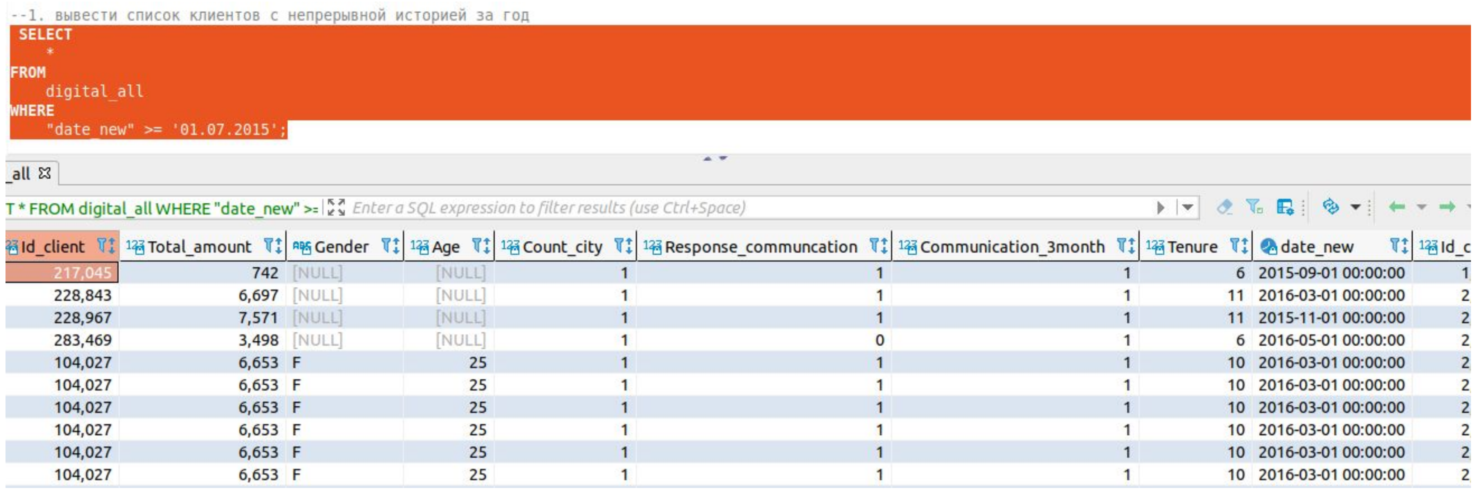
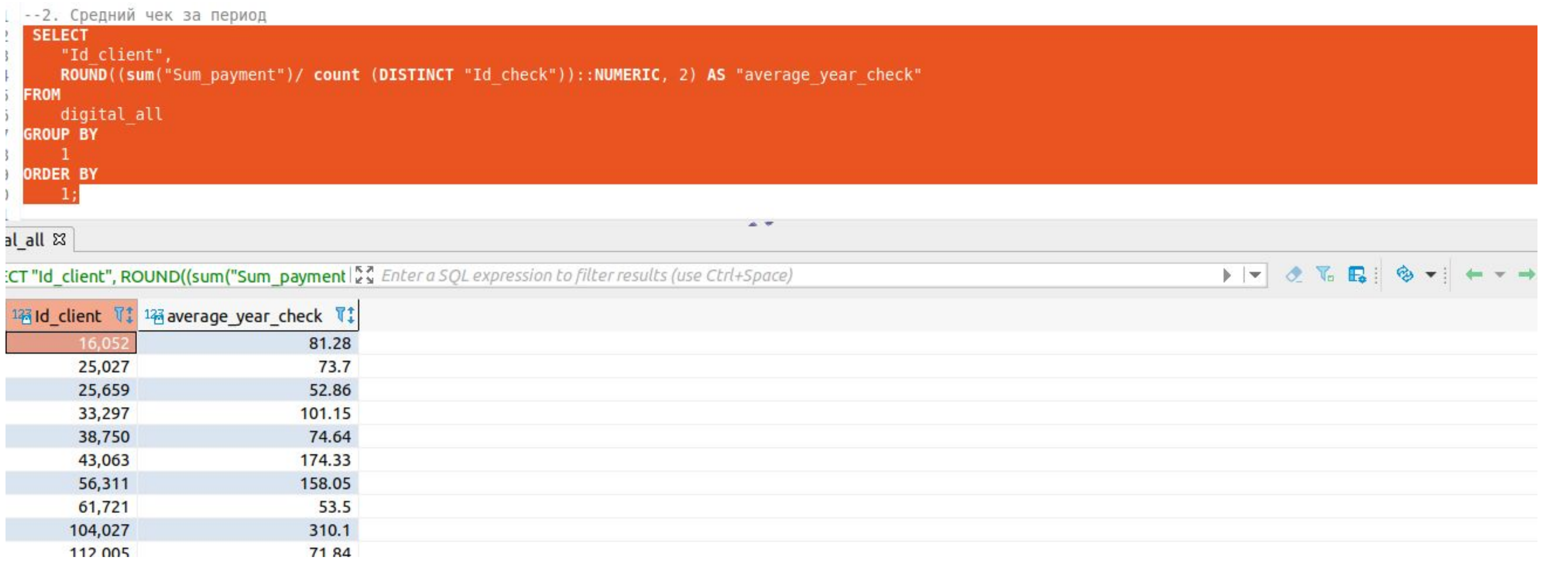


```
--1. вывести список клиентов с непрерывной историей за год
SELECT
    *
FROM
    digital_all
WHERE
    "date_new" >= '01.07.2015';
```

_all ☒

T * FROM digital_all WHERE "date_new" >= | Enter a SQL expression to filter results (use Ctrl+Space)

Id_client	Total_amount	Gender	Age	Count_city	Response_communcation	Communication_3month	Tenure	date_new	Id_c
217,045	742	[NULL]	[NULL]	1	1	1	6	2015-09-01 00:00:00	1
228,843	6,697	[NULL]	[NULL]	1	1	1	11	2016-03-01 00:00:00	2
228,967	7,571	[NULL]	[NULL]	1	1	1	11	2015-11-01 00:00:00	2
283,469	3,498	[NULL]	[NULL]	1	0	1	6	2016-05-01 00:00:00	2
104,027	6,653	F	25	1	1	1	10	2016-03-01 00:00:00	2
104,027	6,653	F	25	1	1	1	10	2016-03-01 00:00:00	2
104,027	6,653	F	25	1	1	1	10	2016-03-01 00:00:00	2
104,027	6,653	F	25	1	1	1	10	2016-03-01 00:00:00	2
104,027	6,653	F	25	1	1	1	10	2016-03-01 00:00:00	2
104,027	6,653	F	25	1	1	1	10	2016-03-01 00:00:00	2

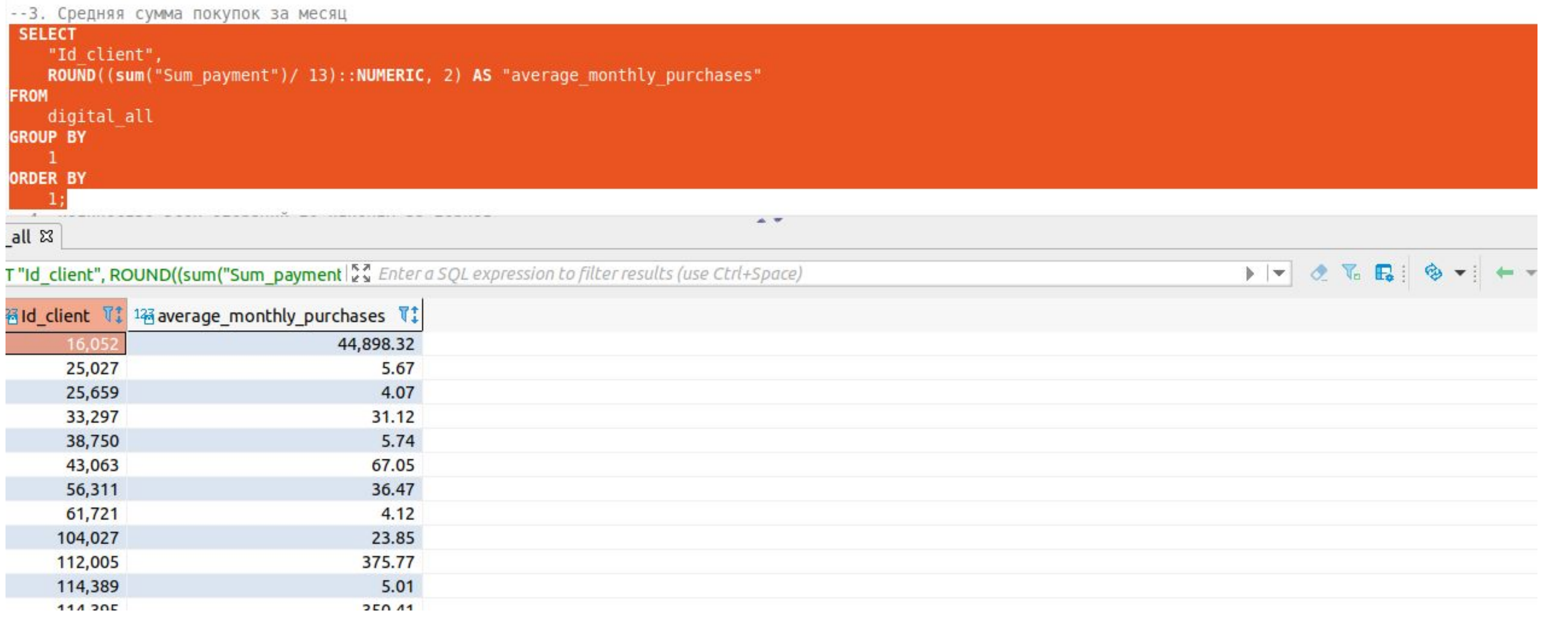


```
--2. Средний чек за период
SELECT
    "Id_client",
    ROUND((sum("Sum_payment")/ count (DISTINCT "Id_check"))::NUMERIC, 2) AS "average_year_check"
FROM
    digital_all
GROUP BY
    1
ORDER BY
    1;
```

al_all ☒

CT "Id_client", ROUND((sum("Sum_payment" | Enter a SQL expression to filter results (use Ctrl+Space)

Id_client	average_year_check
16,052	81.28
25,027	73.7
25,659	52.86
33,297	101.15
38,750	74.64
43,063	174.33
56,311	158.05
61,721	53.5
104,027	310.1
112,005	71.84

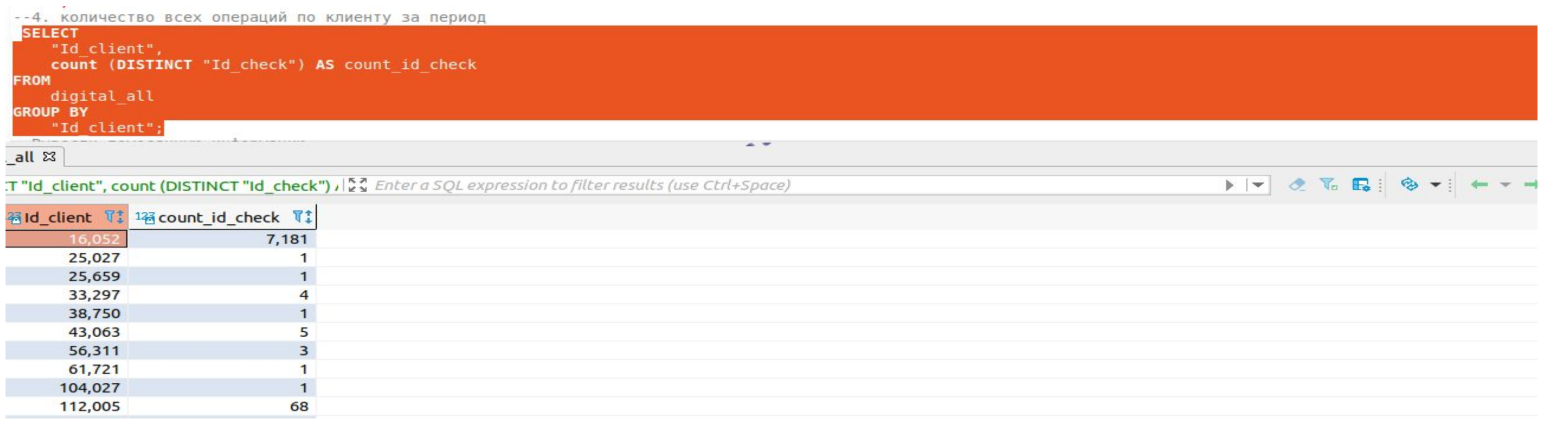


```
--3. Средняя сумма покупок за месяц
SELECT
    "Id_client",
    ROUND((sum("Sum_payment")/ 13)::NUMERIC, 2) AS "average_monthly_purchases"
FROM
    digital_all
GROUP BY
    1
ORDER BY
    1;
```

_all ☒

T "Id_client", ROUND((sum("Sum_payment" | Enter a SQL expression to filter results (use Ctrl+Space)

Id_client	average_monthly_purchases
16,052	44,898.32
25,027	5.67
25,659	4.07
33,297	31.12
38,750	5.74
43,063	67.05
56,311	36.47
61,721	4.12
104,027	23.85
112,005	375.77
114,389	5.01
114,395	350.41

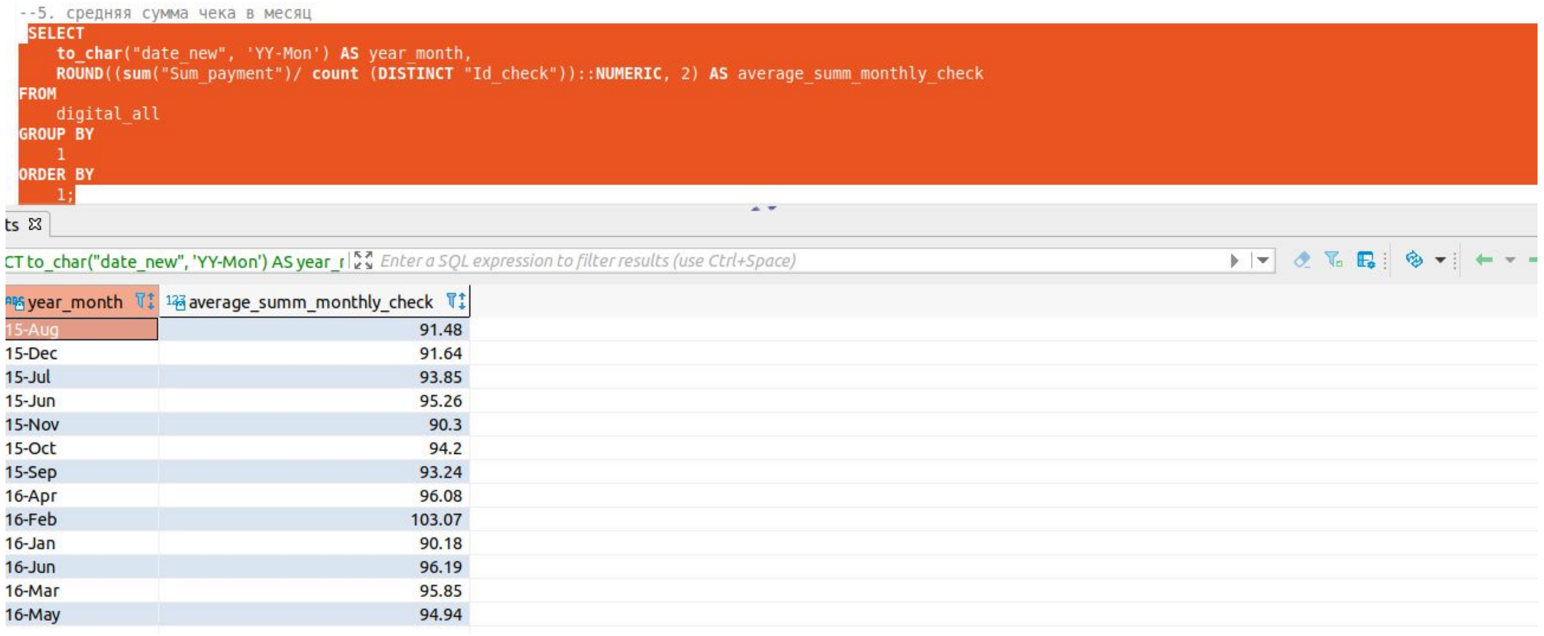


```
--4. количество всех операций по клиенту за период
 SELECT
    "Id_client",
    count (DISTINCT "Id_check") AS count_id_check
FROM
    digital_all
GROUP BY
    "Id_client";
```

_all
T "Id_client", count (DISTINCT "Id_check")
Enter a SQL expression to filter results (use Ctrl+Space)

Id_client	count_id_check
16,052	7,181
25,027	1
25,659	1
33,297	4
38,750	1
43,063	5
56,311	3
61,721	1
104,027	1
112,005	68

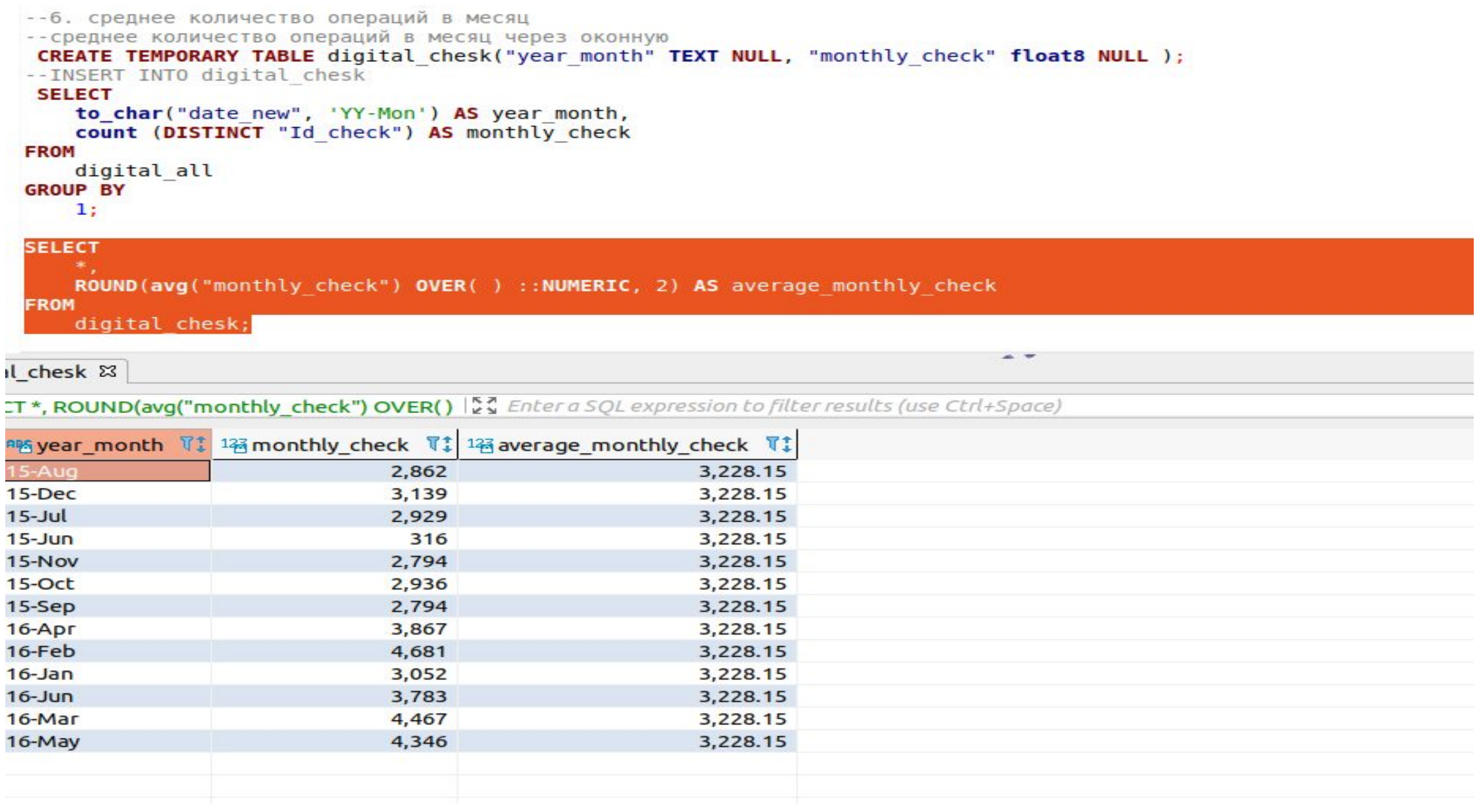
Вывести помесячную информацию:



```
--5. средняя сумма чека в месяц
 SELECT
    to_char("date_new", 'YY-Mon') AS year_month,
    ROUND((sum("Sum_payment")/ count (DISTINCT "Id_check"))::NUMERIC, 2) AS average_summ_monthly_check
FROM
    digital_all
GROUP BY
    1
ORDER BY
    1;
```

ts
CT to_char("date_new", 'YY-Mon') AS year_r
Enter a SQL expression to filter results (use Ctrl+Space)

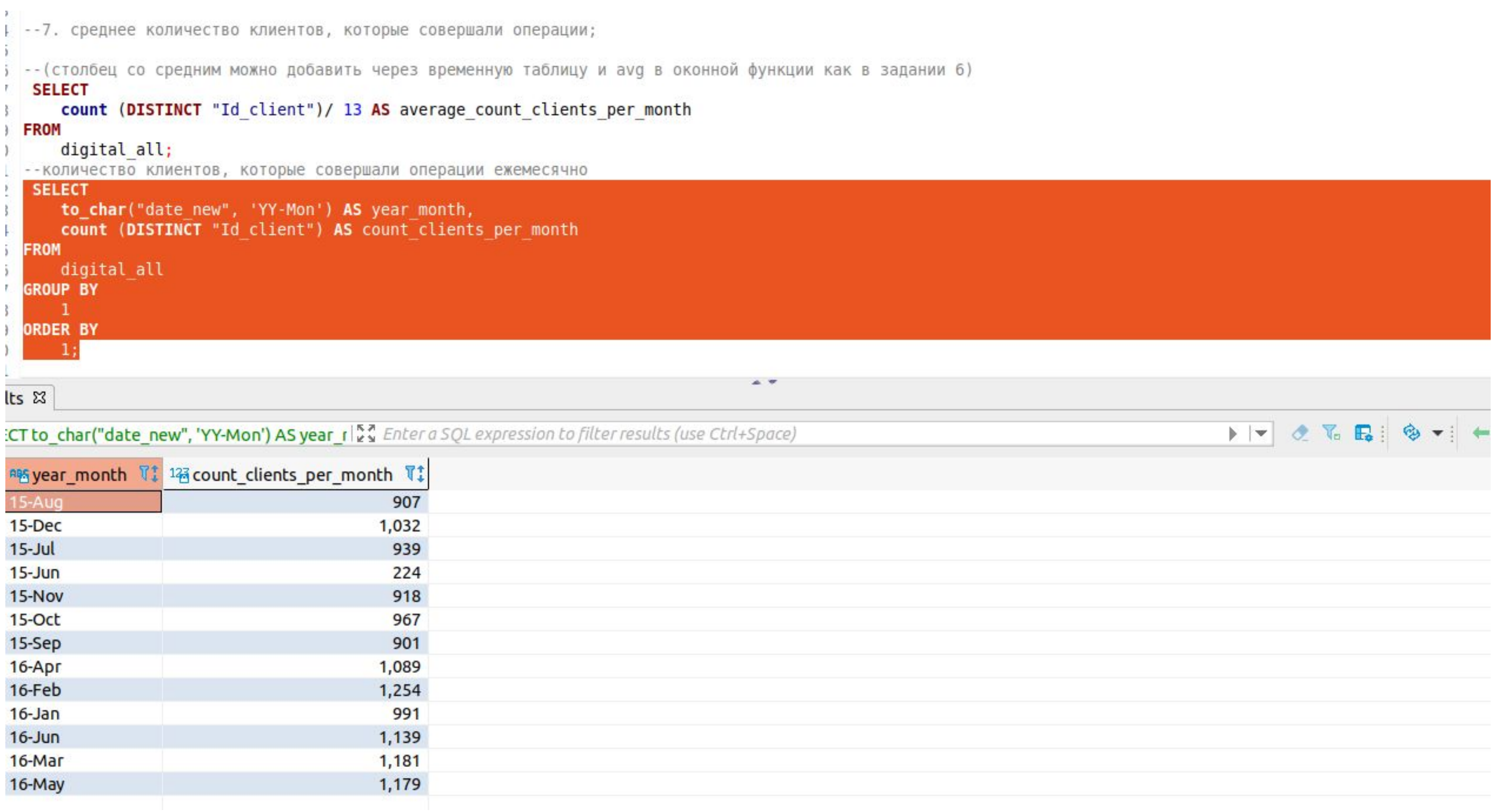
year_month	average_summ_monthly_check
15-Aug	91.48
15-Dec	91.64
15-Jul	93.85
15-Jun	95.26
15-Nov	90.3
15-Oct	94.2
15-Sep	93.24
16-Apr	96.08
16-Feb	103.07
16-Jan	90.18
16-Jun	96.19
16-Mar	95.85
16-May	94.94



```
--6. среднее количество операций в месяц
--среднее количество операций в месяц через оконную
 CREATE TEMPORARY TABLE digital_chesk("year_month" TEXT NULL, "monthly_check" float8 NULL );
--INSERT INTO digital_chesk
 SELECT
    to_char("date_new", 'YY-Mon') AS year_month,
    count (DISTINCT "Id_check") AS monthly_check
FROM
    digital_all
GROUP BY
    1;

SELECT
    *,
    ROUND(avg("monthly_check") OVER( ) ::NUMERIC, 2) AS average_monthly_check
FROM
    digital_chesk;
```

l_chesk
T *, ROUND(avg("monthly_check") OVER()
Enter a SQL expression to filter results (use Ctrl+Space)

year_month	monthly_check	average_monthly_check
15-Aug	2,862	3,228.15
15-Dec	3,139	3,228.15
15-Jul	2,929	3,228.15
15-Jun	316	3,228.15
15-Nov	2,794	3,228.15
15-Oct	2,936	3,228.15
15-Sep	2,794	3,228.15
16-Apr	3,867	3,228.15
16-Feb	4,681	3,228.15
16-Jan	3,052	3,228.15
16-Jun	3,783	3,228.15
16-Mar	4,467	3,228.15
16-May	4,346	3,228.15

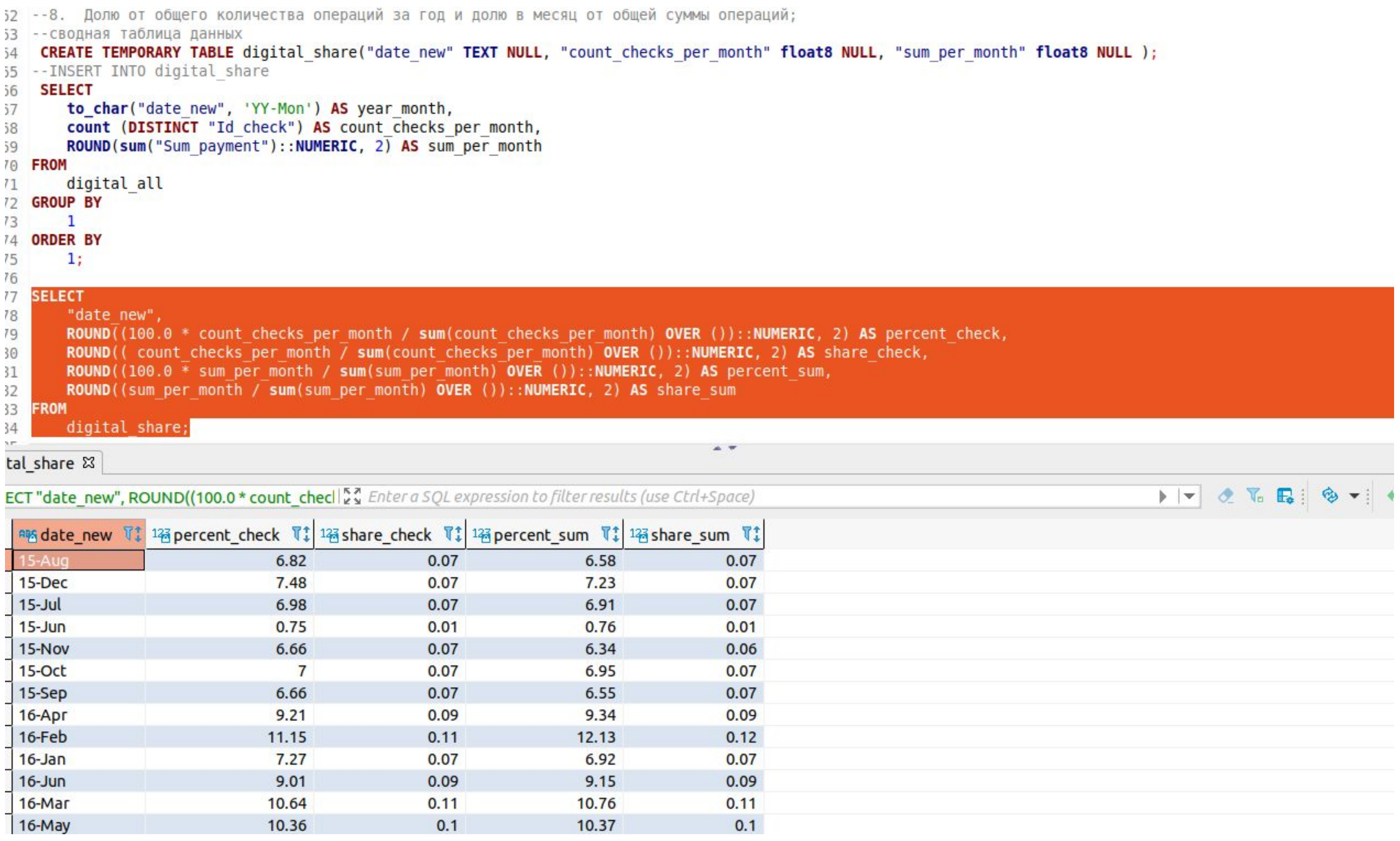


```sql
--7. среднее количество клиентов, которые совершали операции;

--(столбец со средним можно добавить через временную таблицу и avg в оконной функции как в задании 6)
 SELECT
    count (DISTINCT "Id_client")/ 13 AS average_count_clients_per_month
FROM
    digital_all;
--количество клиентов, которые совершали операции ежемесячно
 SELECT
    to_char("date_new", 'YY-Mon') AS year_month,
    count (DISTINCT "Id_client") AS count_clients_per_month
FROM
    digital_all
GROUP BY
    1
ORDER BY
    1;
```

lts ☒

ECT to_char("date_new", 'YY-Mon') AS year_r   Enter a SQL expression to filter results (use Ctrl+Space)

year_month	count_clients_per_month
15-Aug	907
15-Dec	1,032
15-Jul	939
15-Jun	224
15-Nov	918
15-Oct	967
15-Sep	901
16-Apr	1,089
16-Feb	1,254
16-Jan	991
16-Jun	1,139
16-Mar	1,181
16-May	1,179

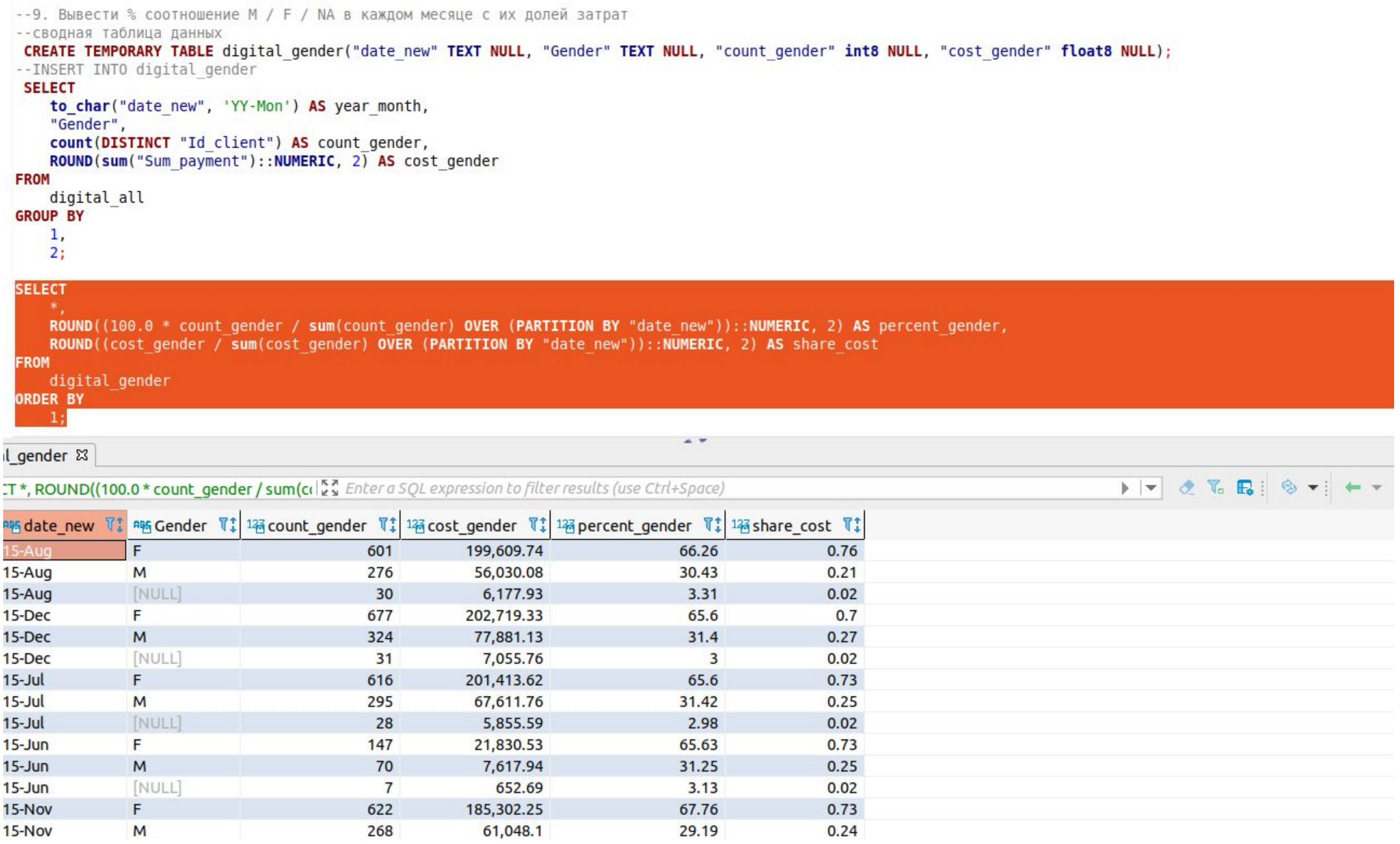


```sql
--8.  Долю от общего количества операций за год и долю в месяц от общей суммы операций;
--сводная таблица данных
 CREATE TEMPORARY TABLE digital_share("date_new" TEXT NULL, "count_checks_per_month" float8 NULL, "sum_per_month" float8 NULL );
--INSERT INTO digital_share
 SELECT
    to_char("date_new", 'YY-Mon') AS year_month,
    count (DISTINCT "Id_check") AS count_checks_per_month,
    ROUND(sum("Sum_payment")::NUMERIC, 2) AS sum_per_month
 FROM
    digital_all
 GROUP BY
    1
 ORDER BY
    1;

 SELECT
    "date_new",
    ROUND((100.0 * count_checks_per_month / sum(count_checks_per_month) OVER ())::NUMERIC, 2) AS percent_check,
    ROUND(( count_checks_per_month / sum(count_checks_per_month) OVER ())::NUMERIC, 2) AS share_check,
    ROUND((100.0 * sum_per_month / sum(sum_per_month) OVER ())::NUMERIC, 2) AS percent_sum,
    ROUND((sum_per_month / sum(sum_per_month) OVER ())::NUMERIC, 2) AS share_sum
 FROM
    digital_share;
```

tal_share ☒

ECT "date_new", ROUND((100.0 * count_checl   Enter a SQL expression to filter results (use Ctrl+Space)

date_new	percent_check	share_check	percent_sum	share_sum
15-Aug	6.82	0.07	6.58	0.07
15-Dec	7.48	0.07	7.23	0.07
15-Jul	6.98	0.07	6.91	0.07
15-Jun	0.75	0.01	0.76	0.01
15-Nov	6.66	0.07	6.34	0.06
15-Oct	7	0.07	6.95	0.07
15-Sep	6.66	0.07	6.55	0.07
16-Apr	9.21	0.09	9.34	0.09
16-Feb	11.15	0.11	12.13	0.12
16-Jan	7.27	0.07	6.92	0.07
16-Jun	9.01	0.09	9.15	0.09
16-Mar	10.64	0.11	10.76	0.11
16-May	10.36	0.1	10.37	0.1

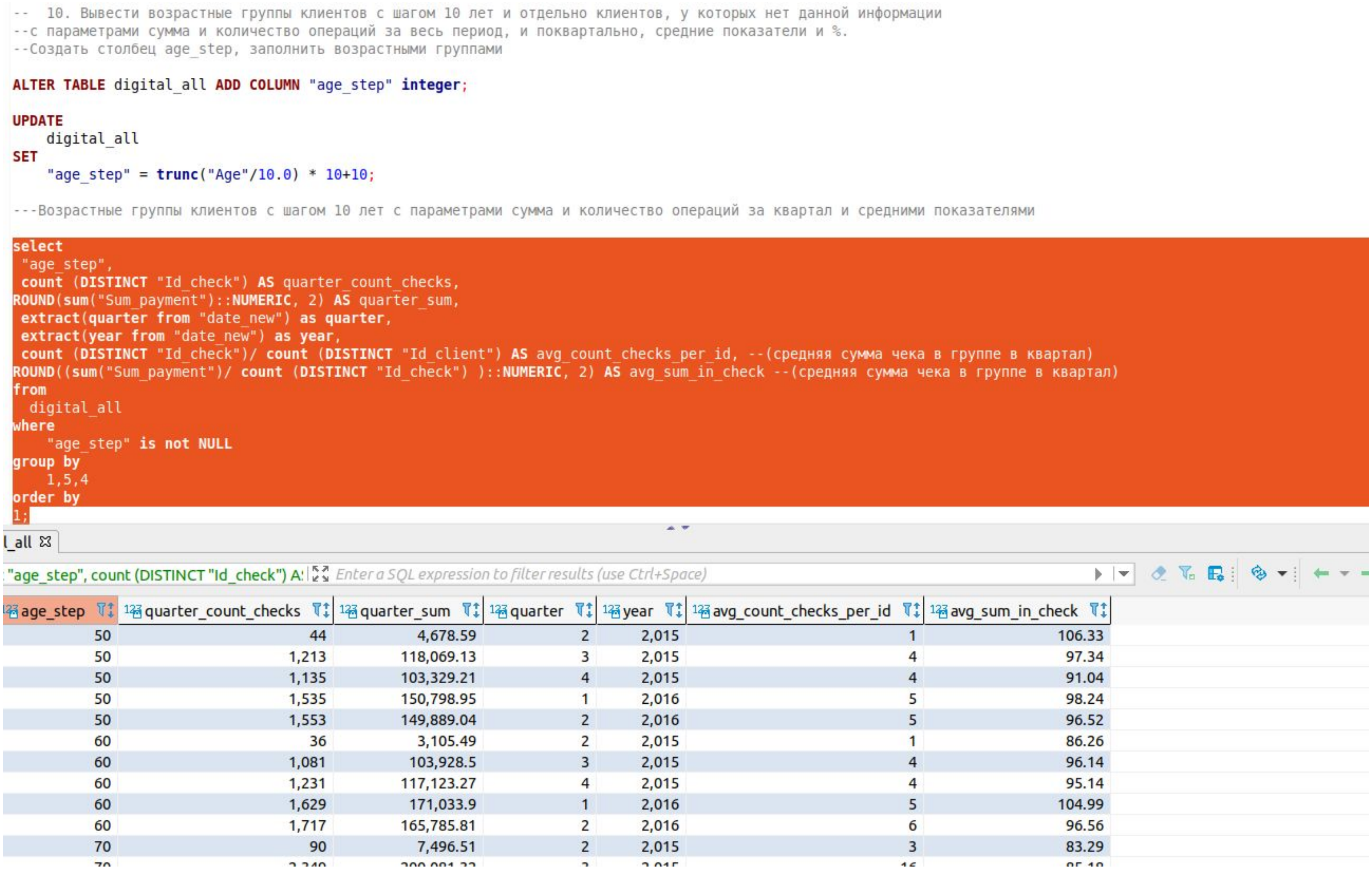


```
--9. Вывести % соотношение M / F / NA в каждом месяце с их долей затрат
--сводная таблица данных
 CREATE TEMPORARY TABLE digital_gender("date_new" TEXT NULL, "Gender" TEXT NULL, "count_gender" int8 NULL, "cost_gender" float8 NULL);
--INSERT INTO digital_gender
 SELECT
    to_char("date_new", 'YY-Mon') AS year_month,
    "Gender",
    count(DISTINCT "Id_client") AS count_gender,
    ROUND(sum("Sum_payment")::NUMERIC, 2) AS cost_gender
FROM
    digital_all
GROUP BY
    1,
    2;

SELECT
    *,
    ROUND((100.0 * count_gender / sum(count_gender) OVER (PARTITION BY "date_new"))::NUMERIC, 2) AS percent_gender,
    ROUND((cost_gender / sum(cost_gender) OVER (PARTITION BY "date_new"))::NUMERIC, 2) AS share_cost
FROM
    digital_gender
ORDER BY
    1;
```

l_gender ☒

T *, ROUND((100.0 * count_gender / sum(c Enter a SQL expression to filter results (use Ctrl+Space)

date_new	Gender	count_gender	cost_gender	percent_gender	share_cost
15-Aug	F	601	199,609.74	66.26	0.76
15-Aug	M	276	56,030.08	30.43	0.21
15-Aug	[NULL]	30	6,177.93	3.31	0.02
15-Dec	F	677	202,719.33	65.6	0.7
15-Dec	M	324	77,881.13	31.4	0.27
15-Dec	[NULL]	31	7,055.76	3	0.02
15-Jul	F	616	201,413.62	65.6	0.73
15-Jul	M	295	67,611.76	31.42	0.25
15-Jul	[NULL]	28	5,855.59	2.98	0.02
15-Jun	F	147	21,830.53	65.63	0.73
15-Jun	M	70	7,617.94	31.25	0.25
15-Jun	[NULL]	7	652.69	3.13	0.02
15-Nov	F	622	185,302.25	67.76	0.73
15-Nov	M	268	61,048.1	29.19	0.24

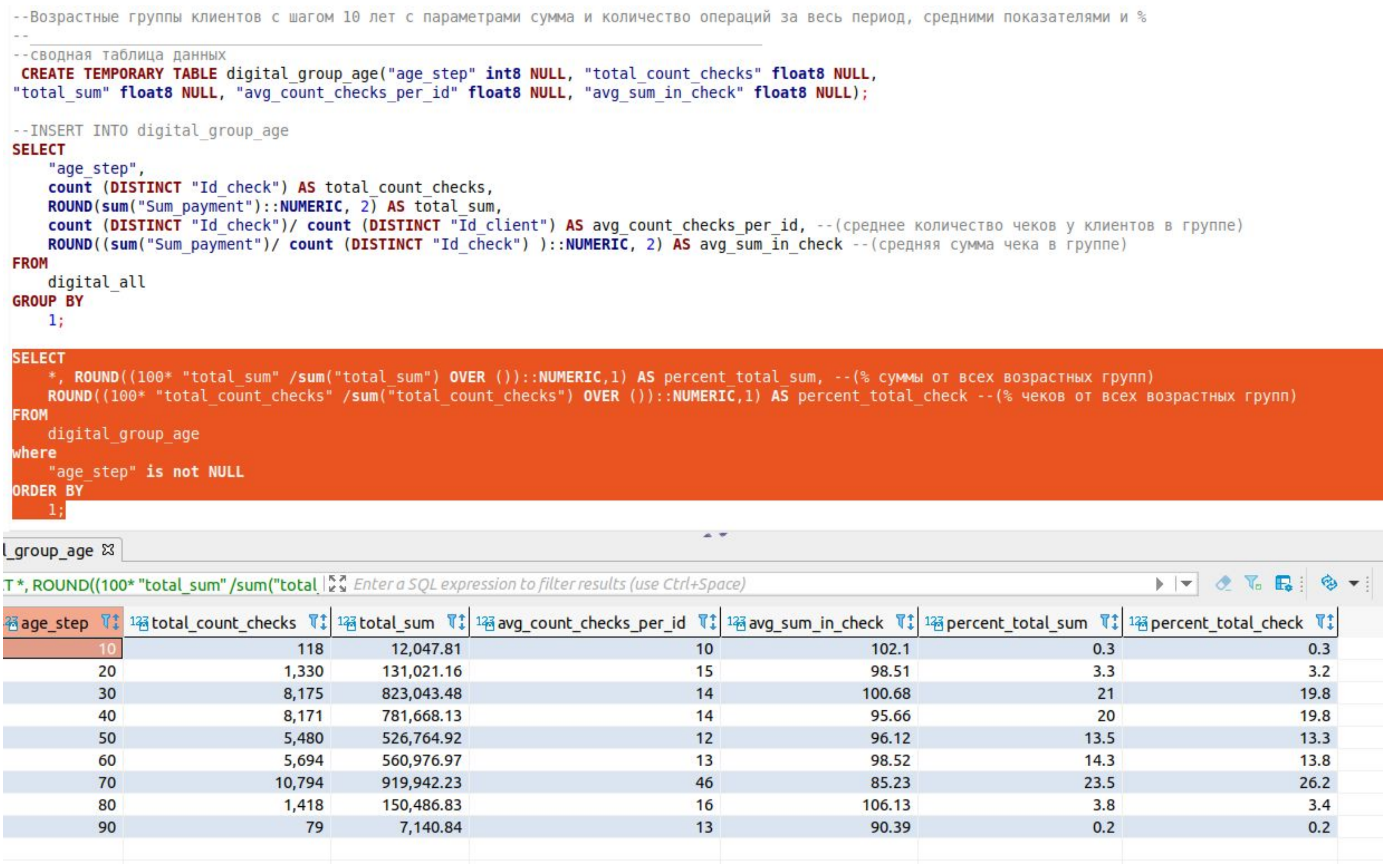


```
--  10. Вывести возрастные группы клиентов с шагом 10 лет и отдельно клиентов, у которых нет данной информации
--с параметрами сумма и количество операций за весь период, и поквартально, средние показатели и %.
--Создать столбец age_step, заполнить возрастными группами

ALTER TABLE digital_all ADD COLUMN "age_step" integer;

UPDATE
    digital_all
SET
    "age_step" = trunc("Age"/10.0) * 10+10;

---Возрастные группы клиентов с шагом 10 лет с параметрами сумма и количество операций за квартал и средними показателями

select
 "age_step",
 count (DISTINCT "Id_check") AS quarter_count_checks,
ROUND(sum("Sum_payment")::NUMERIC, 2) AS quarter_sum,
 extract(quarter from "date_new") as quarter,
 extract(year from "date_new") as year,
 count (DISTINCT "Id_check")/ count (DISTINCT "Id_client") AS avg_count_checks_per_id, --(средняя сумма чека в группе в квартал)
ROUND((sum("Sum_payment")/ count (DISTINCT "Id_check") )::NUMERIC, 2) AS avg_sum_in_check --(средняя сумма чека в группе в квартал)
from
  digital_all
where
    "age_step" is not NULL
group by
    1,5,4
order by
1;
```

l_all ☒

"age_step", count (DISTINCT "Id_check") A: Enter a SQL expression to filter results (use Ctrl+Space)

age_step	quarter_count_checks	quarter_sum	quarter	year	avg_count_checks_per_id	avg_sum_in_check
50	44	4,678.59	2	2,015	1	106.33
50	1,213	118,069.13	3	2,015	4	97.34
50	1,135	103,329.21	4	2,015	4	91.04
50	1,535	150,798.95	1	2,016	5	98.24
50	1,553	149,889.04	2	2,016	5	96.52
60	36	3,105.49	2	2,015	1	86.26
60	1,081	103,928.5	3	2,015	4	96.14
60	1,231	117,123.27	4	2,015	4	95.14
60	1,629	171,033.9	1	2,016	5	104.99
60	1,717	165,785.81	2	2,016	6	96.56
70	90	7,496.51	2	2,015	3	83.29

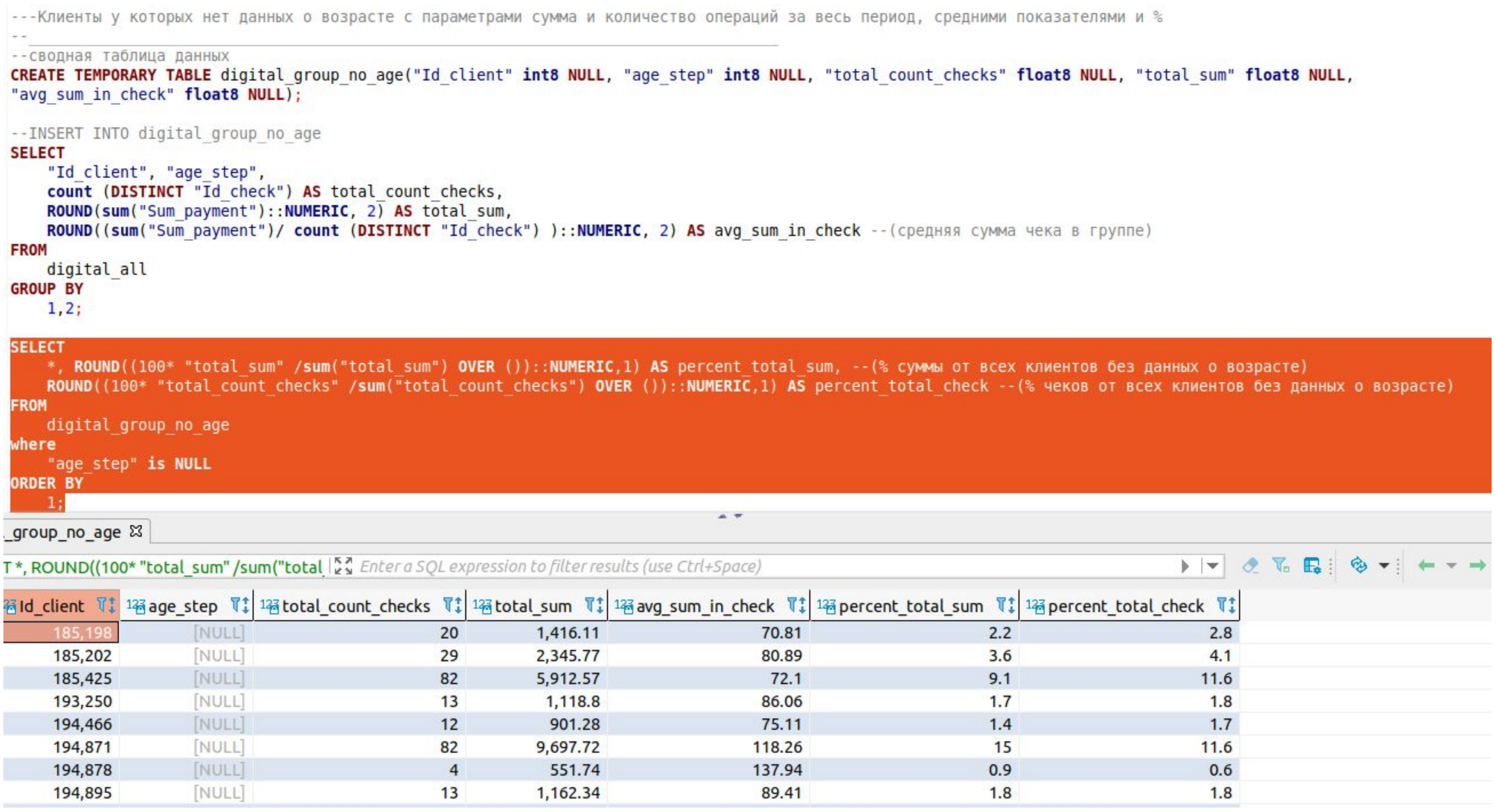


```
--Возрастные группы клиентов с шагом 10 лет с параметрами сумма и количество операций за весь период, средними показателями и %
--
--сводная таблица данных
 CREATE TEMPORARY TABLE digital_group_age("age_step" int8 NULL, "total_count_checks" float8 NULL,
"total_sum" float8 NULL, "avg_count_checks_per_id" float8 NULL, "avg_sum_in_check" float8 NULL);

--INSERT INTO digital_group_age
SELECT
    "age_step",
    count (DISTINCT "Id_check") AS total_count_checks,
    ROUND(sum("Sum_payment")::NUMERIC, 2) AS total_sum,
    count (DISTINCT "Id_check")/ count (DISTINCT "Id_client") AS avg_count_checks_per_id, --(среднее количество чеков у клиентов в группе)
    ROUND((sum("Sum_payment")/ count (DISTINCT "Id_check") )::NUMERIC, 2) AS avg_sum_in_check --(средняя сумма чека в группе)
FROM
    digital_all
GROUP BY
    1;

SELECT
    *, ROUND((100* "total_sum" /sum("total_sum") OVER ())::NUMERIC,1) AS percent_total_sum, --(% суммы от всех возрастных групп)
    ROUND((100* "total_count_checks" /sum("total_count_checks") OVER ())::NUMERIC,1) AS percent_total_check --(% чеков от всех возрастных групп)
FROM
    digital_group_age
where
    "age_step" is not NULL
ORDER BY
    1;
```

l_group_age ☒

T *, ROUND((100* "total_sum" /sum("total | Enter a SQL expression to filter results (use Ctrl+Space)

age_step	total_count_checks	total_sum	avg_count_checks_per_id	avg_sum_in_check	percent_total_sum	percent_total_check
10	118	12,047.81	10	102.1	0.3	0.3
20	1,330	131,021.16	15	98.51	3.3	3.2
30	8,175	823,043.48	14	100.68	21	19.8
40	8,171	781,668.13	14	95.66	20	19.8
50	5,480	526,764.92	12	96.12	13.5	13.3
60	5,694	560,976.97	13	98.52	14.3	13.8
70	10,794	919,942.23	46	85.23	23.5	26.2
80	1,418	150,486.83	16	106.13	3.8	3.4
90	79	7,140.84	13	90.39	0.2	0.2

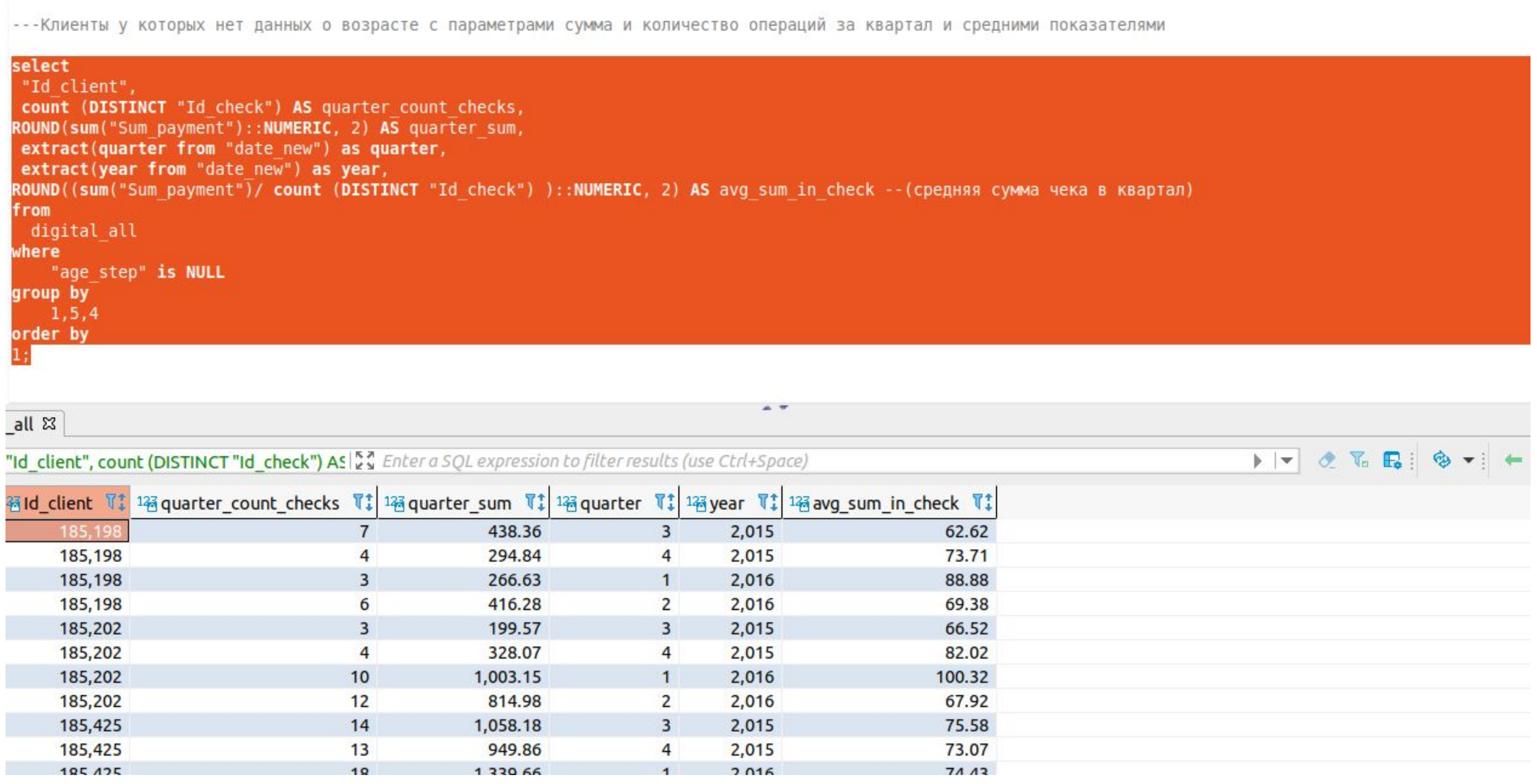


```
---Клиенты у которых нет данных о возрасте с параметрами сумма и количество операций за весь период, средними показателями и %
--
--сводная таблица данных
CREATE TEMPORARY TABLE digital_group_no_age("Id_client" int8 NULL, "age_step" int8 NULL, "total_count_checks" float8 NULL, "total_sum" float8 NULL,
"avg_sum_in_check" float8 NULL);

--INSERT INTO digital_group_no_age
SELECT
    "Id_client", "age_step",
    count (DISTINCT "Id_check") AS total_count_checks,
    ROUND(sum("Sum_payment")::NUMERIC, 2) AS total_sum,
    ROUND((sum("Sum_payment")/ count (DISTINCT "Id_check") )::NUMERIC, 2) AS avg_sum_in_check --(средняя сумма чека в группе)
FROM
    digital_all
GROUP BY
    1,2;

SELECT
    *, ROUND((100* "total_sum" /sum("total_sum") OVER ())::NUMERIC,1) AS percent_total_sum, --(% суммы от всех клиентов без данных о возрасте)
    ROUND((100* "total_count_checks" /sum("total_count_checks") OVER ())::NUMERIC,1) AS percent_total_check --(% чеков от всех клиентов без данных о возрасте)
FROM
    digital_group_no_age
where
    "age_step" is NULL
ORDER BY
    1;
```

_group_no_age ☒

T *, ROUND((100* "total_sum" /sum("total | Enter a SQL expression to filter results (use Ctrl+Space)

Id_client	age_step	total_count_checks	total_sum	avg_sum_in_check	percent_total_sum	percent_total_check
185,198	[NULL]	20	1,416.11	70.81	2.2	2.8
185,202	[NULL]	29	2,345.77	80.89	3.6	4.1
185,425	[NULL]	82	5,912.57	72.1	9.1	11.6
193,250	[NULL]	13	1,118.8	86.06	1.7	1.8
194,466	[NULL]	12	901.28	75.11	1.4	1.7
194,871	[NULL]	82	9,697.72	118.26	15	11.6
194,878	[NULL]	4	551.74	137.94	0.9	0.6
194,895	[NULL]	13	1,162.34	89.41	1.8	1.8



```
---Клиенты у которых нет данных о возрасте с параметрами сумма и количество операций за квартал и средними показателями

select
 "Id_client",
 count (DISTINCT "Id_check") AS quarter_count_checks,
ROUND(sum("Sum_payment")::NUMERIC, 2) AS quarter_sum,
 extract(quarter from "date_new") as quarter,
 extract(year from "date_new") as year,
ROUND((sum("Sum_payment")/ count (DISTINCT "Id_check") )::NUMERIC, 2) AS avg_sum_in_check --(средняя сумма чека в квартал)
from
  digital_all
where
    "age_step" is NULL
group by
    1,5,4
order by
1;
```

_all

"Id_client", count (DISTINCT "Id_check") AS

Enter a SQL expression to filter results (use Ctrl+Space)

Id_client	quarter_count_checks	quarter_sum	quarter	year	avg_sum_in_check
185,198	7	438.36	3	2,015	62.62
185,198	4	294.84	4	2,015	73.71
185,198	3	266.63	1	2,016	88.88
185,198	6	416.28	2	2,016	69.38
185,202	3	199.57	3	2,015	66.52
185,202	4	328.07	4	2,015	82.02
185,202	10	1,003.15	1	2,016	100.32
185,202	12	814.98	2	2,016	67.92
185,425	14	1,058.18	3	2,015	75.58
185,425	13	949.86	4	2,015	73.07
185,425	18	1,339.66	1	2,016	74.43