-10-14 04:23:41', 0.83, 24),

(191, '2022-07-29 18:28:49', -1.81, 36),

(192, '2022-12-10 13:28:06', -0.32, 43),

(193, '2022-05-01 15:51:47', -1.55, 28),

(194, '2020-08-21 01:18:25', -0.63, 50),

(195, '2020-09-21 09:05:30', -0.97, 3),

(196, '2020-04-28 12:11:14', -1.3, 46),

(197, '2020-12-25 13:53:22', -0.03, 29),

(198, '2023-05-05 10:43:47', 1.24, 8),

(199, '2024-02-12 23:47:16', 1.52, 34),

(200, '2022-03-02 23:13:16', -0.34, 29);

insert into clients (id, name) values (1, 'Sweet Treats'),

(2, 'Candy Corner'),

(3, 'Sugar Rush'),

(4, 'The Sweet Spot'),

(5, 'Lollipop Lane'),

(6, 'Gummy Haven'),

(7, 'Chocolate Paradise'),

(8, 'Jellybean Junction'),

(9, 'Caramel Cove'),

(10, 'Marshmallow Meadows'),

(11, 'Cotton Candy Castle'),

(12, 'Toffee Terrace'),

(13, 'Fudge Avenue'),

(14, 'Rock Candy Ridge'),

(15, 'Sour Patch Plaza'),

(16, 'Licorice Lane'),

(17, 'Bubblegum Boulevard'),

(18, 'Peppermint Place'),

(19, 'Cookie Crumble Corner'),

(20, 'Ice Cream Isle'),

(21, 'Peanut Butter Palace'),

(22, 'Honeycomb Haven'),

(23, 'Taffy Trail'),

(24, 'Cupcake Cove'),

(25, 'Gelato Grove'),

(26, 'Sherbet Street'),

(27, 'Bonbon Bay'),

(28, 'Cherry Chew Lane'),

(29, 'Praline Path'),

(30, 'Cinnamon Square'),

(31, 'Butterscotch Boulevard'),

(32, 'Minty Meadows'),

(33, 'Gingerbread Lane'),

(34, 'Jawbreaker Junction'),

(35, 'Nougat Nook'),

(36, 'Sundae Street'),

(37, 'Candyfloss Court'),

(38, 'Truffle Terrace'),

(39, 'Popcorn Plaza'),

(40, 'Caramel Apple Avenue'),

(41, 'Soda Pop Square'),

(42, 'Grape Gummy Grove'),

(43, 'Banana Bonanza'),

(44, 'Peachy Pie Place'),

(45, 'Blueberry Bliss Boulevard'),

(46, 'Strawberry Swirl Street'),

(47, 'Raspberry Ripple Road'),

(48, 'Mango Madness Manor'),

(49, 'Pineapple Paradise'),

(50, 'Sweet Tooth Delights');

insert into goods\_clients (id, goods\_id, clients\_id) values

insert into goods\_clients (id, goods\_id, clients\_id) values

(1, 18, 24), (2,19, 4),

(3, 9, 21), (4, 17, 50),

(5, 11, 9), (6, 31, 45),

(7, 2, 29), (8, 14, 3),

(9, 30, 44), (10, 5, 42),

(11, 3, 37), (12, 11, 11),

(13, 27, 45), (14, 6, 35),

(15, 29, 18), (16, 1, 18),

(17, 18, 1), (18, 29, 7),

(19, 24, 5), (20, 44, 2),

(21, 40, 23), (22, 36, 40),

(23, 15, 1), (24, 23, 28),

(25, 3, 44), (26, 31, 37),

(27, 1, 6), (28, 16, 33),

(29, 26, 36), (30, 10, 41),

(31, 33, 18), (32, 15, 46),

(33, 31, 14), (34, 49, 5),

(35, 29, 27), (36, 27, 19),

(37, 7, 1), (38, 7, 49),

(39, 44, 34), (40, 18, 13),

(41, 44, 18), (42, 19, 44),

(43, 44, 2), (44, 17, 39),

(45, 26, 27), (46, 3, 4),

(47, 14, 20), (48, 40, 36),

(49, 9, 22), (50, 19, 1),

(51, 12, 14), (52, 45, 14),

(53, 44, 35), (54, 38, 18),

(55, 5, 39), (56, 38, 11),

(57, 41, 12), (58, 22, 1),

(59, 43, 37), (60, 23, 22),

(61, 32, 18), (62, 45, 30),

(63, 25, 28), (64, 4, 41),

(65, 33, 25), (66, 36, 7),

(67, 2, 41), (68, 40, 40),

(69, 5, 1), (70, 25, 40),

(71, 38, 40), (72, 16, 17),

(73, 38, 24), (74, 43, 15),

(75, 25, 29), (76, 21, 15),

(77, 2, 15), (78, 10, 1),

(79, 28, 14), (80, 14, 42),

(81, 8, 26), (82, 44, 29),

(83, 16, 45), (84, 18, 43),

(85, 10, 31), (86, 10, 4),

(87, 20, 38), (88, 8, 25),

(89, 50, 5), (90, 30, 47),

(91, 2, 39), (92, 21, 8),

(93, 44, 15), (94, 28, 46),

(95, 40, 47), (96, 50, 25),

(97, 9, 15), (98, 46, 19),

(99, 46, 26), (100, 49, 48),

(101, 16, 12), (102, 38, 5),

(103, 45, 20), (104, 8, 29),

(105, 20, 13), (106, 10, 34),

(107, 14, 21), (108, 9, 10),

(109, 9, 23), (110, 3, 41),

(111, 41, 10), (112, 15, 41),

(113, 21, 15), (114, 38, 46),

(115, 35, 21), (116, 5, 29),

(117, 22, 20), (118, 40, 36),

(119, 20, 33), (120, 33, 2),

(121, 28, 4), (122, 32, 26),

(123, 30, 30), (124, 22, 3),

(125, 7, 16), (126, 20, 12),

(127, 36, 48), (128, 38, 30),

(129, 50, 30), (130, 35, 23),

(131, 19, 30), (132, 32, 21),

(133, 31, 30), (134, 22, 45),

(135, 48, 47), (136, 40, 15),

(137, 21, 40), (138, 23, 37),

(139, 9, 44), (140, 41, 32),

(141, 10, 44), (142, 1, 3),

(143, 42, 7), (144, 28, 23),

(145, 45, 32), (146, 50, 5),

(147, 21, 28), (148, 11, 43),

(149, 41, 12), (150, 3, 27),

(151, 25, 18), (152, 37, 29),

(153, 31, 40), (154, 35, 32),

(155, 6, 1), (156, 49, 29),

(157, 10, 48), (158, 3, 24),

(159, 5, 36), (160, 12, 49),

(161, 24, 11), (162, 3, 36),

(163, 43, 29), (164, 47, 2),

(165, 11, 4), (166, 8, 14),

(167, 33, 13), (168, 25, 47),

(169, 31, 48), (170, 27, 34),

(171, 1, 12), (172, 9, 33),

(173, 23, 22), (174, 40, 10),

(175, 16, 1), (176, 24, 16),

(177, 5, 4), (178, 44, 8),

(179, 18, 8), (180, 37, 30),

(181, 11, 46), (182, 43, 45),

(183, 22, 29), (184, 5, 47),

(185, 42, 1), (186, 18, 35),

(187, 31, 45), (188, 3, 13),

(189, 11, 33), (190, 12, 38),

(191, 16, 16), (192, 45, 38),

(193, 40, 45), (194, 38, 24),

(195, 10, 1), (196, 50, 27),

(197, 34, 41), (198, 32, 43),

(199, 12, 48), (200, 6, 46),

(201, 8, 15), (202, 13, 25),

(203, 30, 48), (204, 48, 6),

(205, 17, 29), (206, 50, 13),

(207, 34, 42), (208, 9, 29),

(209, 10, 1), (210, 47, 9),

(211, 38, 7), (212, 13, 23),

(213, 33, 32), (214, 49, 45),

(215, 35, 47), (216, 31, 13),

(217, 1, 21), (218, 30, 34),

(219, 36, 35), (220, 31, 35),

(221, 4, 27), (222, 18, 43),

(223, 17, 25), (224, 32, 13),

(225, 9, 21), (226, 15, 10),

(227, 42, 24), (228, 48, 35),

(229, 50, 2), (230, 27, 9),

(231, 3, 34), (232, 11, 1),

(233, 48, 6), (234, 3, 41),

(235, 7, 16), (236, 46, 10),

(237, 46, 4), (238, 22, 13),

(239, 1, 48), (240, 5, 7),

(241, 10, 3), (242, 19, 17),

(243, 47, 17), (244, 4, 36),

(245, 47, 41), (246, 43, 8),

(247, 46, 4), (248, 36, 16),

(249, 44, 6), (250, 15, 15),

(251, 31, 4), (252, 39, 33),

(253, 41, 16), (254, 30, 4),

(255, 21, 37), (256, 19, 27),

(257, 43, 40), (258, 4, 17),

(259, 39, 47), (260, 9, 9),

(261, 7, 50), (262, 36, 12),

(263, 33, 24), (264, 2, 22),

(265, 49, 15), (266, 47, 4),

(267, 15, 19), (268, 43, 22),

(269, 43, 33), (270, 10, 39),

(271, 18, 12), (272, 25, 25),

(273, 30, 34), (274, 50, 3),

(275, 40, 4), (276, 13, 29),

(277, 40, 21), (278, 28, 1),

(279, 30, 43), (280, 45, 28),

(281, 32, 40), (282, 2, 14),

(283, 45, 48), (284, 5, 22),

(285, 3, 8), (286, 6, 50),

(287, 31, 11), (288, 4, 22),

(289, 34, 25), (290, 25, 41),

(291, 24, 15), (292, 8, 25),

(293, 16, 47), (294, 27, 23),

(295, 18, 24), (296, 14, 35),

(297, 8, 29), (298, 25, 5),

(299, 31, 28), (300, 28, 15),

(301, 21, 36), (302, 45, 24),

(303, 17, 34), (304, 1, 35),

(305, 43, 27), (306, 27, 33),

(307, 42, 34), (308, 44, 33),

(309, 42, 11), (310, 49, 32),

(311, 36, 16), (312, 40, 50),

(313, 42, 32), (314, 39, 44),

(315, 30, 50), (316, 3, 6),

(317, 49, 37), (318, 11, 18),

(319, 32, 5), (320, 1, 45),

(321, 29, 50), (322, 11, 44),

(323, 50, 6), (324, 33, 37),

(325, 6, 16), (326, 48, 4),

(327, 18, 28), (328, 49, 15),

(329, 31, 9), (330, 47, 38),

(331, 39, 23), (332, 9, 34),

(333, 49, 12), (334, 31, 9),

(335, 1, 20), (336, 1, 11),

(337, 29, 5), (338, 13, 13),

(339, 48, 14), (340, 2, 12),

(341, 24, 24), (342, 42, 2),

(343, 11, 24), (344, 36, 42),

(345, 37, 10), (346, 43, 46),

(347, 28, 48), (348, 14, 45),

(349, 14, 39), (350, 35, 4),

(351, 44, 5), (352, 42, 8),

(353, 28, 48), (354, 35, 9),

(355, 32, 28), (356, 49, 44),

(357, 1, 30), (358, 48, 12),

(359, 38, 34), (360, 33, 8),

(361, 15, 47), (362, 49, 13),

(363, 45, 39), (364, 29, 3),

(365, 3, 45), (366, 18, 11),

(367, 38, 49), (368, 17, 4),

(369, 37, 13), (370, 12, 29),

(371, 12, 20), (372, 7, 14),

(373, 22, 45), (374, 1, 36),

(375, 49, 10), (376, 2, 35),

(377, 37, 26), (378, 4, 15),

(379, 6, 43), (380, 13, 11),

(381, 29, 42), (382, 37, 9),

(383, 13, 22), (384, 15, 23),

(385, 29, 28), (386, 10, 47),

(387, 18, 31), (388, 31, 26),

(389, 12, 3), (390, 20, 3),

(391, 5, 30), (392, 42, 42),

(393, 41, 11), (394, 15, 34),

(395, 9, 22), (396, 31, 45),

(397, 22, 4), (398, 33, 3),

(399, 35, 46), (400, 4, 2),

(401, 27, 18), (402, 1, 20),

(403, 7, 27), (404, 33, 32),

(405, 6, 43), (406, 37, 36),

(407, 33, 50), (408, 21, 17),

(409, 10, 31), (410, 50, 47),

(411, 3, 9), (412, 18, 45),

(413, 12, 21), (414, 46, 41),

(415, 8, 1), (416, 23, 45),

(417, 43, 22), (418, 4, 17),

(419, 19, 45), (420, 41, 32),

(421, 31, 15), (422, 49, 20),

(423, 25, 35), (424, 24, 17),

(425, 16, 17), (426, 23, 30),

(427, 11, 45), (428, 42, 43),

(429, 17, 4), (430, 16, 13),

(431, 6, 36), (432, 1, 38),

(433, 30, 6), (434, 30, 2),

(435, 15, 4), (436, 25, 45),

(437, 50, 29), (438, 34, 25),

(439, 47, 4), (440, 39, 3),

(441, 45, 38), (442, 49, 42),

(443, 36, 47), (444, 18, 12),

(445, 9, 5), (446, 22, 30),

(447, 15, 6), (448, 12, 10),

(449, 27, 28), (450, 43, 8),

(451, 26, 30), (452, 47, 32),

(453, 6, 13), (454, 37, 35),

(455, 14, 28), (456, 10, 48),

(457, 41, 1), (458, 10, 17),

(459, 38, 4), (460, 38, 2),

(461, 23, 37), (462, 48, 44),

(463, 40, 49), (464, 31, 43),

(465, 16, 17), (466, 11, 35),

(467, 2, 9), (468, 35, 34),

(469, 34, 33), (470, 49, 40),

(471, 49, 50), (472, 31, 4),

(473, 21, 11), (474, 20, 34),

(475, 1, 10), (476, 29, 14),

(477, 20, 1), (478, 8, 34),

(479, 34, 43), (480, 33, 36),

(481, 39, 20), (482, 7, 5),

(483, 18, 26), (484, 33, 28),

(485, 8, 37), (486, 25, 21),

(487, 3, 50), (488, 9, 12),

(489, 10, 44), (490, 36, 15),

(491, 15, 45), (492, 17, 16),

(493, 6, 3), (494, 34, 22),

(495, 44, 12), (496, 39, 48),

(497, 15, 17), (498, 3, 12),

(499, 43, 1), (500, 12, 36),

(501, 31, 10), (502, 43, 38),

(503, 14, 43), (504, 34, 41),

(505, 18, 40), (506, 12, 11),

(507, 36, 47), (508, 22, 17),

(509, 22, 27), (510, 37, 21),

(511, 20, 41), (512, 44, 9),

(513, 17, 18), (514, 8, 17),

(515, 41, 7), (516, 9, 37),

(517, 8, 32), (518, 35, 21),

(519, 9, 38), (520, 40, 6),

(521, 7, 30), (522, 47, 50),

(523, 46, 10), (524, 39, 37),

(525, 48, 14), (526, 18, 48),

(527, 34, 45), (528, 29, 31),

(529, 9, 21), (530, 37, 10),

(531, 10, 5), (532, 7, 15),

(533, 8, 41), (534, 50, 11),

(535, 43, 32), (536, 28, 49),

(537, 42, 35), (538, 14, 5),

(539, 29, 9), (540, 49, 15),

(541, 40, 30), (542, 24, 30),

(543, 50, 45), (544, 49, 3),

(545, 22, 5), (546, 12, 40),

(547, 45, 31), (548, 22, 4),

(549, 8, 40), (550, 2, 16),

(551, 47, 26), (552, 28, 5),

(553, 26, 1), (554, 16, 34),

(555, 33, 44), (556, 41, 9),

(557, 41, 28), (558, 4, 16),

(559, 28, 33), (560, 38, 29),

(561, 37, 17), (562, 5, 23),

(563, 30, 9), (564, 16, 50),

(565, 23, 37), (566, 1, 14),

(567, 34, 47), (568, 40, 17),

(569, 20, 23), (570, 34, 3),

(571, 33, 29), (572, 30, 21),

(573, 36, 31), (574, 15, 19),

(575, 47, 23), (576, 24, 1),

(577, 25, 8), (578, 1, 39),

(579, 9, 26), (580, 41, 14),

(581, 47, 32), (582, 48, 4),

(583, 40, 48), (584, 43, 2),

(585, 35, 47), (586, 12, 1),

(587, 34, 35), (588, 6, 18),

(589, 4, 9), (590, 25, 39),

(591, 41, 10), (592, 4, 5),

(593, 18, 25), (594, 3, 42),

(595, 13, 38), (596, 21, 27),

(597, 49, 7), (598, 42, 9),

(599, 41, 30), (600, 16, 29),

(601, 30, 31), (602, 48, 11),

(603, 37, 44), (604, 50, 50),

(605, 6, 36), (606, 42, 28),

(607, 45, 34), (608, 44, 39),

(609, 11, 44), (610, 9, 46),

(611, 4, 14), (612, 4, 36),

(613, 11, 40), (614, 36, 1),

(615, 36, 34), (616, 21, 41),

(617, 37, 49), (618, 6, 9),

(619, 12, 50), (620, 10, 16),

(621, 28, 17), (622, 17, 33),

(623, 40, 42), (624, 47, 11),

(625, 18, 48), (626, 9, 4),

(627, 4, 16), (628, 7, 33),

(629, 17, 24), (630, 39, 38),

(631, 47, 12), (632, 9, 28),

(633, 48, 35), (634, 31, 44),

(635, 12, 27), (636, 34, 20),

(637, 30, 4), (638, 3, 19),

(639, 19, 34), (640, 40, 50),

(641, 4, 33), (642, 4, 36),

(643, 11, 44), (644, 43, 26),

(645, 9, 45), (646, 3, 16),

(647, 28, 8), (648, 40, 29),

(649, 3, 3), (650, 2, 14),

(651, 5, 41), (652, 15, 4),

(653, 21, 28), (654, 10, 25),

(655, 18, 37), (656, 13, 20),

(657, 18, 24), (658, 17, 22),

(659, 48, 2), (660, 15, 48),

(661, 2, 32), (662, 24, 37),

(663, 8, 7), (664, 1, 6),

(665, 2, 22), (666, 50, 16),

(667, 12, 44), (668, 35, 9),

(669, 11, 29), (670, 49, 9),

(671, 38, 4), (672, 22, 31),

(673, 26, 7), (674, 9, 22),

(675, 24, 6), (676, 39, 46),

(677, 6, 11), (678, 21, 5),

(679, 26, 39), (680, 35, 32),

(681, 27, 47), (682, 39, 41),

(683, 12, 25), (684, 11, 25),

(685, 3, 1), (686, 23, 10),

(687, 47, 26), (688, 47, 47),

(689, 46, 5), (690, 15, 35),

(691, 7, 47), (692, 11, 37),

(693, 23, 44), (694, 40, 15),

(695, 9, 5), (696, 2, 42),

(697, 15, 29), (698, 13, 30),

(699, 28, 32), (700, 20, 18),

(701, 39, 3), (702, 9, 1),

(703, 31, 31), (704, 36, 23),

(705, 16, 41), (706, 2, 10),

(707, 32, 43), (708, 42, 27),

(709, 6, 28), (710, 11, 13),

(711, 1, 49), (712, 44, 40),

(713, 1, 20), (714, 5, 10),

(715, 16, 50), (716, 37, 19),

(717, 5, 48), (718, 32, 7),

(719, 6, 4), (720, 24, 45),

(721, 44, 37), (722, 38, 10),

(723, 36, 40), (724, 49, 42),

(725, 38, 14), (726, 25, 21),

(727, 5, 7), (728, 21, 30),

(729, 13, 11), (730, 44, 44),

(731, 30, 27), (732, 4, 37),

(733, 47, 42), (734, 27, 48),

(735, 34, 17), (736, 46, 32),

(737, 26, 35), (738, 37, 15),

(739, 15, 39), (740, 22, 34),

(741, 1, 46), (742, 2, 17),

(743, 37, 27), (744, 28, 1),

(745, 18, 32), (746, 2, 15),

(747, 39, 31), (748, 18, 49),

(749, 45, 8), (750, 25, 12),

(751, 50, 6), (752, 25, 30),

(753, 28, 27), (754, 11, 23),

(755, 41, 44), (756, 18, 21),

(757, 9, 37), (758, 36, 46),

(759, 49, 4), (760, 21, 44),

(761, 49, 42), (762, 46, 2),

(763, 28, 19), (764, 22, 17),

(765, 8, 2), (766, 4, 38),

(767, 44, 15), (768, 32, 40),

(769, 6, 46), (770, 31, 24),

(771, 39, 48), (772, 2, 38),

(773, 37, 21), (774, 18, 1),

(775, 8, 32), (776, 2, 46),

(777, 8, 35), (778, 28, 4),

(779, 33, 46), (780, 44, 11),

(781, 20, 41), (782, 12, 34),

(783, 13, 37), (784, 26, 25),

(785, 26, 29), (786, 36, 48),

(787, 16, 1), (788, 27, 29),

(789, 18, 3), (790, 41, 26),

(791, 32, 18), (792, 44, 47),

(793, 11, 16), (794, 33, 33),

(795, 35, 16), (796, 45, 8),

(797, 11, 16), (798, 42, 49),

(799, 20, 18), (800, 34, 23),

(801, 44, 49), (802, 38, 25),

(803, 35, 10), (804, 3, 35),

(805, 27, 28), (806, 5, 17),

(807, 9, 28), (808, 23, 8),

(809, 21, 31), (810, 9, 40),

(811, 6, 22), (812, 48, 20),

(813, 48, 49), (814, 23, 21),

(815, 13, 23), (816, 47, 19),

(817, 50, 38), (818, 37, 18),

(819, 48, 9), (820, 3, 22),

(821, 15, 40), (822, 50, 40),

(823, 41, 27), (824, 27, 31),

(825, 22, 31), (826, 6, 35),

(827, 19, 17), (828, 42, 25),

(829, 15, 31), (830, 31, 37),

(831, 10, 11), (832, 33, 10),

(833, 23, 10), (834, 23, 10),

(835, 12, 23), (836, 33, 16),

(837, 19, 8), (838, 8, 39),

(839, 21, 38), (840, 28, 1),

(841, 12, 14), (842, 21, 24),

(843, 27, 1), (844, 26, 1),

(845, 44, 46), (846, 29, 18),

(847, 7, 43), (848, 21, 27),

(849, 41, 49), (850, 22, 7),

(851, 38, 46), (852, 27, 6),

(853, 48, 12), (854, 1, 48),

(855, 43, 5), (856, 28, 35),

(857, 26, 8), (858, 35, 20),

(859, 7, 13), (860, 19, 50),

(861, 48, 1), (862, 12, 1),

(863, 17, 31), (864, 12, 38),

(865, 20, 8), (866, 12, 39),

(867, 6, 29), (868, 6, 28),

(869, 26, 46), (870, 35, 32),

(871, 2, 30), (872, 39, 12),

(873, 5, 37), (874, 29, 11),

(875, 29, 34), (876, 31, 15),

(877, 14, 30), (878, 16, 30),

(879, 21, 39), (880, 30, 31),

(881, 9, 23), (882, 48, 42),

(883, 17, 7), (884, 38, 16),

(885, 44, 14), (886, 37, 30),

(887, 48, 38), (888, 35, 2),

(889, 9, 2), (890, 41, 49),

(891, 30, 46), (892, 41, 2),

(893, 30, 24), (894, 43, 19),

(895, 23, 47), (896, 48, 8),

(897, 36, 29), (898, 27, 19),

(899, 11, 30), (900, 40, 50),

(901, 18, 34), (902, 48, 27),

(903, 2, 7), (904, 17, 19),

(905, 33, 22), (906, 50, 4),

(907, 25, 41), (908, 31, 43),

(909, 37, 46), (910, 42, 7),

(911, 2, 4), (912, 46, 43),

(913, 39, 4), (914, 26, 33),

(915, 21, 18), (916, 29, 9),

(917, 1, 12), (918, 10, 5),

(919, 3, 9), (920, 40, 21),

(921, 10, 24), (922, 30, 31),

(923, 41, 23), (924, 39, 10),

(925, 20, 35), (926, 32, 31),

(927, 36, 9), (928, 30, 31),

(929, 33, 5), (930, 38, 24),

(931, 48, 35), (932, 8, 44),

(933, 6, 3), (934, 9, 41),

(935, 13, 11), (936, 12, 10),

(937, 25, 22), (938, 9, 6),

(939, 21, 42), (940, 47, 4),

(941, 48, 23), (942, 28, 33),

(943, 4, 4), (944, 41, 22),

(945, 34, 45), (946, 36, 15),

(947, 35, 16), (948, 17, 38),

(949, 40, 4), (950, 27, 16),

(951, 49, 35), (952, 4, 41),

(953, 10, 12), (954, 20, 48),

(955, 24, 48), (956, 13, 39),

(957, 31, 15), (958, 2, 7),

(959, 46, 15), (960, 15, 1),

(961, 36, 27), (962, 27, 44),

(963, 24, 18), (964, 18, 14),

(965, 1, 32), (966, 9, 8),

(967, 1, 49), (968, 27, 27),

(969, 17, 50), (970, 6, 12),

(971, 29, 50), (972, 34, 21),

(973, 7, 28), (974, 27, 19),

(975, 4, 30), (976, 19, 36),

(977, 14, 15), (978, 41, 49),

(979, 41, 13), (980, 25, 47),

(981, 9, 10), (982, 11, 25),

(983, 23, 44), (984, 15, 10),

(985, 32, 14), (986, 47, 28),

(987, 16, 39), (988, 10, 31),

(989, 3, 14), (990, 22, 41),

(991, 36, 9), (992, 2, 47),

(993, 16, 34), (994, 41, 37),

(995, 13, 2), (996, 37, 13),

(997, 33, 46), (998, 31, 14),

(999, 43, 46), (1000, 6, 9);

insert into viewed

(id, datetime, goods\_clients\_id) values (1, '2023-07-02 18:44:29', 61),

(2, '2020-03-05 11:13:49', 828),

(3, '2020-07-19 15:13:47', 698),

(4, '2024-03-06 07:09:48', 874),

(5, '2021-06-03 16:21:16', 665),

(6, '2020-10-26 13:24:39', 530),

(7, '2021-12-09 21:04:53', 877),

(8, '2023-02-20 02:05:52', 932),

(9, '2023-04-21 04:21:35', 536),

(10, '2024-01-25 08:52:17', 992),

(11, '2022-08-29 18:58:19', 511),

(12, '2023-07-02 15:24:01', 513),

(13, '2021-04-07 12:43:58', 173),

(14, '2022-06-25 10:33:56', 677),

(15, '2021-05-31 08:28:08', 958),

(16, '2022-06-29 10:28:05', 225),

(17, '2023-03-20 09:53:08', 782),

(18, '2023-05-14 02:06:10', 230),

(19, '2022-11-21 04:59:07', 371),

(20, '2022-08-21 00:58:02', 795),

(21, '2023-01-30 16:27:32', 600),

(22, '2023-05-28 20:42:35', 290),

(23, '2021-03-21 05:12:48', 957),

(24, '2020-04-25 07:14:07', 554),

(25, '2020-09-01 11:41:37', 120),

(26, '2021-07-06 04:02:16', 260),

(27, '2021-10-28 15:13:08', 166),

(28, '2020-04-01 02:08:02', 799),

(29, '2020-07-19 09:28:49', 797),

(30, '2020-02-17 11:25:42', 702),

(31, '2021-08-10 12:19:23', 168),

(32, '2022-05-01 02:26:21', 408),

(33, '2022-05-08 06:07:42', 213),

(34, '2023-04-10 12:49:14', 864),

(35, '2023-05-02 11:14:50', 147),

(36, '2022-07-22 16:10:43', 881),

(37, '2022-06-25 23:51:13', 773),

(38, '2021-06-26 09:05:10', 253),

(39, '2020-08-06 02:43:27', 705),

(40, '2022-02-04 02:30:40', 327),

(41, '2022-09-08 00:23:49', 913),

(42, '2022-12-25 15:04:45', 31),

(43, '2021-12-13 02:45:37', 702),

(44, '2022-10-10 11:53:11', 638),

(45, '2023-02-26 23:20:48', 300),

(46, '2022-02-10 10:16:53', 237),

(47, '2023-01-29 08:35:58', 801),

(48, '2020-04-30 14:48:31', 65),

(49, '2023-04-24 06:22:28', 190),

(50, '2022-09-06 03:13:56', 1000),

(51, '2023-12-30 21:22:23', 677),

(52, '2020-12-16 13:07:24', 82),

(53, '2020-12-25 23:52:41', 571),

(54, '2021-08-01 06:43:34', 484),

(55, '2021-01-07 17:14:32', 981),

(56, '2023-08-04 21:29:17', 451),

(57, '2023-03-12 08:19:14', 712),

(58, '2022-11-26 02:11:16', 194),

(59, '2020-12-27 14:32:55', 261),

(60, '2023-01-27 23:23:34', 484),

(61, '2022-08-10 06:18:24', 594),

(62, '2023-09-30 08:22:38', 160),

(63, '2021-06-13 13:17:10', 917),

(64, '2020-05-29 20:39:53', 833),

(65, '2020-09-22 06:54:36', 598),

(66, '2023-12-02 04:40:02', 56),

(67, '2024-01-05 00:06:00', 274),

(68, '2023-09-20 08:51:18', 564),

(69, '2024-02-05 14:36:12', 227),

(70, '2022-05-08 01:56:49', 655),

(71, '2022-01-30 20:31:01', 475),

(72, '2022-02-11 18:24:04', 402),

(73, '2022-07-05 10:04:05', 247),

(74, '2022-02-02 22:02:39', 370),

(75, '2021-10-13 17:42:00', 371),

(76, '2020-12-05 11:24:35', 584),

(77, '2023-06-13 11:26:13', 351),

(78, '2023-09-13 07:11:10', 468),

(79, '2022-07-22 18:39:24', 139),

(80, '2020-03-10 06:49:00', 535),

(81, '2021-04-20 18:15:15', 803),

(82, '2022-01-19 17:32:18', 16),

(83, '2023-12-13 20:22:24', 731),

(84, '2020-04-24 03:24:47', 732),

(85, '2020-09-20 22:34:14', 932),

(86, '2022-08-16 18:14:38', 664),

(87, '2020-08-04 10:19:07', 310),

(88, '2023-04-30 10:31:42', 656),

(89, '2022-05-11 12:35:55', 919),

(90, '2020-10-09 12:57:54', 930),

(91, '2020-03-20 18:38:11', 53),

(92, '2020-09-02 05:26:13', 263),

(93, '2022-12-21 02:31:59', 663),

(94, '2021-11-08 03:05:20', 981),

(95, '2023-12-28 09:40:10', 752),

(96, '2023-02-03 02:03:39', 812),

(97, '2022-12-22 09:31:18', 982),

(98, '2023-07-06 03:25:33', 817),

(99, '2024-01-23 22:55:43', 412),

(100, '2023-09-20 16:32:17', 15),

(101, '2021-05-13 17:53:53', 146),

(102, '2024-03-24 21:24:30', 151),

(103, '2022-01-06 18:36:04', 113),

(104, '2023-11-17 19:34:47', 386),

(105, '2024-03-05 15:32:36', 9),

(106, '2021-11-23 05:56:27', 641),

(107, '2021-09-10 19:02:46', 807),

(108, '2020-10-06 20:30:50', 92),

(109, '2022-03-18 01:26:24', 231),

(110, '2022-07-09 11:58:22', 141),

(111, '2021-05-28 09:14:40', 905),

(112, '2022-05-23 10:17:39', 310),

(113, '2023-06-02 09:10:26', 64),

(114, '2022-08-31 18:36:52', 812),

(115, '2023-06-03 09:47:03', 639),

(116, '2024-03-25 00:52:29', 283),

(117, '2022-02-27 04:05:42', 86),

(118, '2021-06-13 17:28:25', 36),

(119, '2020-04-09 11:54:53', 706),

(120, '2023-03-21 07:24:28', 996),

(121, '2023-08-25 21:11:39', 704),

(122, '2021-05-15 12:32:15', 70),

(123, '2023-07-23 03:28:32', 608),

(124, '2023-05-21 02:29:02', 466),

(125, '2022-11-01 20:51:39', 684),

(126, '2023-04-09 12:38:29', 778),

(127, '2023-09-20 13:46:28', 697),

(128, '2022-06-28 08:57:09', 397),

(129, '2021-03-27 12:40:03', 440),

(130, '2023-03-21 04:13:21', 694),

(131, '2024-02-14 02:33:29', 250),

(132, '2022-12-27 05:32:22', 935),

(133, '2021-08-20 11:59:10', 759),

(134, '2021-11-21 00:14:46', 804),

(135, '2023-08-10 14:16:14', 780),

(136, '2022-08-17 01:35:57', 763),

(137, '2020-04-06 18:44:47', 610),

(138, '2021-04-06 16:31:08', 929),

(139, '2020-07-06 16:45:59', 962),

(140, '2023-10-05 22:27:02', 175),

(141, '2021-01-22 22:24:07', 494),

(142, '2024-03-17 18:01:53', 13),

(143, '2023-05-08 06:05:13', 965),

(144, '2023-07-17 20:30:11', 1),

(145, '2024-02-18 08:51:10', 564),

(146, '2020-06-02 11:10:51', 314),

(147, '2022-05-24 11:32:11', 48),

(148, '2020-07-28 14:23:12', 891),

(149, '2022-02-07 11:06:07', 259),

(150, '2020-10-27 19:30:14', 17),

(151, '2023-09-16 10:02:17', 452),

(152, '2023-07-26 16:43:29', 372),

(153, '2023-09-24 20:14:41', 549),

(154, '2020-10-08 10:45:07', 998),

(155, '2022-12-11 20:21:10', 274),

(156, '2022-11-21 11:28:09', 746),

(157, '2023-07-08 08:12:04', 253),

(158, '2021-04-20 04:55:11', 906),

(159, '2023-03-11 05:32:15', 266),

(160, '2023-11-24 23:50:36', 708),

(161, '2023-02-09 09:40:51', 438),

(162, '2023-08-21 00:34:41', 169),

(163, '2021-05-05 18:21:07', 230),

(164, '2023-03-26 00:31:37', 508),

(165, '2021-09-02 14:00:24', 571),

(166, '2023-11-05 06:09:05', 498),

(167, '2023-12-31 22:17:24', 883),

(168, '2024-03-18 23:17:29', 805),

(169, '2022-08-12 06:01:46', 835),

(170, '2021-06-01 04:05:48', 42),

(171, '2022-02-18 21:11:34', 111),

(172, '2022-06-17 22:40:18', 361),

(173, '2023-02-17 13:03:12', 834),

(174, '2024-03-29 13:01:08', 622),

(175, '2021-09-12 05:26:32', 262),

(176, '2023-06-08 16:00:54', 460),

(177, '2021-10-18 14:54:55', 440),

(178, '2022-08-23 16:41:11', 793),

(179, '2020-03-31 19:32:49', 986),

(180, '2023-02-26 04:15:38', 47),

(181, '2023-07-11 09:35:38', 323),

(182, '2022-11-14 23:59:35', 606),

(183, '2023-02-17 20:11:43', 97),

(184, '2023-04-24 14:26:09', 871),

(185, '2020-01-18 17:50:35', 328),

(186, '2023-02-06 10:59:30', 673),

(187, '2020-09-30 22:12:18', 347),

(188, '2021-03-03 09:16:53', 27),

(189, '2020-02-27 09:19:39', 345),

(190, '2020-06-25 18:28:07', 500),

(191, '2022-12-24 09:29:45', 648),

(192, '2023-08-28 05:05:15', 359),

(193, '2023-04-15 09:04:48', 54),

(194, '2024-03-04 15:29:40', 273),

(195, '2021-10-31 06:34:45', 785),

(196, '2022-10-06 20:02:40', 563),

(197, '2023-12-22 22:49:06', 889),

(198, '2022-10-28 21:42:08', 63),

(199, '2020-10-26 17:09:42', 324),

(200, '2021-11-15 03:32:16', 518);

insert into bought

(id, datetime, quantity, goods\_clients\_id)

values

(1, '2022-03-18 18:38:34', 9, 880),

(2, '2023-10-08 23:12:09', 4, 130),

(3, '2020-08-09 17:39:00', 10, 361),

(4, '2021-10-21 06:56:34', 8, 70),

(5, '2021-06-01 09:47:47', 9, 313),

(6, '2020-08-28 11:19:03', 6, 330),

(7, '2023-01-04 21:01:56', 7, 778),

(8, '2022-08-21 18:28:26', 9, 419),

(9, '2022-12-13 15:23:35', 2, 411),

(10, '2023-12-05 02:06:06', 3, 587),

(11, '2022-08-25 16:58:11', 7, 568),

(12, '2021-07-18 21:09:39', 8, 545),

(13, '2023-02-12 19:12:09', 3, 668),

(14, '2020-05-08 13:03:50', 9, 85),

(15, '2020-07-04 22:17:27', 9, 785),

(16, '2022-01-14 23:14:59', 10, 513),

(17, '2024-02-23 19:32:10', 6, 297),

(18, '2023-06-05 05:14:59', 4, 791),

(19, '2023-06-05 12:38:59', 6, 271),

(20, '2023-04-13 19:48:10', 6, 218),

(21, '2021-04-17 17:40:29', 6, 875),

(22, '2020-02-09 20:12:34', 2, 168),

(23, '2023-06-03 03:42:22', 1, 353),

(24, '2022-10-22 22:00:39', 5, 249),

(25, '2020-02-25 16:15:21', 3, 914),

(26, '2020-03-17 06:10:17', 10, 620),

(27, '2022-05-18 23:34:49', 4, 918),

(28, '2021-05-18 13:55:14', 6, 178),

(29, '2022-02-18 18:50:09', 3, 552),

(30, '2021-11-16 01:41:35', 4, 306),

(31, '2021-10-10 14:54:37', 1, 117),

(32, '2022-02-04 22:13:29', 9, 483),

(33, '2022-08-24 16:56:01', 8, 883),

(34, '2020-01-07 23:57:02', 3, 935),

(35, '2024-03-27 01:46:03', 10, 266),

(36, '2021-03-30 03:58:51', 9, 960),

(37, '2023-12-21 20:08:02', 1, 659),

(38, '2022-04-16 19:02:08', 4, 149),

(39, '2023-07-26 00:44:10', 5, 704),

(40, '2020-08-31 05:09:38', 2, 637),

(41, '2023-04-13 03:15:51', 3, 851),

(42, '2024-03-01 08:08:16', 2, 73),

(43, '2023-07-03 01:22:57', 4, 104),

(44, '2022-03-22 11:37:49', 10, 210),

(45, '2021-02-04 17:09:13', 5, 153),

(46, '2021-04-29 06:46:38', 5, 403),

(47, '2021-12-27 02:24:05', 4, 814),

(48, '2022-09-30 16:42:07', 10, 719),

(49, '2021-02-24 02:14:58', 6, 609),

(50, '2020-01-14 08:05:58', 7, 93),

(51, '2020-04-02 13:25:40', 4, 777),

(52, '2021-07-21 13:50:39', 5, 912),

(53, '2020-11-03 19:40:06', 9, 79),

(54, '2022-05-06 19:56:29', 9, 501),

(55, '2023-06-09 21:34:08', 2, 750),

(56, '2021-06-12 15:48:26', 8, 227),

(57, '2024-02-18 12:17:15', 6, 397),

(58, '2021-02-22 01:59:16', 10, 928),

(59, '2020-11-15 08:00:55', 1, 471),

(60, '2021-05-29 04:17:34', 1, 469),

(61, '2023-06-20 13:04:01', 6, 286),

(62, '2023-11-24 01:29:34', 8, 960),

(63, '2023-08-09 14:30:16', 8, 270),

(64, '2023-10-14 18:21:49', 3, 378),

(65, '2021-04-24 11:06:49', 10, 462),

(66, '2021-04-19 00:09:13', 4, 472),

(67, '2021-09-30 19:02:52', 4, 584),

(68, '2023-10-11 17:58:55', 7, 32),

(69, '2020-03-12 03:27:40', 3, 633),

(70, '2023-05-03 09:02:43', 8, 409),

(71, '2022-08-07 17:12:09', 3, 139),

(72, '2022-01-06 19:14:39', 3, 640),

(73, '2021-07-28 03:49:15', 4, 303),

(74, '2020-09-18 03:15:26', 10, 222),

(75, '2022-10-21 04:05:49', 3, 754),

(76, '2021-04-17 20:51:04', 4, 153),

(77, '2023-12-15 21:53:47', 4, 533),

(78, '2021-06-28 19:12:11', 10, 741),

(79, '2023-06-02 10:10:28', 10, 192),

(80, '2020-09-06 12:09:16', 10, 783),

(81, '2023-03-16 05:52:06', 2, 945),

(82, '2023-11-05 00:39:41', 6, 116),

(83, '2023-06-25 12:20:17', 2, 1),

(84, '2024-02-25 08:36:44', 7, 573),

(85, '2024-02-02 13:28:52', 8, 412),

(86, '2022-02-03 13:01:34', 3, 611),

(87, '2021-03-28 04:28:16', 1, 625),

(88, '2022-05-04 03:19:10', 3, 749),

(89, '2020-07-22 22:03:12', 7, 749),

(90, '2020-04-10 13:17:43', 6, 518),

(91, '2022-06-07 22:25:07', 7, 322),

(92, '2022-01-22 14:08:15', 3, 723),

(93, '2021-11-07 13:11:44', 1, 354),

(94, '2020-08-06 02:08:19', 10, 602),

(95, '2022-09-29 17:31:51', 9, 456),

(96, '2021-08-20 18:24:01', 2, 230),

(97, '2021-10-31 06:28:33', 3, 709),

(98, '2023-07-13 17:07:52', 8, 211),

(99, '2023-12-19 08:53:59', 4, 913),

(100, '2020-07-26 07:16:00', 7, 272),

(101, '2023-07-11 02:46:38', 7, 902),

(102, '2021-01-26 19:55:15', 8, 469),

(103, '2022-09-27 22:24:31', 3, 927),

(104, '2023-02-21 19:26:04', 10, 43),

(105, '2023-06-08 00:05:40', 1, 516),

(106, '2021-05-27 15:46:04', 3, 974),

(107, '2022-06-30 21:01:45', 8, 208),

(108, '2021-06-10 07:24:36', 10, 108),

(109, '2022-05-01 22:27:07', 6, 683),

(110, '2023-02-18 17:42:54', 6, 573),

(111, '2022-01-24 01:41:31', 1, 670),

(112, '2024-03-19 22:19:48', 8, 792),

(113, '2020-07-03 06:39:48', 1, 825),

(114, '2023-08-21 17:42:59', 7, 769),

(115, '2024-01-15 00:49:34', 9, 227),

(116, '2022-02-22 13:40:19', 9, 116),

(117, '2023-07-26 00:57:07', 3, 415),

(118, '2022-11-02 04:41:50', 10, 241),

(119, '2020-09-05 18:11:44', 10, 321),

(120, '2021-12-21 18:48:22', 6, 548),

(121, '2021-12-28 08:18:53', 7, 837),

(122, '2022-06-03 08:38:11', 10, 877),

(123, '2021-06-22 16:01:57', 10, 607),

(124, '2021-10-19 10:42:50', 10, 379),

(125, '2023-09-30 12:49:03', 4, 951),

(126, '2023-06-16 04:43:14', 1, 483),

(127, '2021-09-02 11:15:37', 6, 705),

(128, '2023-10-05 14:34:51', 1, 16),

(129, '2022-05-10 16:29:20', 3, 529),

(130, '2022-06-25 22:14:13', 7, 900),

(131, '2020-04-07 21:35:14', 2, 593),

(132, '2023-09-06 13:58:20', 2, 178),

(133, '2024-01-07 14:14:32', 9, 422),

(134, '2024-01-28 09:00:37', 10, 913),

(135, '2023-08-06 16:42:56', 10, 703),

(136, '2020-01-24 18:08:00', 10, 552),

(137, '2022-12-25 21:24:39', 4, 648),

(138, '2022-04-07 15:35:22', 7, 794),

(139, '2021-07-13 08:21:20', 4, 32),

(140, '2022-08-25 10:23:41', 7, 664),

(141, '2023-05-30 12:24:33', 4, 448),

(142, '2022-01-27 16:02:20', 6, 955),

(143, '2021-08-05 04:15:29', 10, 539),

(144, '2020-08-29 18:33:28', 3, 526),

(145, '2020-05-03 22:08:50', 1, 281),

(146, '2022-07-29 03:43:42', 6, 464),

(147, '2022-09-10 02:03:21', 7, 174),

(148, '2023-09-30 10:15:47', 3, 508),

(149, '2021-06-13 03:14:34', 4, 533),

(150, '2022-05-24 13:45:12', 9, 602);

insert into ratings (id, datetime, rating, goods\_clients\_id) values (1, '2020-01-19 07:05:43', 1, 314),

(2, '2022-01-11 00:53:00', 3, 82),

(3, '2022-11-16 11:17:32', 3, 351),

(4, '2021-05-09 11:34:35', 5, 577),

(5, '2020-01-08 22:26:35', 2, 728),

(6, '2023-07-28 17:32:55', 4, 528),

(7, '2022-06-12 08:25:16', 1, 642),

(8, '2023-03-10 10:03:22', 5, 18),

(9, '2021-12-18 11:54:22', 5, 503),

(10, '2023-03-22 00:04:58', 1, 179),

(11, '2022-03-27 09:46:29', 2, 921),

(12, '2021-06-09 21:25:19', 1, 995),

(13, '2023-05-02 02:47:30', 4, 706),

(14, '2021-08-31 06:22:07', 5, 430),

(15, '2022-03-10 18:21:00', 3, 437),

(16, '2021-03-26 06:51:15', 3, 606),

(17, '2021-01-03 12:48:49', 3, 454),

(18, '2021-07-19 01:11:21', 4, 165),

(19, '2020-03-14 05:49:53', 3, 168),

(20, '2020-05-14 15:56:27', 5, 486),

(21, '2024-02-15 11:22:25', 1, 94),

(22, '2022-06-06 15:39:35', 3, 185),

(23, '2023-04-13 03:52:15', 5, 597),

(24, '2020-01-06 08:44:27', 1, 561),

(25, '2022-05-16 16:36:41', 1, 258),

(26, '2023-04-27 12:02:30', 3, 666),

(27, '2020-11-18 17:55:47', 5, 610),

(28, '2022-09-18 12:26:24', 2, 333),

(29, '2020-04-11 14:10:44', 1, 908),

(30, '2020-05-06 08:34:10', 5, 219),

(31, '2021-06-13 05:14:12', 1, 485),

(32, '2022-06-11 20:11:28', 1, 827),

(33, '2023-11-10 23:39:16', 5, 907),

(34, '2020-09-01 18:35:12', 1, 567),

(35, '2020-02-28 08:04:02', 2, 610),

(36, '2023-02-15 08:45:47', 3, 114),

(37, '2022-08-20 00:56:51', 2, 610),

(38, '2020-04-11 09:37:03', 2, 164),

(39, '2021-08-28 02:15:52', 4, 771),

(40, '2020-04-13 07:43:39', 3, 880),

(41, '2022-04-04 03:49:28', 2, 979),

(42, '2023-10-28 05:46:00', 4, 826),

(43, '2023-11-05 13:09:34', 2, 63),

(44, '2021-04-22 06:02:32', 4, 730),

(45, '2022-12-16 22:53:52', 5, 860),

(46, '2020-05-30 05:26:20', 4, 232),

(47, '2021-09-24 19:54:40', 5, 447),

(48, '2021-08-02 11:42:40', 5, 233),

(49, '2022-11-10 03:59:50', 1, 232),

(50, '2022-09-10 00:08:19', 3, 658),

(51, '2021-02-01 16:07:32', 2, 37),

(52, '2022-12-19 20:52:09', 4, 840),

(53, '2021-04-23 16:39:46', 4, 944),

(54, '2021-10-31 14:47:16', 5, 222),

(55, '2020-12-07 23:42:48', 2, 109),

(56, '2023-12-31 16:03:20', 3, 368),

(57, '2024-03-06 17:38:03', 2, 383),

(58, '2021-01-20 01:20:34', 2, 250),

(59, '2022-09-20 20:48:54', 4, 837),

(60, '2020-06-02 19:45:54', 4, 35),

(61, '2020-04-22 02:09:01', 1, 288),

(62, '2022-02-02 23:49:09', 4, 909),

(63, '2020-12-26 06:41:58', 5, 753),

(64, '2021-07-27 00:52:10', 1, 756),

(65, '2022-01-06 23:30:38', 5, 93),

(66, '2022-02-24 06:07:11', 4, 702),

(67, '2021-02-26 03:27:54', 2, 452),

(68, '2022-03-01 01:24:43', 3, 597),

(69, '2024-02-22 12:32:32', 4, 6),

(70, '2021-12-20 03:48:47', 1, 226),

(71, '2022-11-05 11:19:58', 2, 89),

(72, '2020-09-06 19:31:38', 2, 842),

(73, '2020-04-10 05:28:47', 5, 11),

(74, '2021-08-30 06:20:05', 4, 109),

(75, '2020-08-03 18:18:29', 4, 420),

(76, '2023-02-15 17:18:09', 5, 128),

(77, '2021-10-03 23:04:13', 5, 573),

(78, '2020-03-07 06:23:36', 3, 808),

(79, '2022-07-26 03:19:14', 2, 227),

(80, '2020-02-17 16:29:47', 2, 42),

(81, '2020-09-01 02:50:26', 4, 647),

(82, '2021-04-23 23:10:00', 2, 124),

(83, '2023-12-25 22:55:38', 2, 281),

(84, '2020-03-13 05:11:08', 5, 793),

(85, '2023-02-28 19:50:27', 5, 372),

(86, '2023-02-03 18:25:27', 4, 236),

(87, '2020-02-20 15:31:02', 4, 100),

(88, '2020-07-29 04:43:24', 4, 119),

(89, '2023-11-24 16:23:52', 5, 752),

(90, '2023-03-17 18:08:08', 4, 779),

(91, '2022-01-15 10:43:40', 3, 993),

(92, '2023-11-04 02:22:23', 1, 372),

(93, '2022-10-19 10:26:10', 5, 824),

(94, '2022-12-31 11:25:12', 3, 812),

(95, '2021-10-13 13:44:13', 5, 908),

(96, '2023-07-18 03:12:33', 5, 318),

(97, '2020-10-27 09:53:41', 2, 429),

(98, '2020-12-30 10:35:37', 2, 794),

(99, '2021-04-21 16:25:45', 4, 53),

(100, '2022-09-23 23:56:40', 3, 484);

**Задача 2. Напишете заявка, в която демонстрирате SELECT с ограничаващо условие по избор** - ще изведем информация за всички клиенти и стоките, които са купили от определен доставчик:

SELECT clients.name AS ClientName, goods.name AS GoodName, bought.datetime AS DateOfPurchase FROM clients

JOIN goods\_clients ON clients.id = goods\_clients.clients\_id

JOIN goods ON goods\_clients.goods\_id = goods.id

JOIN suppliers ON goods.suppliers\_id = suppliers.id

JOIN bought ON goods\_clients.id = bought.goods\_clients\_id

WHERE suppliers.name = 'Lollipop Land Industries'

ORDER BY ClientName, Goodname, DateOfPurchase;

**Задача 3. Напишете заявка, в която използвате агрегатна функция и GROUP BY по ваш избор** – ще изведем броя промени на цените на стоките подредени по броя промени и името на стоката:

SELECT goods.name AS GoodName, COUNT(changes.id) AS ChangesCount FROM goods

JOIN changes ON goods.id = changes.goods\_id

GROUP BY GoodName

ORDER BY ChangesCount DESC, GoodName;

**Задача 4. Напишете заявка, в която демонстрирате INNER JOIN по ваш избор** – ще напишем заявка, с която ще изведем клиентите, които са видели и купили една и съща стока:

SELECT clients.name AS ClientName , goods.name AS GoodName FROM clients

JOIN goods\_clients ON clients.id = goods\_clients.clients\_id

JOIN goods ON goods\_clients.goods\_id = goods.id

INNER JOIN bought ON goods\_clients.id = bought.id

INNER JOIN viewed ON goods\_clients.id = viewed.id

ORDER BY ClientName, GoodName;

**Задача 5. Напишете заявка, в която демонстрирате OUTER JOIN по ваш избор** – ще напишем заявка, с която ще изведем цялата информация за всички стоки, които са и купени и видяни:

SELECT clients.name AS ClientName, goods.name AS GoodName, bought.datetime AS DateOfBuy, viewed.datetime AS DateOfView FROM clients

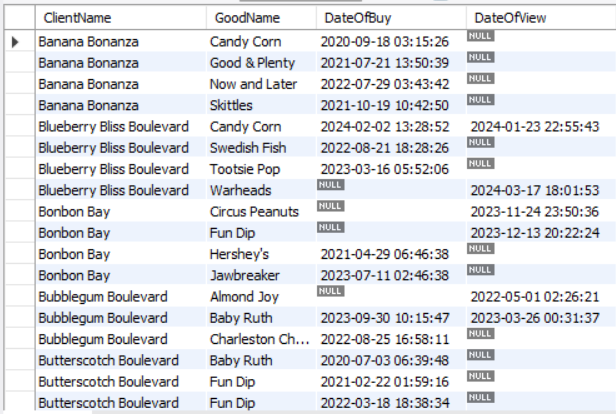
JOIN goods\_clients ON clients.id = goods\_clients.clients\_id

JOIN goods ON goods\_clients.goods\_id = goods.id

LEFT OUTER JOIN viewed ON goods\_clients.id = viewed.goods\_clients\_id

LEFT OUTER JOIN bought ON goods\_clients.id = bought.goods\_clients\_id

WHERE bought.datetime IS NOT NULL OR viewed.datetime IS NOT NULL

ORD

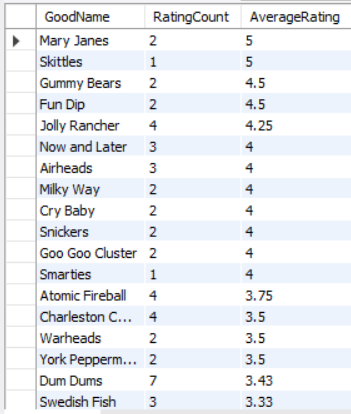
**Задача 6. Напишете заявка, в която демонстрирате вложен SELECT по ваш избор** – ще създадем заявка, с която ще изведем всички стоки предлагани от определен доставчик:

SELECT goods.name AS GoodName FROM goods

WHERE suppliers\_id IN(SELECT id FROM suppliers WHERE suppliers.name = 'Toffee Time Enterprises');

**Задача 7. Напишете заявка, в която демонстрирате едновременно JOIN и агрегатна функция** – ще изведем имената на стоките, броя на рейтингите за определена стока и средно аритметично на тези рейтинги, като ще ги групираме по името на стоката и ще ги подредим по рейтинг и техния брой:

SELECT goods.name AS GoodName, COUNT(ratings.id) AS RatingCount, ROUND(AVG(ratings.rating), 2) AS AverageRating FROM goods

JOIN goods\_clients ON goods.id = goods\_clients.goods\_id

JOIN ratings ON goods\_clients.id = ratings.id

GROUP BY Goodname

ORDER BY AverageRating DESC, RatingCount DESC;

**Задача 8. Създайте тригер по ваш избор** – ще създадем един тригер, който прави лог на всички нови редове, направени по таблицата ratings:

Първо си създаваме log таблицата:

CREATE TABLE ratingsLog(

id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

operation ENUM('INSERT', 'UPDATE', 'DELETE') NOT NULL,

old\_ratings\_id INT,

new\_ratings\_id INT NOT NULL,

old\_datetime DATETIME DEFAULT NULL,

new\_datetime DATETIME DEFAULT NULL,

old\_rating ENUM('1', '2', '3', '4', '5'),

new\_rating ENUM('1', '2', '3', '4', '5') NOT NULL,

old\_goods\_clients\_id INT,

new\_goods\_clients\_id INT NOT NULL,

dateOfLog DATETIME NOT NULL

);

След това и тригера:

DELIMITER @

CREATE TRIGGER InvalidRating AFTER INSERT ON ratings FOR EACH ROW

BEGIN

INSERT INTO ratingsLog(operation, old\_ratings\_id, new\_ratings\_id, old\_datetime, new\_datetime, old\_rating, new\_rating, old\_goods\_clients\_id, new\_goods\_clients\_id, dateOfLog)

VALUES('INSERT', NULL, NEW.rating, NULL, NEW.datetime, NULL, NEW.rating, NULL, NEW.goods\_clients\_id, NOW());

END;

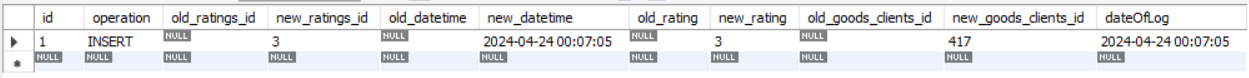
@

DELIMITER ;

Тестваме резултата с пробен insert:

INSERT INTO ratings (datetime, rating, goods\_clients\_id) values (NOW(), 3, 417);

А ето и отразения запис в ratingsLog таблицата:



**Задача 9. Създайте процедура, в която демонстрирате използване на курсор** – Ще създадем процедура, която изкарва оригиналната цена на стоката. Ще реализираме процеса с процедура, курсор и вложена таблица:

DELIMITER $

CREATE PROCEDURE originalPrice()

BEGIN

DECLARE finished INT DEFAULT 0;

DECLARE temp\_good\_name VARCHAR(255);

DECLARE temp\_current\_price FLOAT;

DECLARE temp\_price\_change FLOAT;

DECLARE fillCursor CURSOR FOR (SELECT goods.name, goods.price, SUM(changes.price\_change) FROM goods JOIN changes ON goods.id = changes.goods\_id GROUP BY goods.name);

DECLARE CONTINUE HANDLER FOR NOT FOUND SET finished = 1;

CREATE TEMPORARY TABLE tempChangeTable (good\_name VARCHAR(255), current\_price FLOAT, price\_change FLOAT);

OPEN fillCursor;

fillCursorLoop: LOOP

FETCH fillCursor INTO temp\_good\_name, temp\_current\_price, temp\_price\_change;

IF finished = 1 THEN

SET finished = 0;

LEAVE fillCursorLoop;

END IF;

INSERT INTO tempChangeTable (good\_name, current\_price, price\_change) VALUES

(temp\_good\_name, temp\_current\_price, temp\_price\_change);

END LOOP fillCursorLoop;

CLOSE fillCursor;

SELECT tempChangeTable.good\_name AS GoodName,

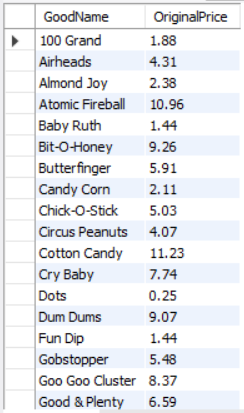
ROUND((tempChangeTable.current\_price - tempChangeTable.price\_change), 2) AS OriginalPrice

FROM tempChangeTable

ORDER BY tempChangeTable.good\_name;

DROP TEMPORARY TABLE IF EXISTS tempChangeTable;

END$



DELIMITER ;

А ето и резултата: