Congratulations! You passed!
TO PASS 80% or higher

Keep Learning

Retake the assignment in **7h 41m**

GRADE
90%

Week 3 Core Quiz

LATEST SUBMISSION GRADE
90%

1. For which of these tasks would you need to use a **WHERE** clause? Check all that apply. **1 / 1 point**

- [x] For a table that includes which of many offices each employee works, finding all the employees in the Chicago office

Correct
Correct. A **WHERE** clause can look for rows that show the condition that the employee's office is Chicago.

- [ ] For a table of web logs, which show the IP addresses of every visit, removing rows with duplicate IP addresses
- [x] For a table of inventory items, including quantity and price, finding all inventory items priced under $5

Correct
Correct. A **WHERE** clause can look for rows that show the condition that the price is less than $5.

- [ ] For a table of pets, including their owners and ages, finding the range of values in their ages

2. The following query will fail: **1 / 1 point**

SELECT name, shop, aisle FROM fun.inventory WHERE price + 5;

What are the issues with this query? Check all that apply.

- [x] The expression in the **WHERE** clause must evaluate to a Boolean value

Correct
Correct. The expression **price + 5** returns a number, not a true or false value, so the engine be able to choose whether to include a row or not.

- [ ] The table reference in the **FROM** clause cannot have a dot (**.**) in the name.
- [ ] The column in the **WHERE** clause is not in the **SELECT** list
- [ ] The expression in the **WHERE** clause must be in the **SELECT** list

3. Write and run a query on **wax.crayons** to find colors with **205** as the **red** value. Which of the following colors are returned? Check all that apply. **1 / 1 point**

- [ ] Almond
- [x] Antique Brass

Correct
Correct. The **red** value is **205**.

- [ ] Atomic Tangerine
- [ ] Banana Mania
- [x] Mahogany

Correct
Correct. The **red** value is **205**.

- [x] Silver

Correct
Correct. The **red** value is **205**.

- [ ] Tan
- [x] Wisteria

Correct
Correct. The **red** value is **205**.

4. Select the expressions that are equivalent to **x != 2** in SQL. Check all that apply. **1 / 1 point**

- [x] x < 2 OR x > 2

Correct
Correct. If **x** is not equal to **2**, it must be either less than **2** or greater than **2**.

- [x] NOT x = 2

Correct
Correct. **NOT** negates the value returned by **x = 2**, so this expression is **true** when **x = 2** is **false** and **false** when **x = 2** is **true**.

- [ ] x < 2 OR 2 > x
- [x] x <> 2

Correct
Correct. **<>** is a valid alternative comparison operator that means "not equal to."

- [ ] x NOT = 2
- [ ] x < 2 AND x > 2

5. The table **table_name** includes the following row: **1 / 1 point**

id	bool1	bool2	bool3
34	true	false	true

Which of the following would include that row in the result set? Check all that apply.

- [ ] SELECT * FROM table_name WHERE bool1 AND NOT (bool2 OR bool3)
- [x] SELECT * FROM table_name WHERE NOT bool2 AND bool3

Correct
Correct. This first evaluates **NOT bool2**, which is **true**; then it compares **true AND bool3**, so the expression is **true**. The row will be included.

- [ ] SELECT * FROM table_name WHERE NOT bool1 OR bool2 AND bool3
- [x] SELECT * FROM table_name WHERE bool1 AND bool2 OR bool3

Correct
Correct. This first evaluates **bool1 AND bool2** to be **false**, then compare **false OR bool3**. Since **bool3** is true, the expression is **true** and the row will be included.

- [x] SELECT * FROM table_name WHERE NOT (bool1 AND bool2)

Correct
Correct. This first evaluates the * *AND* in parentheses, which is **false**, and then the **NOT** negates that, giving a final evaluation of **true**. The row will be included.

6. Which of the following would provide results that include a row with **int_x=-25**? Check all that apply. **1 / 1 point**

- [ ] SELECT * FROM table_name WHERE int_x BETWEEN (-50, 0);
- [ ] SELECT * FROM table_name WHERE int_x IN (-50, 0);
- [x] SELECT * FROM table_name WHERE int_x BETWEEN -50 AND 0;

✓ **Correct**
Correct. Since -50 ≤ -25 ≤ 0, -25 is **BETWEEN -50 AND 0**.

- [x] SELECT * FROM table_name WHERE int_x NOT IN (-50, 0);

✓ **Correct**
Correct. Since -25 is neither -50 nor 0, -25 is **NOT IN (-50, 0)**.

- [ ] SELECT * FROM table_name WHERE int_x NOT BETWEEN (-50, 0);
- [ ] SELECT * FROM table_name WHERE int_x IN -50 AND 0;
- [ ] SELECT * FROM table_name WHERE int_x NOT BETWEEN -50 AND 0;
- [ ] SELECT * FROM table_name WHERE int_x NOT IN -50 AND 0;

7. The following shows just a few rows from a table for students in a school. (GPA is grade point average, where 4.0 means the student is getting the highest scores possible. Absences is how many days the student has not attended school, and detention is a punishment for bad behavior.) **1 / 1 point**

students

id	name	age	gpa	absences	detentions
930	Olufunmilayo Ayton	16	4.00	3	2
667	Vincent Michaelson	15	2.53	12	0
907	Asa Quigg	15	3.57	1	0
168	Kiran Patil	17	3.28	0	3
368	Amaal Al-Amin	16	4.00	NULL	2

Check all the students whose rows would be included in the results of

SELECT name FROM students WHERE absences < 2;

- [ ] Olufunmilayo Ayton
- [ ] Vincent Michaelson
- [x] Asa Quigg

✓ **Correct**
Correct. This student has 1 absence, and 1 < 2.

- [x] Kiran Patil

✓ **Correct**
Correct. This student has 0 absences, and 0 < 2.

- [ ] Amaal Al-Amin

8. The **offices** table in the **default** database on the VM includes one row with **NULL** in **state_province** column. Which of the following would provide that row in the result? Check all that apply. **1 / 1 point**

- [ ] SELECT * from default.offices WHERE state_province="Santa Fe" AND state_province IS NOT NULL
- [ ] SELECT * from default.offices WHERE state_province IS NOT NULL
- [x] SELECT * from default.offices WHERE state_province IS NULL

✓ **Correct**
Correct. Since **state_province** is **NULL**, the expression in the **WHERE** clause is **true** and the row is included.

- [ ] SELECT * from default.offices WHERE state_province="Santa Fe" OR state_province IS NOT NULL
- [x] SELECT * from default.offices WHERE state_province="Santa Fe" OR state_province IS NULL

✓ **Correct**
Correct. Since state_province is **NULL**, the expression in the **WHERE** clause is * **true** and the row is included.

- [ ] SELECT * from default.offices WHERE state_province="Santa Fe" AND state_province IS NULL

9. The following shows Amaal Al-Amin's data from a table for students in a school. (GPA is grade point average, where 4.0 **1 / 1 point**

9. The following shows Amaal Al-Amin's data from a table for students in a school. (GPA is grade point average, where 4.0 means the student is getting the highest scores possible. Absences is how many days the student has not attended school, and detention is a punishment for bad behavior.) 0 / 1 point

id	name	age	gpa	absences	detentions
368	Amaal Al-Amin	16	4.00	NULL	2

Which of the following **WHERE** clauses would include Amaal's row when used in a **SELECT** query? Check all that apply.

- [ ] WHERE gpa < 3.50 AND absences < 3
- [ ] WHERE absences < 3
- [ ] WHERE NOT absences < 3
- [ ] WHERE gpa > 3.50 AND absences < 3
- [x] WHERE gpa < 3.50 OR absences < 3

! **This should not be selected**
Incorrect. You might want to review the reading "Missing Values with Logical Operators" and pay attention to the conditions under which an **OR** expression would be **NULL**.

- [ ] WHERE gpa > 3.50 OR absences < 3

10. You have a database in which some bad data in the column named **score** is marked with **NULL** and some is marked with the value **-1**. For your purposes, you can do more with values marked **-1**, so you want to replace all **NULL** values in the **score** column with **-1** but otherwise leave the **score** values as they are. Which of the following will do this? Check all that apply. 1 / 1 point

- [x] CASE WHEN score IS NULL THEN -1 ELSE score END

✓ **Correct**
Correct. If **score IS NULL** is **true**, then it uses **-1**; for all other rows, it uses the existing value of **score**.

- [ ] nullif(score, -1)
- [x] ifnull(score, -1) *Note:* For some engines this is **nvl(score, -1)**

✓ **Correct**
Correct. If score is **NULL**, this replaces it with **-1**; otherwise it will use the existing value of **score**.

- [x] if(score IS NULL, -1, score)

✓ **Correct**
Correct. If **score IS NULL** is **true**, then it uses **-1**; for all other rows, it uses the existing value of **score**.

- [ ] CASE WHEN score = -1 THEN NULL ELSE score END
- [ ] if(score = -1, NULL, score)