# May 9

## Hardship data

SELECT aq.[pk]

,aq.[account\_no]

,tt.description as transaction\_type

,aq.[transact\_sub\_type]

,cn.description as credit\_note

,aq.fk\_transact\_status

,aq.[fk\_account]

,aq.[amount]

,aq.[date\_new]

,aq.[date\_active]

,aq.[date\_applied]

,aq.[date\_voided]

,aq.[comments]

FROM [AQTEST].[dbo].[aq\_transaction] as aq,

[AQTEST].[dbo].[payment\_tran] as pt,

[AQTEST].[dbo].[credit\_note] as cn,

[AQTEST].[dbo].[transact\_type] as tt

where aq.pk = pt.pk\_transaction

and pt.fk\_credit\_note = cn.code

and aq.fk\_transact\_type = tt.code

and aq.[fk\_transact\_type] = 'P'

and aq.[transact\_sub\_type] in ('CN', 'CNR', 'W', 'WR')

and aq.date\_applied = '2024-07-08 00:00:00.000'

1. **Create a New Repository on GitHub:**
   * Log in to your GitHub account.
   * Click the "+" icon in the top right and select "New repository".
   * Give your repository a name, choose its visibility (public/private), and click "Create repository".  
     *(For more details, see* [*GitHub’s Create a Repo guide*](https://docs.github.com/en/get-started/quickstart/create-a-repo)*).*

https://github.com/AndrewDilley/ESC.git

1. **Link Your Local Repository to GitHub:**  
   In your terminal, add a remote pointing to your new GitHub repository. Replace <YourUsername> and <RepoName> with your details:

CD C:\Users\andrew.dilley\development\ESC

git init

git remote add origin https://github.com/AndrewDilley/ESC.git

git add .

git commit -m "initial commit"

git remote -v

git push --set-upstream origin master

git push

# May 12

# Hardship rebate data

/\*

This query:

- returns the hardship rebate data which is currently copied and pasted from Aquarate

- into the Hardship Rebates tab of the FC2024/11363 spreadsheet

- developed and tested by Andrew Dilley on 12/5/25

\*/

SELECT

aq.[pk],

aq.[account\_no],

tt.description AS transaction\_type,

ptt.description AS transaction\_sub\_type,

aq.[transact\_sub\_type],

cn.description AS credit\_note,

aq.fk\_transact\_status as status,

aq.fk\_account,

aq.amount,

aq.date\_active date\_raised,

aq.date\_applied,

aq.date\_voided,

aq.comments

FROM

[AQTEST].[dbo].[aq\_transaction] AS aq

JOIN

[AQTEST].[dbo].[payment\_tran] AS pt ON aq.pk = pt.pk\_transaction

JOIN

[AQTEST].[dbo].[credit\_note] AS cn ON pt.fk\_credit\_note = cn.code

JOIN

[AQTEST].[dbo].[transact\_type] AS tt ON aq.fk\_transact\_type = tt.code

JOIN

[AQTEST].[dbo].[pay\_tran\_type] AS ptt ON aq.transact\_sub\_type = ptt.code

WHERE

aq.fk\_transact\_type = 'P'

AND aq.transact\_sub\_type IN ('CN', 'CNR', 'W', 'WR')

AND cn.description IN ('Hardship Long Term Debt Write Off', 'Hardship Bonus Credits')

AND aq.date\_active BETWEEN '2024-07-01' AND '2025-06-30'

ORDER BY

aq.date\_active,

aq.account\_no;

Validations

WITH base\_data AS (

SELECT

aq.account\_no,

aq.amount,

aq.date\_active,

YEAR(aq.date\_active) AS year,

MONTH(aq.date\_active) AS month,

CAST(aq.date\_active AS DATE) AS txn\_date

FROM

[AQTEST].[dbo].[aq\_transaction] AS aq

JOIN

[AQTEST].[dbo].[payment\_tran] AS pt ON aq.pk = pt.pk\_transaction

JOIN

[AQTEST].[dbo].[credit\_note] AS cn ON pt.fk\_credit\_note = cn.code

JOIN

[AQTEST].[dbo].[transact\_type] AS tt ON aq.fk\_transact\_type = tt.code

JOIN

[AQTEST].[dbo].[pay\_tran\_type] AS ptt ON aq.transact\_sub\_type = ptt.code

WHERE

aq.fk\_transact\_type = 'P'

AND aq.transact\_sub\_type IN ('CN', 'CNR', 'W', 'WR')

AND cn.description IN ('Hardship Long Term Debt Write Off', 'Hardship Bonus Credits')

AND aq.date\_active BETWEEN '2024-07-01' AND '2025-06-30'

),

first\_seen AS (

SELECT

account\_no,

MIN(txn\_date) AS first\_seen\_date

FROM base\_data

GROUP BY account\_no

),

combined AS (

SELECT

b.year,

b.month,

b.amount,

CASE

WHEN b.txn\_date = f.first\_seen\_date THEN 1

ELSE 0

END AS is\_new\_account

FROM base\_data b

LEFT JOIN first\_seen f ON b.account\_no = f.account\_no

)

SELECT

year,

month,

SUM(amount) AS hardship\_grants\_dollars,

SUM(is\_new\_account) AS hardship\_grants\_number

FROM combined

GROUP BY year, month

ORDER BY year, month;

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Community rebate

Write off data

/\*

This query:

- returns the Net Tenant Debt Abandoned or write off data, which is currently copied and pasted from Aquarate

- into the month tabs of the FC2024/11363 spreadsheet

- and then used in the Summary Tab (called Net Tenant Debt Abandoned)

- developed and tested by Andrew Dilley on 12/5/25

\*/

SELECT

aq.[pk],

aq.[account\_no],

tt.description AS transaction\_type,

aq.[transact\_sub\_type],

aq.fk\_transact\_status as status,

aq.fk\_account,

aq.amount,

aq.date\_active date\_raised,

aq.date\_applied,

aq.date\_voided,

aq.comments

FROM

[AQTEST].[dbo].[aq\_transaction] AS aq

JOIN

[AQTEST].[dbo].[payment\_tran] AS pt ON aq.pk = pt.pk\_transaction

JOIN

[AQTEST].[dbo].[transact\_type] AS tt ON aq.fk\_transact\_type = tt.code

WHERE

aq.fk\_transact\_type = 'P'

and LTRIM(RTRIM(aq.transact\_sub\_type)) in ('W', 'WR')

AND aq.date\_active BETWEEN '2024-07-01' AND '2025-06-30'

ORDER BY

aq.date\_active,

aq.account\_no;

validating the write off data

SELECT

YEAR(aq.date\_active) AS year,

MONTH(aq.date\_active) AS month,

SUM(aq.amount) AS total\_amount,

COUNT(CASE WHEN LTRIM(RTRIM(aq.transact\_sub\_type)) = 'W' THEN 1 END) AS w\_count,

COUNT(CASE WHEN LTRIM(RTRIM(aq.transact\_sub\_type)) = 'WR' THEN 1 END) AS wr\_count,

COUNT(CASE WHEN LTRIM(RTRIM(aq.transact\_sub\_type)) = 'W' THEN 1 END)

- COUNT(CASE WHEN LTRIM(RTRIM(aq.transact\_sub\_type)) = 'WR' THEN 1 END)

AS accounts\_abandoned\_no

FROM

[AQTEST].[dbo].[aq\_transaction] AS aq

JOIN

[AQTEST].[dbo].[payment\_tran] AS pt ON aq.pk = pt.pk\_transaction

JOIN

[AQTEST].[dbo].[transact\_type] AS tt ON aq.fk\_transact\_type = tt.code

WHERE

aq.fk\_transact\_type = 'P'

AND LTRIM(RTRIM(aq.transact\_sub\_type)) IN ('W', 'WR')

AND aq.date\_active BETWEEN '2024-07-01' AND '2025-06-30'

GROUP BY

YEAR(aq.date\_active),

MONTH(aq.date\_active)

ORDER BY

YEAR(aq.date\_active),

MONTH(aq.date\_active);

SELECT t.name AS TableName, c.name AS ColumnName

FROM sys.columns c

JOIN sys.tables t ON c.object\_id = t.object\_id

WHERE c.name LIKE '%description%';

Property counts

SELECT

v.fk\_district,

d.description AS district,

v.fk\_bill\_type,

COUNT(\*) AS record\_count

FROM

[AQTEST].[dbo].[v\_sts\_prop\_annual\_count\_base] AS v

JOIN

[AQTEST].[dbo].[district] AS d ON v.fk\_district = d.code

WHERE

v.pk\_sts\_result\_run = 10818

AND v.fk\_financial\_year = 202324

AND v.pk\_sts\_prop\_count\_type = 'WBA'

GROUP BY

v.fk\_district,

d.description,

v.fk\_bill\_type

ORDER BY

v.fk\_district,

v.fk\_bill\_type;

SELECT

fk\_bill\_type,

COUNT(\*) AS record\_count

FROM

[AQTEST].[dbo].[v\_sts\_prop\_annual\_count\_base]

WHERE

[pk\_sts\_result\_run] = 10818

-- AND [fk\_financial\_year] = 202324

AND pk\_sts\_prop\_count\_type = 'WP'

GROUP BY

fk\_bill\_type

ORDER BY

fk\_bill\_type;

SELECT TOP (1000) [code]

,[description]

,[user\_mod\_ts]

,[user\_add]

,[user\_add\_ts]

,[user\_mod]

,[fk\_sts\_result\_type]

,[rowid]

FROM [AQTEST].[dbo].[sts\_prop\_count\_type]

A screenshot of a computer

AI-generated content may be incorrect.

SELECT TOP (1000) [pk]

,[fk\_sts\_result\_type]

,[fk\_sts\_result\_frequency]

,[result\_name]

,[next\_run\_date]

,[last\_run\_date]

,[result\_view\_name]

,[retention\_days]

,[user\_mod\_ts]

,[user\_add]

,[user\_add\_ts]

,[user\_mod]

,[rowid]

,[hide\_result\_flag]

,[narrative]

FROM [AQTEST].[dbo].[sts\_result]

A screenshot of a computer

AI-generated content may be incorrect.

CD C:\Users\andrew.dilley\development\ESC

git add .

git commit -m "further sql development"

git remote -v

git push