**INTERVIWE PREPS : SQL**

**DATA TYPES**

**Top SQL Data Type Interview Questions (With Answers) OF sql Data types**

**🔹 1. What is the difference between CHAR, VARCHAR, and TEXT?**

**Answer:**

* CHAR(n) stores **fixed-length** strings; padded with spaces.
* VARCHAR(n) stores **variable-length** strings; no padding.
* TEXT stores **large blocks** of text (up to GBs); can't be indexed or sorted easily.

✅ Use TEXT for long comments/articles, VARCHAR for names/emails, and CHAR for fixed-length codes like country codes or PINs.

**🔹 2. Which data type would you use to store a mobile number? Why?**

**Answer:**  
Use a VARCHAR(15) instead of an INT:

* Phone numbers are not used in arithmetic operations.
* Leading zeros (like in 0987654321) are **important** and would be lost in INT.
* Might contain +91, -, or parentheses in international formats.

**🔹 3. Why not use TEXT instead of VARCHAR all the time?**

**Answer:**

* TEXT is not indexable or sortable in many databases (like MySQL).
* Functions like LENGTH(), SUBSTRING() might not work well.
* TEXT is stored outside the main table — **slower performance** for frequent access.

**🔹 4. Difference between INT, FLOAT, and DECIMAL?**

**Answer:**

* INT: Whole numbers, no decimals (e.g., 1, 1000)
* FLOAT: Approximate decimal (e.g., 3.14159) — **not precise**
* DECIMAL(p, s): Exact decimal, good for money (e.g., DECIMAL(10,2) → 12345678.90)

✅ Use DECIMAL for prices, FLOAT for scientific data, INT for counts.

**🔹 5. What is the difference between DATE, DATETIME, and TIMESTAMP?**

**Answer:**

* DATE: Only stores year-month-day (YYYY-MM-DD)
* DATETIME: Stores date and time (YYYY-MM-DD HH:MM:SS)
* TIMESTAMP: Same format as DATETIME, but auto-converts to/from UTC.

**🔹 6. What’s the max length of VARCHAR?**

**Answer:**

* In MySQL: up to **65,535 bytes** (but practically limited by row size and character encoding).
* In PostgreSQL: effectively unlimited.
* Always define a reasonable length (like VARCHAR(255)) for performance.

**🔹 7. What data type would you use for storing TRUE/FALSE values?**

**Answer:**  
Use BOOLEAN or BIT(1). It internally maps to TINYINT(1) in MySQL (0 = false, 1 = true).

**🔹 8. Can a VARCHAR column be indexed?**

**Answer:**  
Yes, but:

* Indexing large VARCHAR fields (e.g., >255 chars) can be slow.
* Some DBs require prefix indexes: INDEX(col(100))

✅ Better to index short, unique columns (like email, username).

**🔹 9. What happens if you insert a string longer than the defined VARCHAR(n)?**

**Answer:**

* It depends on the DB:
  + In **MySQL**, it gets **truncated** (with a warning).
  + In **PostgreSQL**, it **throws an error**.

**🔹 10. What’s the difference between NULL and empty string ('') in CHAR/VARCHAR?**

**Answer:**

* NULL: Means **no value** (unknown or not applicable).
* '': An **empty** string — valid but has no characters.

✅ In interviews: Explain that NULL affects COUNT, AVG, and must be handled using IS NULL.

**💡 Bonus Tip:**

**Always** explain your **reasoning** for choosing data types — interviewers are testing your design thinking and understanding of storage and performance trade-offs.

**ALIAS**

**💬 Common Interview Questions:**

**🔸 1. Why do we use aliases in SQL?**

**Answer**: To simplify long table or column names, improve query readability, and present user-friendly column headers.

**🔸 2. Can we use a column alias in the WHERE clause?**

**Answer**: No — aliases in SELECT cannot be used in WHERE. WHERE executes **before** SELECT.

You **can use an alias in WHERE** **only if** that alias is **not used in the WHERE condition itself**.

**Why this works:**

**SELECT salary AS sal**

**FROM employee**

**WHERE gender = 'male';**

**This works because sal is NOT used in the WHERE clause — you're just giving salary a new name after filtering.**

**Here:**

* **WHERE gender = 'male' executes before the alias sal is even defined.**
* **So this works fine since alias is not involved in the WHERE.**

**❌ But this will NOT work:**

**SELECT salary AS sal**

**FROM employee**

**WHERE sal > 50000; -- ❌ Error**

**Why this fails:**

* **The alias sal is defined in the SELECT stage**
* **But WHERE is executed before SELECT**
* **So the SQL engine doesn't yet know what sal is**

✅ You can use aliases in ORDER BY or **subqueries**, though:

select salary as SAL, gender as'M/F'from employee where country = 'india' order by SAL

**🔸 3. Are table aliases mandatory?**

**Answer**: No, but they are **very helpful** for complex queries with multiple tables.