



# JOIN

[INNER JOIN](#)

[LEFT JOIN](#)

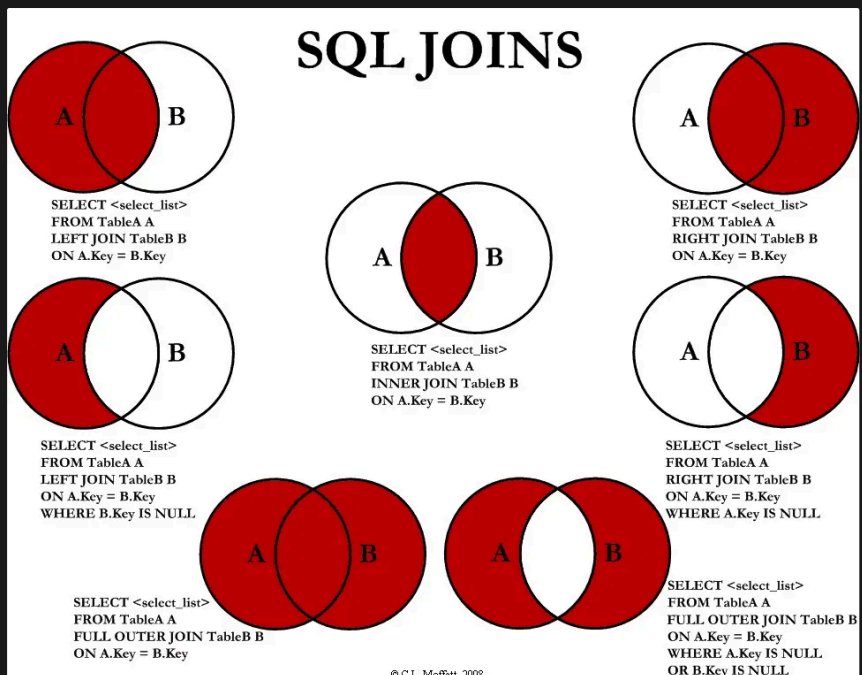
[RIGHT JOIN](#)

[SELF JOIN](#)

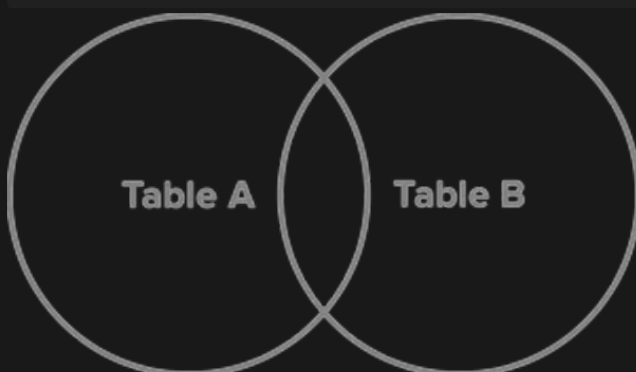
[UPDATE va JOIN](#)

[DELETE va JOIN](#)

[Masala](#)

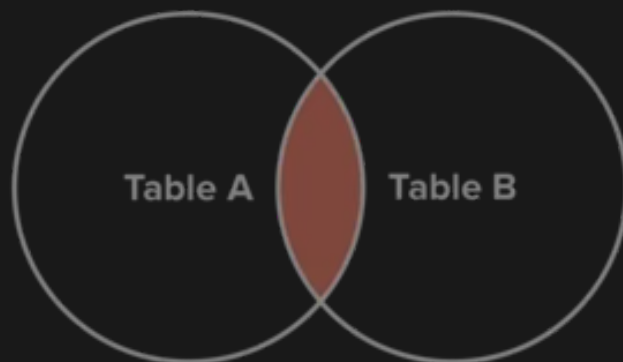


Bir nechta jadvallardan ma'lumot olish uchun JOIN lardan foydalanamiz. Misollarda k o'rsatishligimiz uchun quyidagi ikkita jadvalni yaratamiz va ichiga ma'lumotlarni insert qilamiz. Kurs nomlarini saqlash uchun course jadvali `CREATE TABLE course ( id int NOT NULL AUTO_INCREMENT PRIMARY KEY, name varchar(50) NOT NULL );` Qaysi talaba qaysi kursga borishi haqidagi ma'lumotlarni saqlash uchun "student" jadvali. Bu jadvalda hechqaysi kursga bormaydiganlari ham bo'lishi mumkin. `CREATE TABLE student ( id int unsigned NOT NULL AUTO_INCREMENT, name varchar(30) NOT NULL, course_id INT DEFAULT NULL, PRIMARY KEY (id), FOREIGN KEY (course_id) REFERENCES course(id) );` Jadvallarimizga ma'lumotlar yozamiz: "course" jadvaliga : `INSERT INTO course (name) VALUES ("HTML"); INSERT INTO course (name) VALUES ("PHP"); INSERT INTO course (name) VALUES ("JS"); INSERT INTO course (name) VALUES ("C++"); INSERT INTO course (name) VALUES ("Java"); INSERT INTO course (name) VALUES ("CSS");` "student" jadvaliga : `INSERT INTO student (name, course_id) VALUES ("Sardor", 4); INSERT INTO student (name, course_id) VALUES ("Eshmat", NULL); INSERT INTO student (name, course_id) VALUES ("Toshmat", 1); INSERT INTO student (name, course_id) VALUES ("Elyor", 4); INSERT INTO student (name, course_id) VALUES ("Ogabek", 5); INSERT INTO student (name, course_id) VALUES ("Shaxboz", NULL); INSERT INTO student (name, course_id) VALUES ("Ulugbek", 2); INSERT INTO student (name, course_id) VALUES ("Botir", 4); INSERT INTO student (name, course_id) VALUES ("Sarvar", 5); INSERT INTO student (name, course_id) VALUES ("Muhiddin", 3); INSERT INTO student (name, course_id) VALUES ("Jasur", NULL); INSERT INTO student (name, course_id) VALUES ("Samandar", 2);`



# INNER JOIN

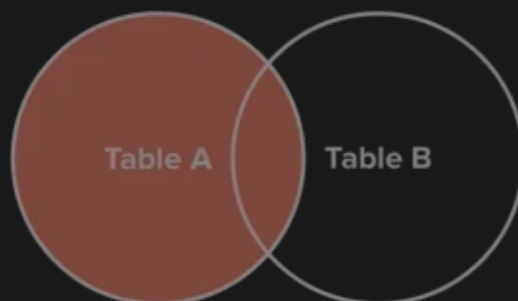
## Inner Join



Qolip: `SELECT kerakli_ustunlar FROM jadval_1 INNER JOIN jadval_2 ON boglanish_sharti;` "student" jadvalida HTML kursga boradiganlar bor, Java kursiga boradiganlar bor, HTML va JAVA kursiga boradiganlar ham bor. Umuman kursga bormaydiganlar ham bor. Bizga kim qaysi kursga borayotganligi haqida ma'lumotni olish uchun INNER JOIN dan foydalanamiz. Ya'ni ikkita jadvalda bir biriga mos bo'lgan ma'lumotlarni olish uchun INNER JOIN dan foydalanamiz. Agar 1-jadvalda ikkinchisiga bog'liq ma'lumot bo'lmasa ularni olmaydi. Ushbu so'rovda "student" jadvalidagi barchasini olib chiqadi. `SELECT * FROM student;` Endi shulardan faqat kursga boradiganlarni va qaysi kursga borishini chiqaramiz `SELECT * FROM student INNER JOIN course on course.id = student.course_id ORDER BY student.id` Natija: +-----+-----+-----+-----+ | id | name | course\_id | id | name | +-----+-----+-----+ | 1 | Sardor | 4 | 4 | C++ | | 3 | Toshmat | 1 | 1 | HTML | | 4 | Elyor | 4 | 4 | C++ | | 5 | Ogabek | 5 | 5 | Java | | 7 | Ulugbek | 2 | 2 | PHP | | 8 | Botir | 4 | 4 | C++ | | 9 | Sarvar | 5 | 5 | Java | | 10 | Muhiddin | 3 | 3 | JS | | 12 | Samandar | 2 | 2 | PHP | +-----+-----+-----+ Yana ham aniqroq ko'rinishi uchun kerakli ustunlarni select qilamiz: `SELECT student.id, student.name as Ismi, course.name as 'Kurs nomi' FROM student INNER JOIN course on course.id = student.course_id ORDER BY student.id` Natija: +-----+-----+-----+ | id | Ismi | Kurs nomi | +-----+-----+-----+ | 1 | Sardor | C++ | | 3 | Toshmat | HTML | | 4 | Elyor | C++ | | 5 | Ogabek | Java | | 7 | Ulugbek | PHP | | 8 | Botir | C++ | | 9 | Sarvar | Java | | 10 | Muhiddin | JS | | 12 | Samandar | PHP | +-----+-----+-----+ Agar bizga faqat "Java" kursiga boradiganlar kerak bo'lsa: `SELECT student.id, student.name as Ismi, course.name as 'Kurs nomi' FROM student INNER JOIN course on course.id = student.course_id WHERE course.name = 'Java' ORDER BY student.id` Natija: +-----+-----+-----+ | id | Ismi | Kurs nomi | +-----+-----+-----+ | 5 | Ogabek | Java | | 9 | Sarvar | Java | +-----+-----+-----+

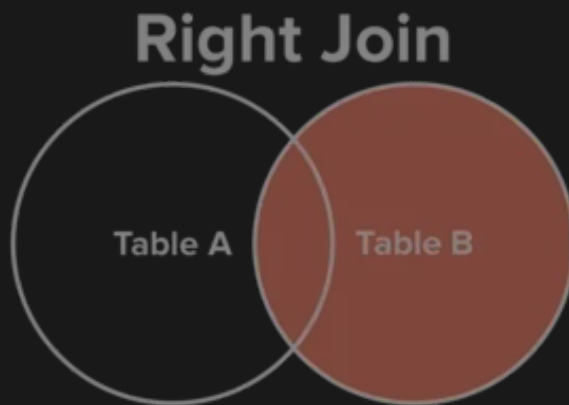
# LEFT JOIN

## Left Join



Qolip: `SELECT kerakli_ustunlar FROM jadval_1 LEFT JOIN jadval_2 ON boglanish_sharti`  
`LEFT JOIN` orqali chap tarafdagi jadvaldan barcha ma'lumotlar va ularga bog'langan jadvaldan, o'ziga tegishlilarini oladi. Yani "student" jadvalidagi barcha kursga boradigan va bormaydiganlarni oladi. Agar ichida kursga boradiganlar bo'lsa, "course" jadvalidan qaysi kursga borishini aniqlaydi. Agar kursga bormasa hech narsa chiqmaydi `LEFT` deyilishiga sabab - chap tarafda turgan jadvaldan barcha ma'lumotlar olinadi. `SELECT student.id, student.name as Ismi, course.name as 'Kurs nomi' FROM student LEFT JOIN course on course.id = student.course_id ORDER BY student.id` Natija: +-----+-----+ | id | Ismi | Kurs nomi | +-----+-----+ | 1 | Sardor | C++ | | 2 | Eshmat | NULL | <----- NULL | 3 | Toshmat | HTML | | 4 | Elyor | C++ | | 5 | Ogabek | Java | | 6 | Shaxboz | NULL | <----- NULL | 7 | Ulugbek | PHP | | 8 | Botir | C++ | | 9 | Sarvar | Java | | 10 | Muhiddin | JS | | 11 | Jasur | NULL | <----- NULL | 12 | Samandar | PHP | +-----+-----+ "student" jadvalidan, "Sardor" ismli talaba qaysi kursga borishini aniqlaymiz `SELECT student.id, student.name as Ismi, course.name as 'Kurs nomi' FROM student LEFT JOIN course on course.id = student.course_id WHERE student.name='Sardor' ORDER BY student.id` Natija: +-----+-----+ | id | Ismi | Kurs nomi | +-----+-----+ | 1 | Sardor | C++ | +-----+-----+

## RIGHT JOIN



Qolip: `SELECT kerakli_ustunlar FROM jadval_1 RIGHT JOIN jadval_2 ON boglanish_sharti`  
`RIGHT JOIN` orqali o'ng tarafdagi jadvaldan barcha ma'lumotlar va ularga bog'langan jadvaldan, o'ziga tegishlilarini oladi. `SELECT student.id, student.name as Ismi, course.name as 'Kurs nomi' FROM student RIGHT JOIN course on course.id = student.course_id ORDER BY student.id` Natija: +-----+-----+ | id | Ismi | Kurs nomi | +-----+-----+ | NULL | NULL | CSS | <-----USHBU QATORG A E'TIBOR QILING | 1 | Sardor | C++ | | 3 | Toshmat | HTML | | 4 | Elyor | C++ | | 5 | Ogabek | Java | | 7 | Ulugbek | PHP | | 8 | Botir | C++ | | 9 | Sarvar | Java | | 10 | Muhiddin | JS | | 12 | Samandar | PHP | +-----+-----+ Asosiy jadval sifatida o'ng tarafdagi, yani `RIGHT JOIN` bo'layotgan jadval olinadi va undagi barcha ma'lumotlar chiqadi. Unga mos ikkinchi jadvalda bor bo'lsa chiqadi, agar bor bo'lmasa `NULL` chiqadi. Yuqoridagidek.

## SELF JOIN

```
select
```

## UPDATE va JOIN

```

Ma'lumotlarni UPDATE qilishda ham joinlardan foydalanish mumkin. Bunda join bo'layo
tgan jadvallardagi kerakli ustunlarni UPDATE qilish mumkin Qolip: UPDATE T1, T2, [I
NNER JOIN | LEFT JOIN] T1 ON T1.c1 = T2. c1 SET T1.c2 = T2.c2, T2.c3 = expr WHERE c
ondition CREATE DATABASE IF NOT EXISTS empdb; USE empdb; -- create tables CREATE TA
BLE merits ( performance INT(11) NOT NULL, percentage FLOAT NOT NULL, PRIMARY KEY
(performance) ); CREATE TABLE employees ( emp_id INT(11) NOT NULL AUTO_INCREMENT, e
mp_name VARCHAR(255) NOT NULL, performance INT(11) DEFAULT NULL, salary FLOAT DEFAU
LT NULL, PRIMARY KEY (emp_id), CONSTRAINT fk_performance FOREIGN KEY (performance)
REFERENCES merits (performance) ); -- insert data for merits table INSERT INTO meri
ts(performance,percentage) VALUES(1,0), (2,0.01), (3,0.03), (4,0.05), (5,0.08); --
insert data for employees table INSERT INTO employees(emp_name,performance,salary)
VALUES('Mary Doe', 1, 50000), ('Cindy Smith', 3, 65000), ('Sue Greenspan', 4, 7500
0), ('Grace Dell', 5, 125000), ('Nancy Johnson', 3, 85000), ('John Doe', 2, 45000),
('Lily Bush', 3, 55000); Endi "employees" jadvalini UPDATE qilami: UPDATE employees
INNER JOIN merits ON employees.performance = merits.performance SET salary = salary
+ salary * percentage; SELECT * FROM employees; Natija: +-----+-----+-----+
-----+ | emp_id | emp_name | performance | salary | +-----+-----+
-----+ | 1 | Mary Doe | 1 | 50000 | | 2 | Cindy Smith |
3 | 66950 | | 3 | Sue Greenspan | 4 | 78750 | | 4 | Grace Dell | 5 | 135000 | | 5 |
Nancy Johnson | 3 | 87550 | | 6 | John Doe | 2 | 45450 | | 7 | Lily Bush | 3 | 5665
0 | +-----+-----+-----+-----+

```

Yuqorida INNER JOIN bilan misol ko'rdik. Endi LEFT JOIN bilan ham misol ko'ramiz. N
atijamiz aniqroq ko'rinishligi uchun "employees" jadvalimizga 2 ta yangi qiymat qo'
shamiz: INSERT INTO employees(emp\_name, performance, salary) VALUES('Jack William',
NULL, 43000), ('Ricky Bond', NULL, 52000); +-----+-----+-----+
-----+ | emp\_id | emp\_name | performance | salary | +-----+-----+-----+
-----+ | 1 | Mary Doe | 1 | 50000 | | 2 | Cindy Smith | 3 | 66950 |
| 3 | Sue Greenspan | 4 | 78750 | | 4 | Grace Dell | 5 | 135000 | | 5 | Nancy Johns
on | 3 | 87550 | | 6 | John Doe | 2 | 45450 | | 7 | Lily Bush | 3 | 56650 | | 8 | J
ack William | NULL | 43000 | -- yangi qator | 9 | Ricky Bond | NULL | 52000 | -- ya
ngi qator +-----+-----+-----+-----+ Biz manashu oxirgi qo'l
hilgan ikkita qatorni LEFT JOIN yordamida UPDATE qilishimiz mumkin: Tushunarli bo'l
ish uchun oldin LEFT JOIN yordamida SELECT qilamiz: SELECT \* FROM employees LEFT JO
IN merits ON employees.performance = merits.performance; LEFT JOIN merits ON empl
oyees.performance = merits.performance; LEFT JOIN merits ON employees.performance = m
erits.performance; Natija: +-----+-----+-----+-----+
-----+ | emp\_id | emp\_name | performance | salary | performance | per
centage | +-----+-----+-----+-----+
--+ | 1 | Mary Doe | 1 | 50000 | 1 | 0 | | 6 | John Doe | 2 | 45450 | 2 | 0.01 | |
2 | Cindy Smith | 3 | 66950 | 3 | 0.03 | | 5 | Nancy Johnson | 3 | 87550 | 3 | 0.03
| | 7 | Lily Bush | 3 | 56650 | 3 | 0.03 | | 3 | Sue Greenspan | 4 | 78750 | 4 | 0.
05 | | 4 | Grace Dell | 5 | 135000 | 5 | 0.08 | | 8 | Jack William | NULL | 43000 |
NULL | NULL | | 9 | Ricky Bond | NULL | 52000 | NULL | NULL | +-----+-----+
-----+-----+ UPDATE employees LEFT JOIN
merits ON employees.performance = merits.performance SET salary = salary + salary \*
0.015 WHERE merits.percentage IS NULL;

## DELETE va JOIN

Ma'lumotlarni DELETE qilishda ham joinlardan foydalanish mumkin. Bunda JOIN bo'layo
tgan ikkala jadvallardagi kerakli qatorlarni o'chirish mumkin: Qolip: DELETE T1, T2
FROM T1 INNER JOIN T2 ON T1.key = T2.key WHERE condition; Yuqoridagi jadval bilan m
isol keltiramiz: "employees" jadvalidan emp\_id = 1 bo'lgan qatorni, va unga "perfor
mance" ustuni orqali bog'langan "merits" jadvalidan ham o'chiramiz. Keling oldin SE
LECT qilamiz ikkala jadvalni ham: SELECT \* FROM employees INNER JOIN merits ON empl
oyees.performance = merits.performance; Natija: +-----+-----+-----+
-----+ | emp\_id | emp\_name | performance | salar
y | performance | percentage | +-----+-----+-----+
-----+ | 1 | Mary Doe | 1 | 50000 | 1 | 0 | | 6 | John Doe | 2 |
45450 | 2 | 0.01 | | 2 | Cindy Smith | 3 | 66950 | 3 | 0.03 | | 5 | Nancy Johnson |
3 | 87550 | 3 | 0.03 | | 7 | Lily Bush | 3 | 56650 | 3 | 0.03 | | 3 | Sue Greenspan
| 4 | 78750 | 4 | 0.05 | | 4 | Grace Dell | 5 | 135000 | 5 | 0.08 | +-----+-----+
-----+-----+ DELETE employees,meri
ts FROM employees INNER JOIN merits ON merits.performance = employees.performance W
HERE employees.emp\_id = 1; Yuqoridagi so'rovni yuboramiz va qaytadan SELECT qilami
z: Natija: +-----+-----+-----+-----+
--+ | emp\_id | emp\_name | performance | salary | performance | percentage | +-----+
-----+-----+-----+ | 6 | John D
oe | 2 | 45450 | 2 | 0.01 | | 2 | Cindy Smith | 3 | 66950 | 3 | 0.03 | | 5 | Nancy
Johnson | 3 | 87550 | 3 | 0.03 | | 7 | Lily Bush | 3 | 56650 | 3 | 0.03 | | 3 | Sue
Greenspan | 4 | 78750 | 4 | 0.05 | | 4 | Grace Dell | 5 | 135000 | 5 | 0.08 | +-----+
-----+-----+-----+

## Masala

"dars" jadvalidan foydalaning 1-masala - "employees" jadvalidan, hodimlarni ismi, familyasi, office qaysi city da joylashgani, va officeni telefon nomerini chiqaring

2-masala - "customers" jadvalidan, 141 nomerli mijozning ismi familyasi, uning barcha yetkazilib berilgan buyurtmalarini nomerini, buyurtma qilgan vaqti va yetkazilgan vaqtini chiqaring

3-masala - "customers" jadvalidan, 141 nomerli mijozning ismi familyasi, uning barcha to'lovlari haqida chek nomerini, to'lov qilgan sana va qancha to'lov qilganligi haqida ma'lumotlarini chiqaring

ism	familya	paymentDate	amount
Diego	Freyre	2003-07-19	36251.03
Diego	Freyre	2004-11-01	36140.38
Diego	Freyre	2005-05-19	46895.48
Diego	Freyre	2004-01-30	59830.55
Diego	Freyre	2004-12-31	116208.40
Diego	Freyre	2005-03-25	65071.26
Diego	Freyre	2005-03-18	120166.58
Diego	Freyre	2003-10-26	49539.37
Diego	Freyre	2003-02-25	40206.20
Diego	Freyre	2003-12-09	63843.55
Diego	Freyre	2004-07-09	35420.74
Diego	Freyre	2004-08-16	20009.53
Diego	Freyre	2004-05-17	26155.91

4-masala "orderdetails" jadvalidan, orderNumberi, qachon buyurtma qilgan sanasi, buyurtma holati va orderdetails jadvalidagi qolgan barcha ustunlari bilan chiqaring

5-masala "orderdetails" jadvalidan, orderNumberi, qachon buyurtma qilgan sanasi, yetkazilgan sana, orderning holati, productCode, produktning nomi, productVendori, nechta zakaz qilingan soni, har birining narxini chiqaring

6-masala "customers" jadvalidan, 141 nomerli mijozning ismi familyasi, uning barcha yetkazilib berilgan buyurtmalarinining, buyurtma qilingan sanasi va yetkazilgan sanasini, buyurtma qilgan maxsulotning nomi, maxsulotning kodi, nechta zakaz qilgani va har birini qanchadan sotilganini aniqlang

7-masala "TOP 10 PRODUCTS" masalasi - Eng ko'p buyurtma qilingan 10 ta maxsulotning, productCode, nomi va nechta buyurtma qilinganligi sonini chiqaring

productCode	productName	soni
S18_3232	1992 Ferrari 360 Spider red	1808
S18_1342	1937 Lincoln Berline	1111
S700_4002	American Airlines: MD-11S	1085
S18_3856	1941 Chevrolet Special Deluxe Cabriolet	1076
S50_1341	1930 Buick Marquette Phaeton	1074
S18_4600	1940s Ford truck	1061
S10_1678	1969 Harley Davidson Ultimate Chopper	1057
S12_4473	1957 Chevy Pickup	1056
S18_2319	1964 Mercedes Tour Bus	1053
S24_3856	1956 Porsche 356A Coupe	1052

8-masala - Eng kam buyurtma qilingan 10 ta maxsulotning, productCode, nomi, nechta buyurtma qilinganligi sonini va necha summalik buyurtma qilganini chiqaring

productCode	productName	jami	summa
S18_4933	1957 Ford Thunderbird	767	50101.57
S24_1046	1970 Chevy Chevelle SS	454	803
S3236.67			
S24_3969	1936 Mercedes Benz 500k Roadster	824	29763.39
S18_2248	1911 Ford Town Car	832	45306.77
S18_2870	1999 Indy 500 Monte Carlo SS	855	100770.12
S18_4409	1932 Alfa Romeo 8C2300 Spider Sport	866	71526.22
S24_4048	1992 Porsche Cayenne Turbo Silver	867	92973.40
S24_3191	1969 Chevrolet Camaro Z28	870	67357.30
S24_2887	1952 Citroen-15CV	873	94248.67
S18_2795	1928 Mercedes-Benz SSK	880	132275.98

9-masala - Har bir mijoz, har yilda qancha \$ lik to'lov qilganini aniqlang. Familya bo'yicha saralang.

contactFirstName	contactLastName	yil	jami
Adrian	Huxley	2003	24013.52
Adrian	Huxley	2005	31835.36
Adrian	Huxley	2004	35806.73
Akiko	Shimamura	2004	70965.55
Akiko	Shimamura	2005	34583.18
Allen	Nelson	2003	32680.31
Allen	Nelson	2005	28322.83
Allen	Nelson	2004	12530.51
Ann	Brown	2003	66884.91
Ann	Brown	2004	13671.82
Anna	O'Hara	2005	56932.30

10-masala - Mijozlar qaysi hodimning mijozni ekanligi haqida ma'lumot chiqaring. Mijozning ismi, familyasi, hodimning ismi, familyasi. Mijozning ismi bo'yicha saralang

mijozning_ismi	mijozning_familyasi	hodimning_ismi	hodimning_familyasi
Adrian	Huxley	Andy	Fixter
Akiko	Shimamura	Mami	Nishi
Alejandra	Camino	NULL	NULL
Alexander	Feuer	NULL	NULL
Alexander	Semenov	NULL	NULL
Allen	Nelson	Julie	Firrelli
Ann	Brown	Larry	Bott
Anna	O'Hara	Andy	Fixter
Annette	Roulet	Gerard	Hernandez
Armand	Kuger	NULL	NULL
Arnold	Cruz	Mami	Nishi

11-masala - 2003 - yil, noyabr oyida buyurtma qilingan maxsulotlarning, jami 130 tadan ortiq sotilganlarini, productCode, produkt nomi va nechta buyurtma qilinganligi sonini chiqaring

productCode	productName	jami
S18_3232	1992 Ferrari 360 Spider red	217
S18_3685	1948 Porsche Type 356 Roadster	132
S18_4522	1904 Buick Runabout	132
S12_3891	1969 Ford Falcon	131

12-masala - 2003 -yil dekabr oyida eng ko'p summali zakazni qilgan mijozni ismi, familyasi, buyurtma qilgan vaqti va buyurtma id sini chiqaring.

13-masala - 2003 -yil dekabr oyida, eng kam sonli zakazni qilgan mijozni ismi, familyasi, buyurtma yetkazilgan vaqti va buyurtma id sini chiqaring.

14-masala - "TOP 10 CUSTOMER" masalasi -2003 yilda TOP 10 ta eng ko'p zakazlarni amalga oshirgan mijozlarning ismi, familyasi, nechta buyurtmani amalga oshirgani haqida ma'lumot chiqaring

15-masala - 2004- yilda eng ko'p summali to'lov amalsga oshirgan 10 ta customerning , ismi familyasi, necha \$ to'laganligini aniqlang

16-masala - Umuman zakaz qilmagan mijozlarni ismi va familyasini chiqaring.