

# SQL Subqueries

# What are subqueries?

Nest queries for advanced filtering

Nested queries are called subqueries

Use SQL logical operators: EXISTS, ANY, ALL, IN, NOT IN

# EXISTS

Used between WHERE clause and subquery

Returns Boolean value:

True if subquery returns at least 1 row

False if subquery returns 0 rows

```
SELECT select_list  
FROM table_name  
WHERE EXISTS (nested_query);
```

# EXISTS example

"Find all authors in amazon\_best\_sellers table who also appear in the president\_books table"

Note: Table alias in outer query can be used in inner query

year [PK] integer	title text	author text
2021	The Four Winds	Kristin Hannah
2020	A Promised Land	Barack Obama
2019	Where the Crawdads Sing	Delia Owens
2018	Becoming	Michelle Obama
2017	Wonder	R. J. Palacio

	id [PK] integer	title text	president text	year integer
1	1	Promises to Keep	Joe Biden	2008
2	2	A Promised Land	Barack Obama	2020
3	3	The Art of the Deal	Donald Trump	1987
4	4	Dreams From My Father	Barack Obama	1995
5	5	How to Get Rich	Donald Trump	2004
6	6	Decision Points	George W. Bush	2010
7	7	Between Hope and History	Bill Clinton	1996

# EXISTS example

"Find all authors in amazon\_best\_sellers table who also appear in the president\_books table"

Note: Table alias in outer query can be used in inner query

```
SELECT a.author
FROM amazon_best_sellers a
WHERE EXISTS(
    SELECT * FROM president_books b WHERE b.president = a.author
);
```


# EXISTS example

"Find all authors in amazon\_best\_sellers table who also appear in the president\_books table"

Note: Table alias in outer query can be used in inner query

```
SELECT a.author
FROM amazon_best_sellers a
WHERE EXISTS(
    SELECT * FROM president_books b WHERE b.president = a.author
);
```



	author text	
1	Barack Obama	

# IN, NOT IN, ANY, ALL

Compare some value to subquery's result set

```
SELECT select_list
FROM table_name
WHERE value
[IN | NOT IN | conditional ALL | conditional ANY ] (nested_query);
```

Recall:

IN operator checks against list of values

List of values may be hard-coded if known ahead of time

```
SELECT b.title, b.year FROM books b
WHERE b.year IN (1990, 1986, 1996, 1998);
```




# IN example

List of values can be made dynamic using result of subquery

"Find all authors in amazon\_best\_sellers table who also appear in the president\_books table"

```
SELECT a.author
FROM amazon_best_sellers a
WHERE a.author IN (
    SELECT president FROM president_books
);
```



	author text	
1	Barack Obama	


# NOT IN example

Negates result of IN operator

"Find all authors in amazon\_best\_sellers who **do not** appear in the president\_books table"

```
SELECT a.author
FROM amazon_best_sellers a
WHERE a.author NOT IN (
    SELECT president FROM president_books
);
```



	author text	
1	Kristin Hannah	
2	Delia Owens	
3	Michelle Obama	
4	R. J. Palacio	

# ANY and ALL

Work in conjunction with comparison operators:  $<$ ,  $<=$ ,  $=$ ,  $\neq$ ,  $>$ ,  $>=$

Filter initial query results based on comparison with subquery


# ANY example

Example: "Find all amazon\_best\_sellers titles and years for books that were published before at least 1 president's book"

Checks against each value in subquery result set and returns true if any comparison evaluates as true, otherwise false

If true, result (from initial query) is added to final result set

```
SELECT a.title, a.year
FROM amazon_best_sellers a
WHERE a.year < ANY (
    SELECT b.year FROM president_books b
);
```



	title text	year [PK] integer
1	Where the Crawdads Sing	2019
2	Becoming	2018
3	Wonder	2017

# ALL example

Example: "Find all amazon\_best\_sellers authors whose first names are alphabetically greater than the first names of every president in president\_books"

Checks against each value in list and returns true only if every check evaluates as true, otherwise false

If true, result (from initial query) is added to final result set

```
SELECT a.author FROM amazon_best_sellers a
WHERE a.author > ALL (
    SELECT b.president FROM president_books b
);
```



	author text	🔒
1	Kristin Hannah	
2	Michelle Obama	
3	R. J. Palacio	