

Recently Asked Data Analyst interview questions

SQL

1. How do you write a query to find duplicate rows in a table?

Answer: To find duplicate rows in a table, we use the GROUP BY clause and the HAVING keyword:

```
SELECT column1, column2, COUNT(*) AS count
```

```
FROM table_name
```

```
GROUP BY column1, column2
```

```
HAVING COUNT(*) > 1;
```

This query groups the data by relevant columns and filters those with a count greater than one, indicating duplicates.

2. How would you perform a left join and filter out nulls in SQL?

Answer: A LEFT JOIN retains all records from the left table and matches records from the right table. To remove nulls from the right table, use a WHERE clause:

```
SELECT a.*, b.*
```

```
FROM table1 a
```

```
LEFT JOIN table2 b ON a.id = b.id
```

```
WHERE b.id IS NOT NULL;
```

This ensures only matching records are retrieved.

3. What is a window function in SQL, and how do you use it for ranking data?

Answer: Window functions perform calculations across a subset of rows related to the current row without collapsing data. The RANK() function assigns ranks based on a specified column:

```
SELECT id, name, salary, RANK() OVER (PARTITION BY department ORDER BY salary DESC) AS rank
```

```
FROM employees;
```

This ranks employees within each department based on salary.

Python

1. How do you import a CSV file into a pandas DataFrame and handle missing data?

Answer: Import pandas as pd

```
Df = pd.read_csv('data.csv') # Import CSV
```

```
Df.fillna(0, inplace=True) # Replace missing values with 0
```

```
Df.dropna(inplace=True) # Drop rows with missing values
```

This ensures missing values are handled efficiently.

2. How do you use list comprehensions to filter and transform data?

Answer: Numbers = [1, 2, 3, 4, 5, 6]

```
Squared_evens = [x**2 for x in numbers if x % 2 == 0]
```

```
Print(squared_evens) # Output: [4, 16, 36]
```

List comprehensions efficiently filter and transform lists in a single line.

3. What are the differences between the apply() and map() functions in pandas?

Answer: Map() applies a function element-wise to a Series:

```
Df['column'] = df['column'].map(lambda x: x * 2)
```

Apply() works on both Series and DataFrames, allowing more flexibility:

```
Df['column'] = df['column'].apply(lambda x: x * 2)
```

```
Df.apply(lambda row: row['A'] + row['B'], axis=1) # Works row-wise
```

Excel

1. How would you use VLOOKUP or XLOOKUP to merge data between two Excel sheets?

Answer: VLOOKUP searches for a value in the first column and returns a value from a specified column:

=VLOOKUP(A2, Sheet2!A:B, 2, FALSE)

XLOOKUP (available in Excel 2019+) offers more flexibility:

=XLOOKUP(A2, Sheet2!A:A, Sheet2!B:B, "Not Found")

2. What is the difference between absolute and relative cell references, and when would you use each?

Answer: Relative references (A1) adjust dynamically when copied across cells.

Absolute references (\$A\$1) remain fixed when copied.

Use absolute references for fixed lookup values and relative references for formulas that need to adjust dynamically.

3. How do you create a pivot table and analyze data with it?

1. Select data → Insert → PivotTable.
2. Drag fields into Rows, Columns, Values, and Filters sections.
3. Use "Value Field Settings" to calculate SUM, COUNT, or AVERAGE.

Power BI

1. How would you create and customize a calculated column in Power BI?

Answer: Go to "Modeling" → "New Column" and use DAX:

TotalRevenue = Sales[Quantity] * Sales[UnitPrice]

Calculated columns store results in the data model and update when refreshed.

2. What is the difference between a slicer and a filter in Power BI, and when would you use each?

Answer: Slicers are visual elements that allow users to interactively filter data on a report.

Filters are applied at the report, page, or visual level in the “Filter Pane.”

Use slicers for interactive dashboards and filters for static data refinement.

3. How do you create relationships between tables in Power BI, and how do they impact your data model?

Answer :

1. Go to “Model View” → Drag and connect related columns.
2. Choose “One-to-Many” or “Many-to-Many” relationships.
3. Relationships enable efficient joins, maintaining data consistency.