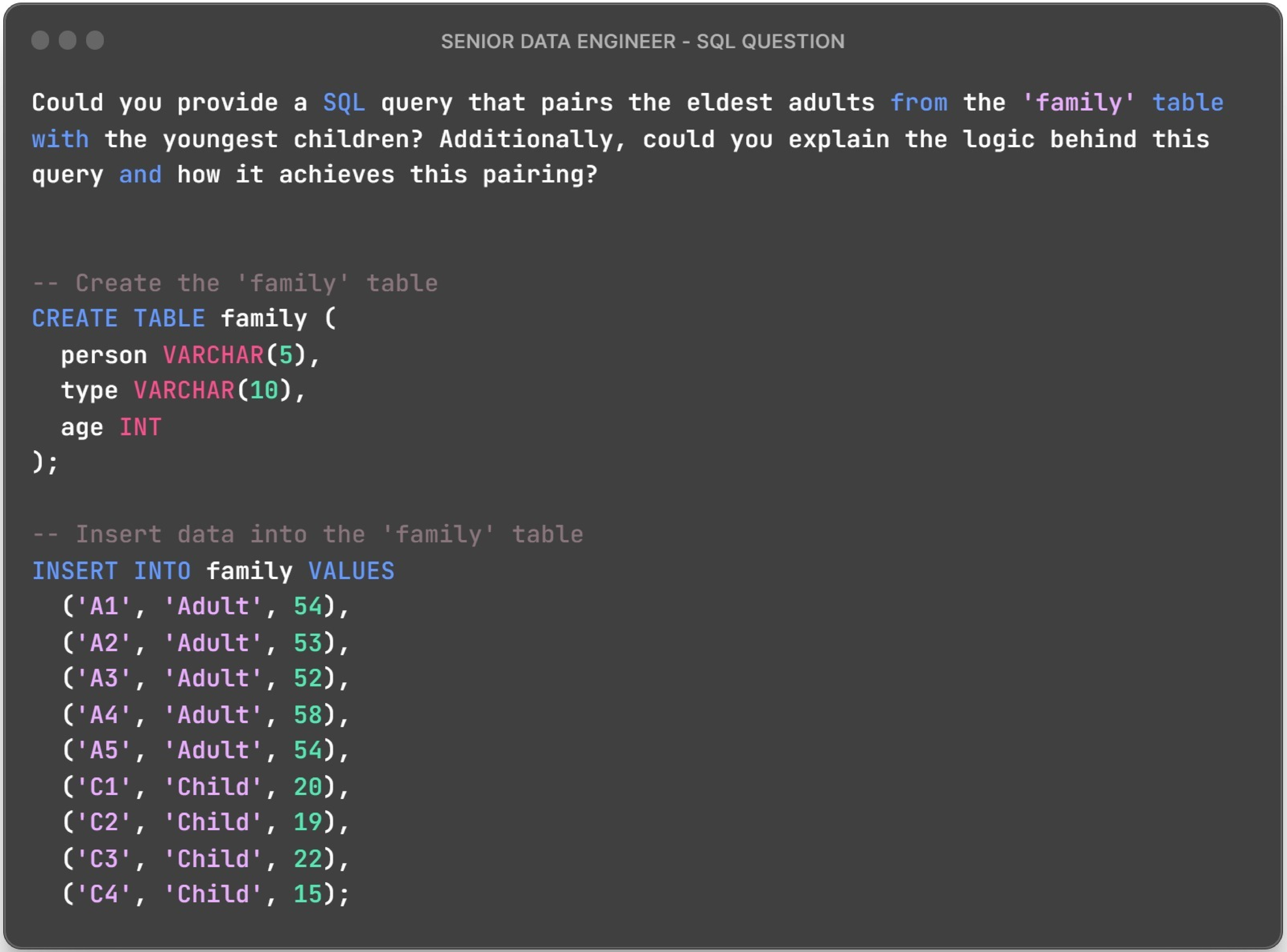
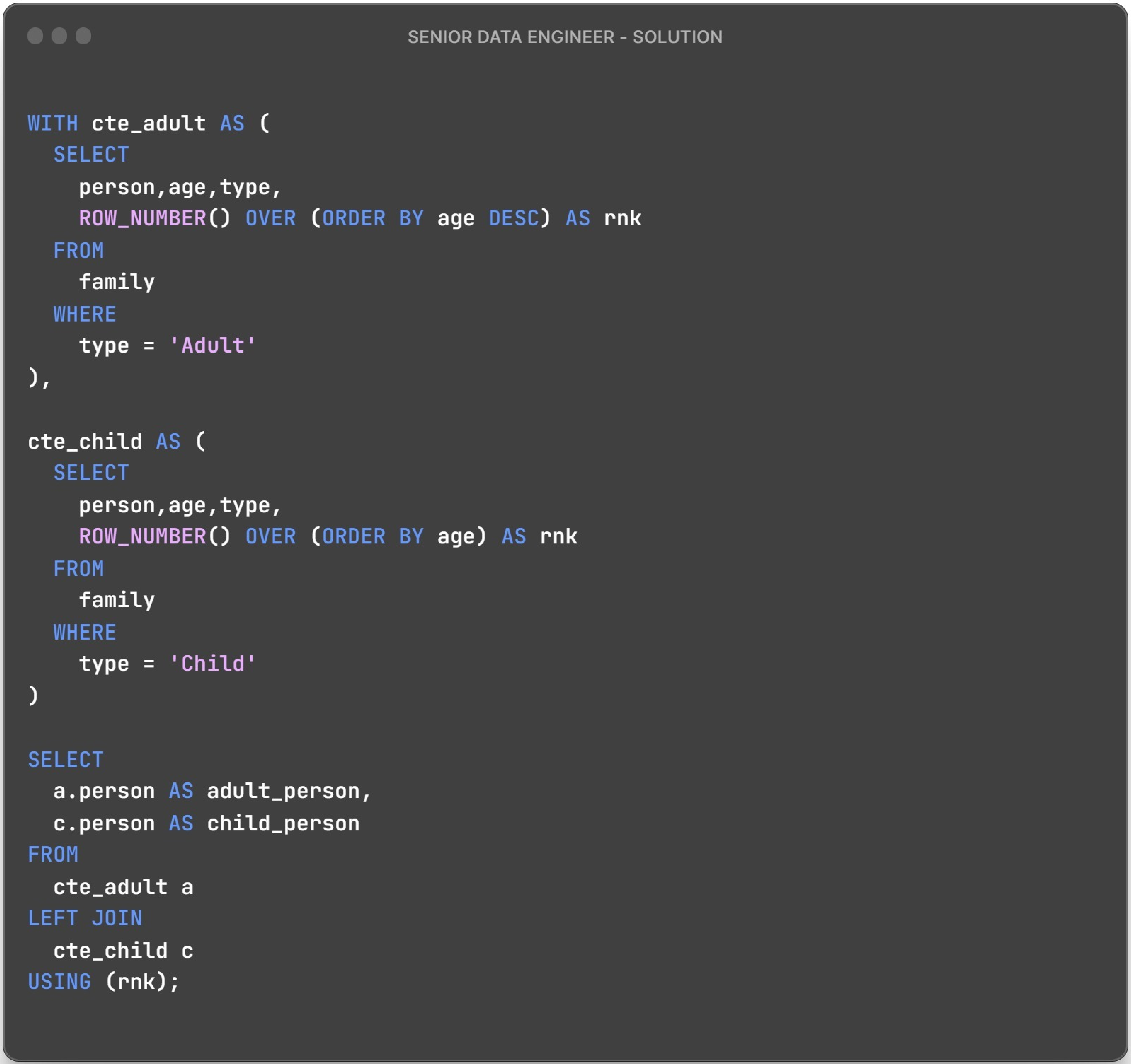
J.P.Morgan

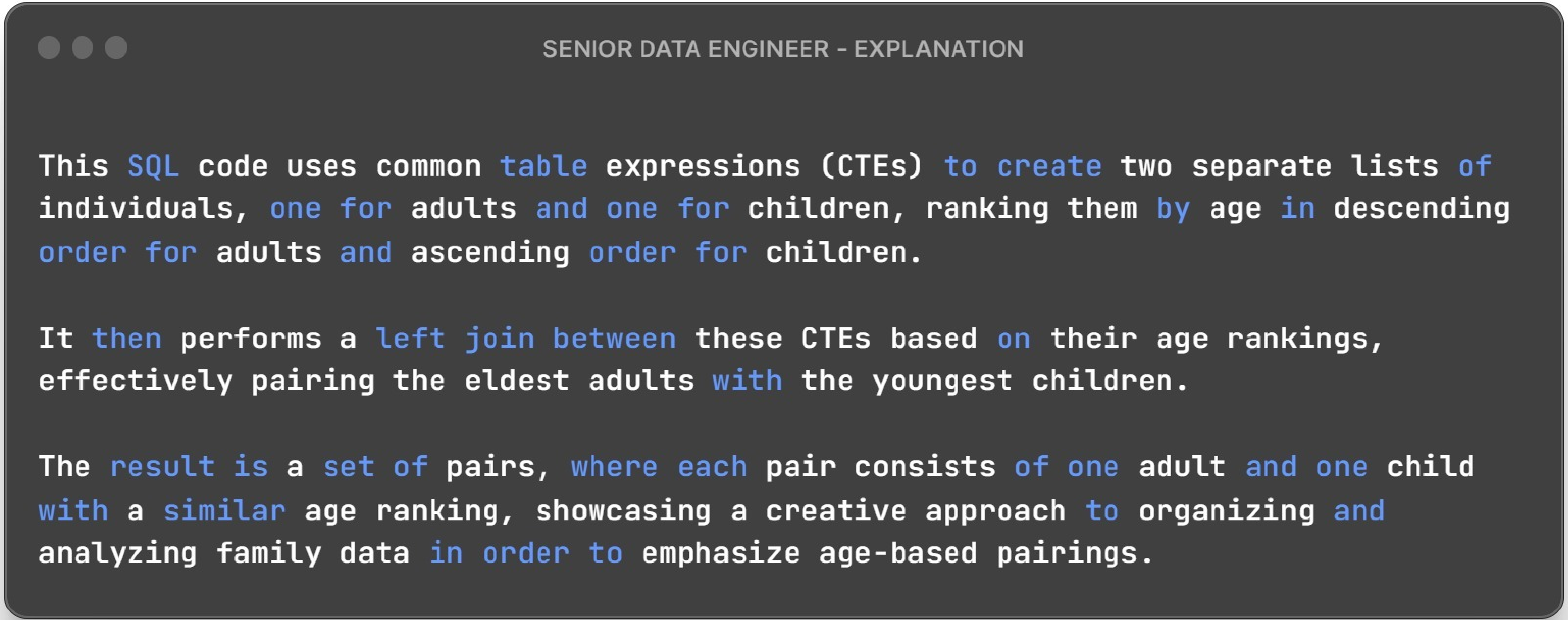

SENIOR DATA ENGINEER - SQL QUESTION

Could you provide a SQL query that pairs the eldest adults from the 'family' table with the youngest children? Additionally, could you explain the logic behind this query and how it achieves this pairing?

```sql
-- Create the 'family' table
CREATE TABLE family (
  person VARCHAR(5),
  type VARCHAR(10),
  age INT
);

-- Insert data into the 'family' table
INSERT INTO family VALUES
  ('A1', 'Adult', 54),
  ('A2', 'Adult', 53),
  ('A3', 'Adult', 52),
  ('A4', 'Adult', 58),
  ('A5', 'Adult', 54),
  ('C1', 'Child', 20),
  ('C2', 'Child', 19),
  ('C3', 'Child', 22),
  ('C4', 'Child', 15);
```


SENIOR DATA ENGINEER - SOLUTION

```sql
WITH cte_adult AS (
  SELECT
    person,age,type,
    ROW_NUMBER() OVER (ORDER BY age DESC) AS rnk
  FROM
    family
  WHERE
    type = 'Adult'
),

cte_child AS (
  SELECT
    person,age,type,
    ROW_NUMBER() OVER (ORDER BY age) AS rnk
  FROM
    family
  WHERE
    type = 'Child'
)

SELECT
  a.person AS adult_person,
  c.person AS child_person
FROM
  cte_adult a
LEFT JOIN
  cte_child c
USING (rnk);
```

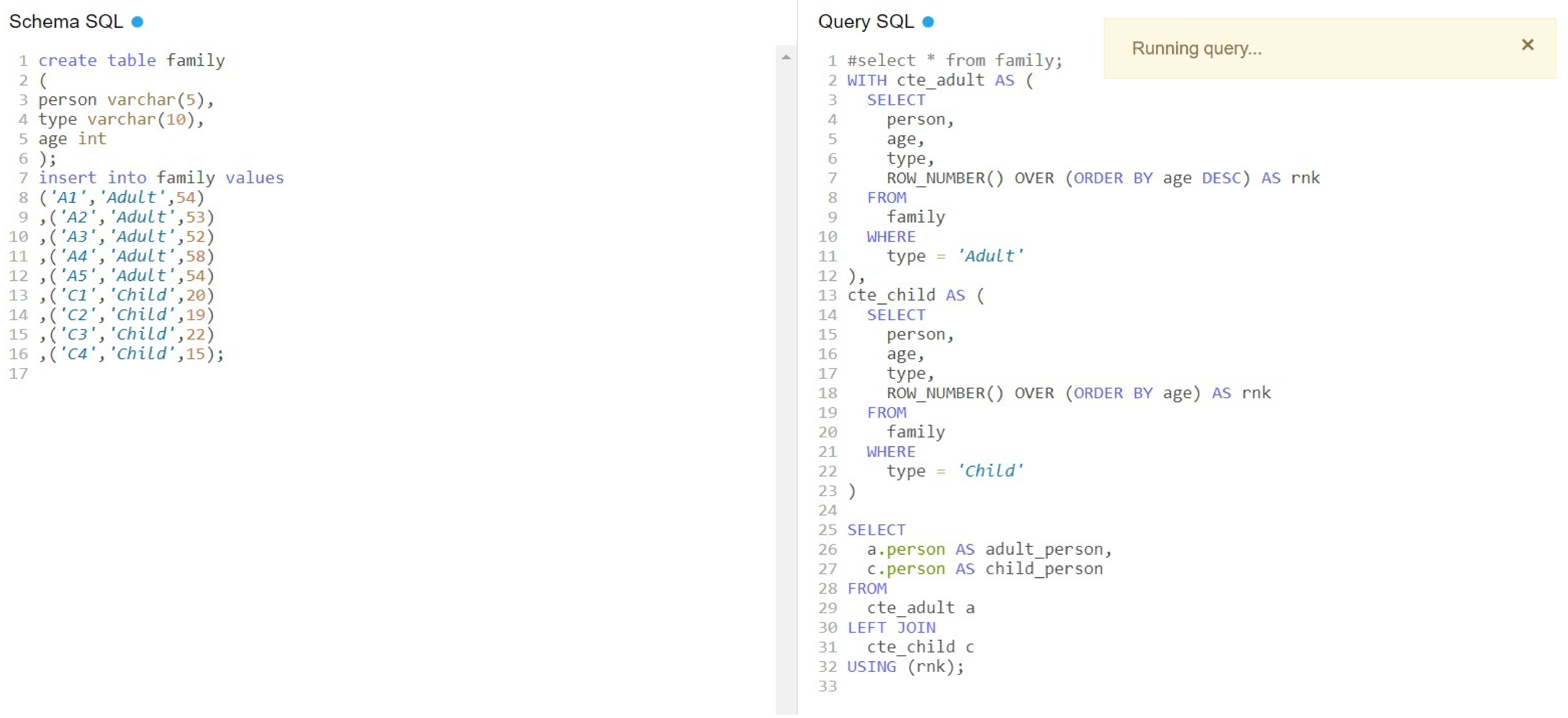


SENIOR DATA ENGINEER - EXPLANATION

This SQL code uses common table expressions (CTEs) to create two separate lists of individuals, one for adults and one for children, ranking them by age in descending order for adults and ascending order for children.

It then performs a left join between these CTEs based on their age rankings, effectively pairing the eldest adults with the youngest children.

The result is a set of pairs, where each pair consists of one adult and one child with a similar age ranking, showcasing a creative approach to organizing and analyzing family data in order to emphasize age-based pairings.

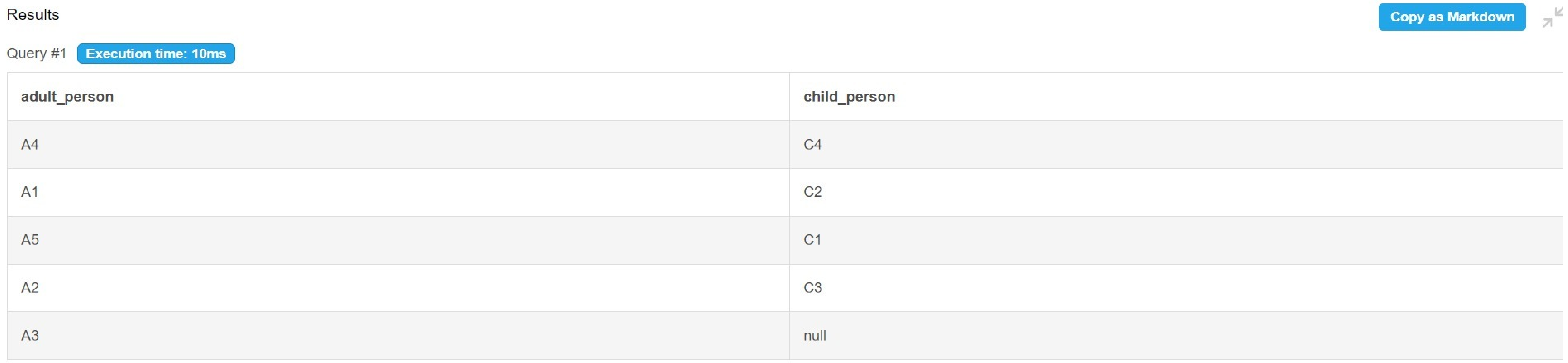

Schema SQL

```
create table family
(
person varchar(5),
type varchar(10),
age int
);
insert into family values
('A1','Adult',54)
,('A2','Adult',53)
,('A3','Adult',52)
,('A4','Adult',58)
,('A5','Adult',54)
,('C1','Child',20)
,('C2','Child',19)
,('C3','Child',22)
,('C4','Child',15);
```

Query SQL

Running query...

```
#select * from family;
WITH cte_adult AS (
  SELECT
    person,
    age,
    type,
    ROW_NUMBER() OVER (ORDER BY age DESC) AS rnk
  FROM
    family
  WHERE
    type = 'Adult'
),
cte_child AS (
  SELECT
    person,
    age,
    type,
    ROW_NUMBER() OVER (ORDER BY age) AS rnk
  FROM
    family
  WHERE
    type = 'Child'
)

SELECT
  a.person AS adult_person,
  c.person AS child_person
FROM
  cte_adult a
LEFT JOIN
  cte_child c
USING (rnk);
```



Results

Query #1 Execution time: 10ms

| adult_person | child_person |
|---|---|
| A4 | C4 |
| A1 | C2 |
| A5 | C1 |
| A2 | C3 |
| A3 | null |