**Initial Data**

**Table1 – FirstTab**

ID Name

5 Pawan

6 Sharlee

7 Krish

NULL Avtaar

**Table2 – SecondTab**

ID

5

NULL

**Q1**

**Query:**

SELECT COUNT(\*)

FROM FirstTab AS ft

WHERE ft.id NOT IN (SELECT id FROM SecondTab WHERE id IS NULL)

**Assumption:** The subquery selects only NULL from SecondTab, so the main query will check for IDs in FirstTab that are not NULL. The valid IDs in FirstTab are 5, 6, 7, and NULL. Since NULL is not counted, we will have 5, 6, and 7, which makes a total of **3**.

**Expected Output:** 3

**Q2**

**Query:**

SELECT COUNT(\*)

FROM FirstTab AS ft

WHERE ft.id NOT IN (SELECT id FROM SecondTab WHERE id = 5)

**Assumption:** The subquery selects only 5 from SecondTab, so the main query will look for IDs in FirstTab that are not 5. The IDs 6, 7, and NULL are included, making a total of **3**.

**Expected Output:** 3

**Q3**

**Query:**

SELECT COUNT(\*)

FROM FirstTab AS ft

WHERE ft.id NOT IN (SELECT id FROM SecondTab)

**Assumption:** The subquery selects both 5 and NULL from SecondTab. In SQL, when NULL is present in an IN or NOT IN clause, it affects the results. Thus, the query checks for IDs in FirstTab that are not 5 or NULL. Since 6 and 7 remain, we will have **2**.

**Expected Output:** 2

**Q4**

**Query:**

SELECT COUNT(\*)

FROM FirstTab AS ft

WHERE ft.id NOT IN (SELECT id FROM SecondTab WHERE id IS NOT NULL)

**Assumption:** The subquery selects only NULL from SecondTab, so the main query checks for IDs in FirstTab that are not NULL. All IDs except NULL (5, 6, and 7) will be counted, giving us a total of **3**.