

**NULL** is used to represent a missing value, but that it usually has one of three different interpretations

1. **Unknown value:** Value exists but is not known, or it is not known whether or not the value exists.
  - Date of birth: Sometimes date of birth is not known (however, it exists), so it is represented by **NULL**.
  - Home phone: Sometimes it is not known whether or not the person has a home phone.
2. **Unavailable or withheld value:** Value exists but is purposely withheld.
  - A person has a home phone but does not want it to be listed, so it is withheld and represented as **NULL**.
3. **Not applicable attribute:** The attribute does not apply to a particular tuple or is undefined for that tuple.
  - An attribute `LastCollegeDegree` would be **NULL** for a person who has no college degrees because it does not apply to that person.

It is often not possible to determine which of the meanings is intended; for example, a **NULL** for the home phone of a person can have any of the three meanings. Hence, SQL does not distinguish among the different meanings of **NULL**.

Null values present special problems in –

1. **Arithmetic operations:** The result of an arithmetic expression (involving, for example, +, −, \*, or /) is **NULL** if any of the input values is **NULL**.
2. **Comparison operations:** The result of any comparison involving a **NULL** value is treated as **UNKNOWN** (other than predicates **IS NULL** and **IS NOT NULL**).
3. **Boolean operations:**

- **AND:**

<b>AND</b>	TRUE	FALSE	UNKNOWN
TRUE	TRUE	FALSE	UNKNOWN
FALSE	FALSE	FALSE	FALSE
UNKNOWN	UNKNOWN	FALSE	UNKNOWN

- **OR:**

<b>OR</b>	TRUE	FALSE	UNKNOWN
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	UNKNOWN
UNKNOWN	TRUE	UNKNOWN	UNKNOWN

- **NOT:**

<b>NOT</b>	
TRUE	FALSE
FALSE	TRUE
UNKNOWN	UNKNOWN

If the **WHERE** clause predicate evaluates to either false or unknown for a tuple, that tuple is not added to the result.