

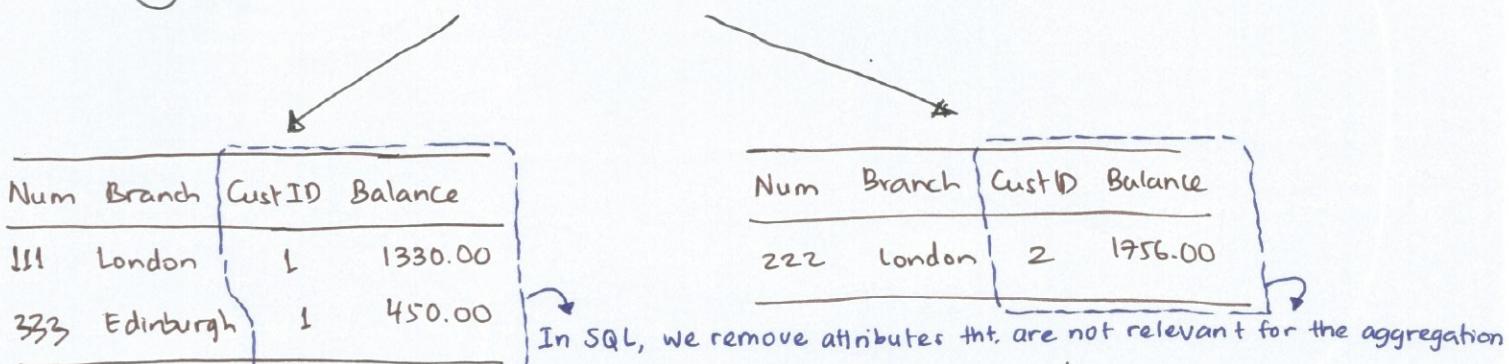
AGGREGATION W/ GROUPING

→ QN: How much money does each customer have in total across all his accounts?

Account

Num	Branch	CustID	Balance
111	London	1	1330.00
222	London	2	1756.00
333	Edinburgh	1	450.00

① Partition Account into **groups** of rows by CustID



② Sum balances in each group separately

CustID	SUM
1	1780.00

CustID	SUM
2	1756.00

③ Take the **union** of the results for each group

SQL

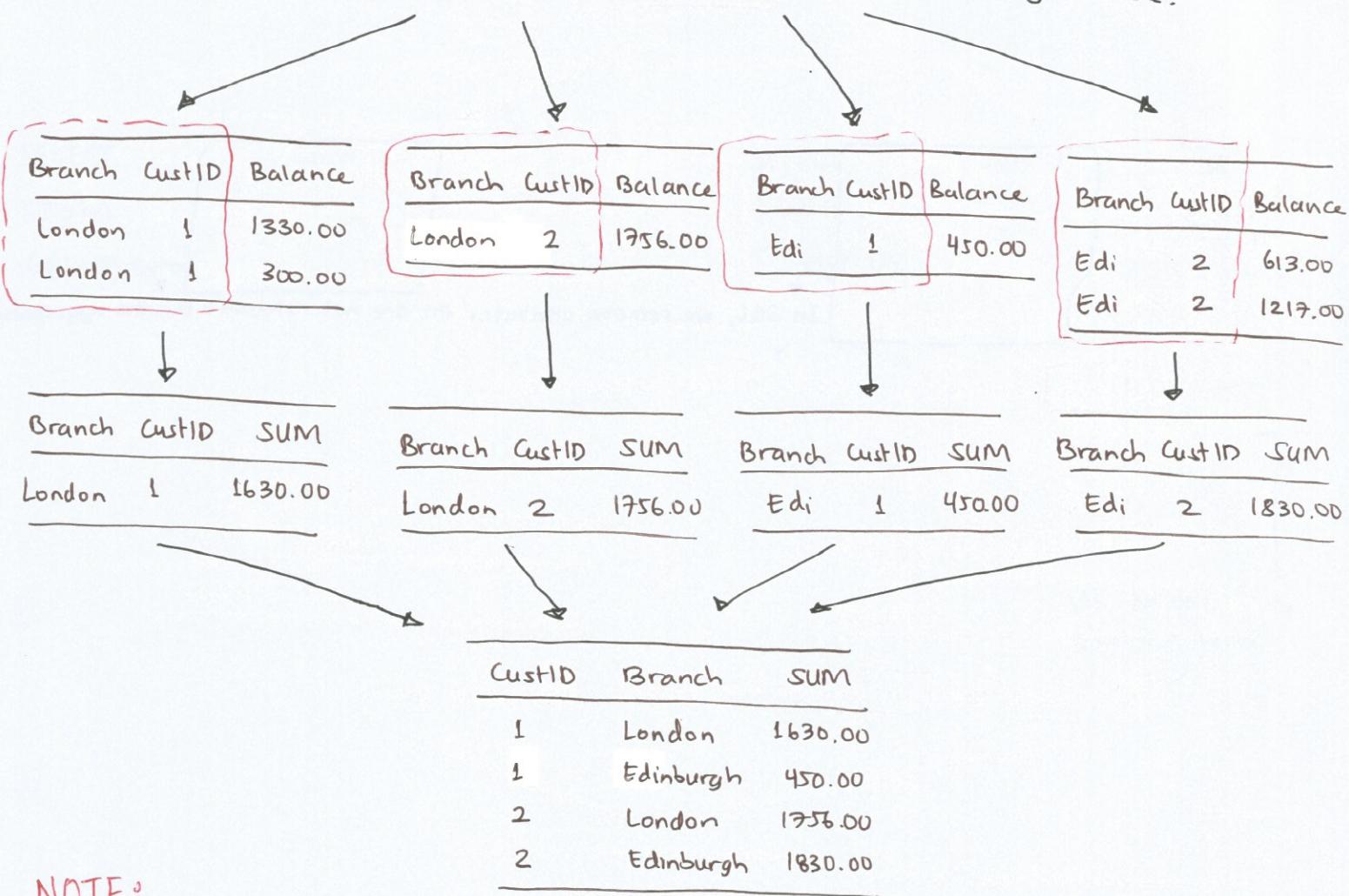
```
SELECT A.custid, SUM(A.balance)
FROM Account A
GROUP BY A.custid;
```

CustID	SUM
1	1780.00
2	1756.00

→ QN: How much money does each customer have in each branch?

Account			
Num	Branch	CustID	Balance
111	London	1	1330.00
222	London	2	1756.00
333	Edinburgh	1	450.00
444	Edinburgh	2	613.00
555	London	1	300.00
666	Edinburgh	2	1217.00

Grouping is now done by pairs of values, rather than single value.



NOTE:

In queries w/ GROUP BY,
attributes in SELECT must:

(1). appear in the GROUP BY

OR

(2). used in an aggregate fn

SQL

```
SELECT A.custid, A.branch, SUM(A.balance)
FROM Account A
GROUP BY A.custid, A.branch;
```

FILTERING

→ QN: What are the branches w/ a total balance (across accounts) of at least 500?

Account			
Num	Branch	CustID	Balance
111	London	1	1330.00
222	London	2	1756.00
333	Edinburgh	1	450.00

```
SELECT A.branch, SUM(A.balance)  
FROM Account A  
GROUP BY A.branch  
HAVING SUM(A.balance) >= 500;
```

→ Note we cannot use WHERE instead of HAVING bc. aggregates have not been computed.

→

Branch	SUM
London	3086.00

ORDER OF EVALUATION

- 1) Take rows from the tables listed in FROM
- 2) Discard rows not satisfying the WHERE condition
- 3) Partition rows according to attributes in GROUP BY
- 4) Compute aggregates
- 5) Discard rows not satisfying the HAVING condition
- 6) Output the values of expressions listed in SELECT

AGGREGATION & ARITHMETIC

→ QN: Display the money available in total to each customer across his accounts

Account					
Num	Branch	Cust ID	Balance	Spend	
111	London	1	1330.00	250.00	
222	London	2	1756.00	356.00	
333	Edi	1	450.00	0.00	

2 METHODS

```
SELECT A.custid, SUM(A.balance - A.spend)  
FROM Account A  
GROUP BY A.custid
```

```
SELECT A.custid, SUM(A.balance) - SUM(A.spend)  
FROM Account A  
GROUP BY A.custid
```

custid	SUM
1	1530.00
2	1400.00

custid	?column?
1	1530.00
2	1400.00

The col. name
that results from
this will have
the default
Postgres name
which is ...

We can use AS to
give it a name