

# SQL: Simple Payments and Adjustments, for DOS Range and Payment Range

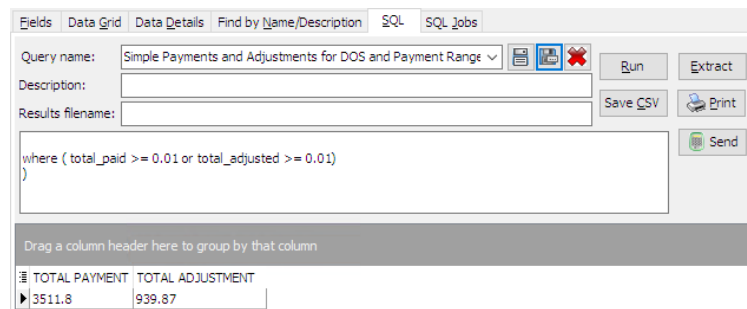
Last Modified on 01/06/2022 10:51 am EST

**i** There are **two versions** of this SQL: Firebird and MySQL. The **MySQL version** only applies to clients who are **BETA testing MySQL**. All other Practices should continue to use the Firebird version of this code. Click to expand and copy the code you need. If you are unsure which code to use, please check with your Practice Administrator.

## About

This report shows payments and adjustments received during a user-defined period, for dates of service performed during another user-defined period.

A sample image of this SQL report run in the Database Viewer is shown below:



The screenshot shows a software interface for running SQL queries. At the top, there are tabs: 'Fields', 'Data Grid', 'Data Details', 'Find by Name/Description', 'SQL', and 'SQL Jobs'. The 'SQL' tab is active. Below the tabs, there are input fields for 'Query name:' (set to 'Simple Payments and Adjustments for DOS and Payment Range'), 'Description:', and 'Results filename:'. To the right of these fields are buttons: 'Run', 'Extract', 'Save CSV', 'Print', and 'Send'. Below the input fields is a text area containing the SQL query: 

```
where ( total_paid >= 0.01 or total_adjusted >= 0.01 )
```

. Below the text area is a section labeled 'Drag a column header here to group by that column'. At the bottom, there is a table with two columns: 'TOTAL PAYMENT' and 'TOTAL ADJUSTMENT'. The first row of data shows '3511.8' under 'TOTAL PAYMENT' and '939.87' under 'TOTAL ADJUSTMENT'.

## Caveats

- Voided amounts are excluded.
- "Payments received dates" are defined as the OP payment date, rather than the posted payment date or the daysheet date.

## SQL Code: Firebird

To highlight and copy the code below to your clipboard, simply click the **Copy** button.

Copy

```
select round(sum(total_paid),2) as total_payment, round(sum(total_adjusted),2) as total_adjustment from
(
select patno, dos, cptcode, total_paid, total_adjusted from
(
select patno, date1 as dos, trnsxno, cptcode from archive_transactions at1 where
archive_Flag = 1 and date1 between :dos_start and :dos_end and txnopaid = 0
) a
left outer join
(select txnopaid, sum(payment+copayrecd) as total_paid, sum(copayadjustment+adjustment) as total_adjusted from ar
chive_transactions where archive_flag = 1 and date1 between :payment_start and :payment_end and txnopaid >0 group
by txnopaid) b on a.trnsxno = b.txnopaid

where ( total_paid >= 0.01 or total_adjusted >= 0.01)
)
```

## SQL Code: MySQL

To highlight and copy the code below to your clipboard, simply click the **Copy** button.

Copy

```
select a.patno, dos, cptcode, staffname as rendering_provider,code1 as appt_type, total_ins_paid, total_ins_adjusted fro
m
(
select patno, date1 as dos, trnsxno, rend_addr_id, cptcode from archive_transactions at1 where
archive_Flag = 1 and date1 between :dos_start_date and :dos_end_date and txnopaid = 0
) a
left outer join
(select txnopaid, sum(payment) as total_ins_paid, sum(adjustment) as total_ins_adjusted from archive_transactions whe
re archive_flag = 1 and date1 between :payment_start_date and :payment_end_date and
txnopaid >0 group by txnopaid) b on a.trnsxno = b.txnopaid
left outer join staff1 on
staff1.staffid = a.rend_addr_id
left outer join (select appt_date,code1, patno from schedule where appt_date between :dos_start_date and :dos_end_dat
e and visit_status not in ( 'Cancelled','No Show')) s on a.patno = s.patno and s.appt_date = a.dos
where ( total_ins_paid >= 0.01 or total_ins_adjusted >= 0.01)
```