not (a ~ b)

```
define "Not Equivalent Clause 1":
    [Condition: "test-vs"] C
        where not (C.clinicalStatus ~ "Condition Active")
```

This requires two checks for full coverage

- One for **not**
- One for **a ~ b**

a !~ b

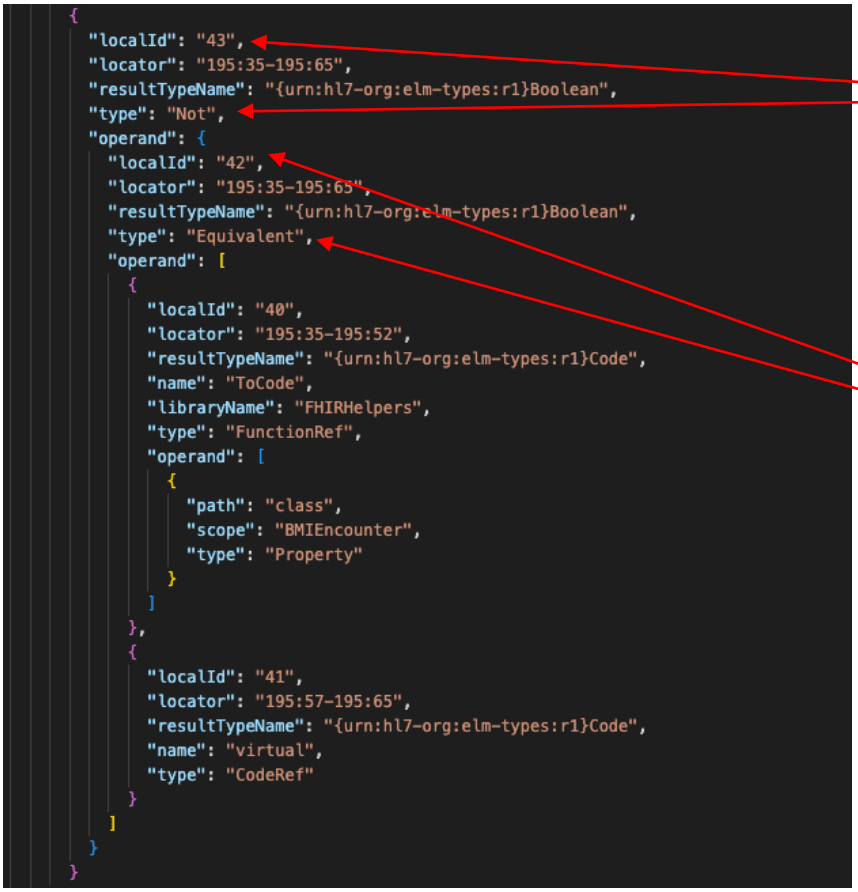
both are translated the same

```
define "Not Equivalent Clause 2":
    [Condition: "test-vs"] C
        where C.clinicalStatus !~ "Condition Active"
```

This ALSO requires those two checks for full coverage, even though that need is effectively hidden

Not Equivalent (!~)

```
BMIEncounter.class !~ "virtual"
```



Not clause gets a localId of 43 that will evaluate to **true** if there is a test case where **BMIEncounter.class is NOT EQUIVALENT** to **“virtual”**

Equivalent clause gets a localId of 42 that will evaluate to **false** unless there is a test case where **BMIEncounter.class IS EQUIVALENT** to **“virtual”**

It was impossible to see this in the highlighting because the **NOT** clause was evaluating to **true** which resulted in the clause being highlighted for coverage.

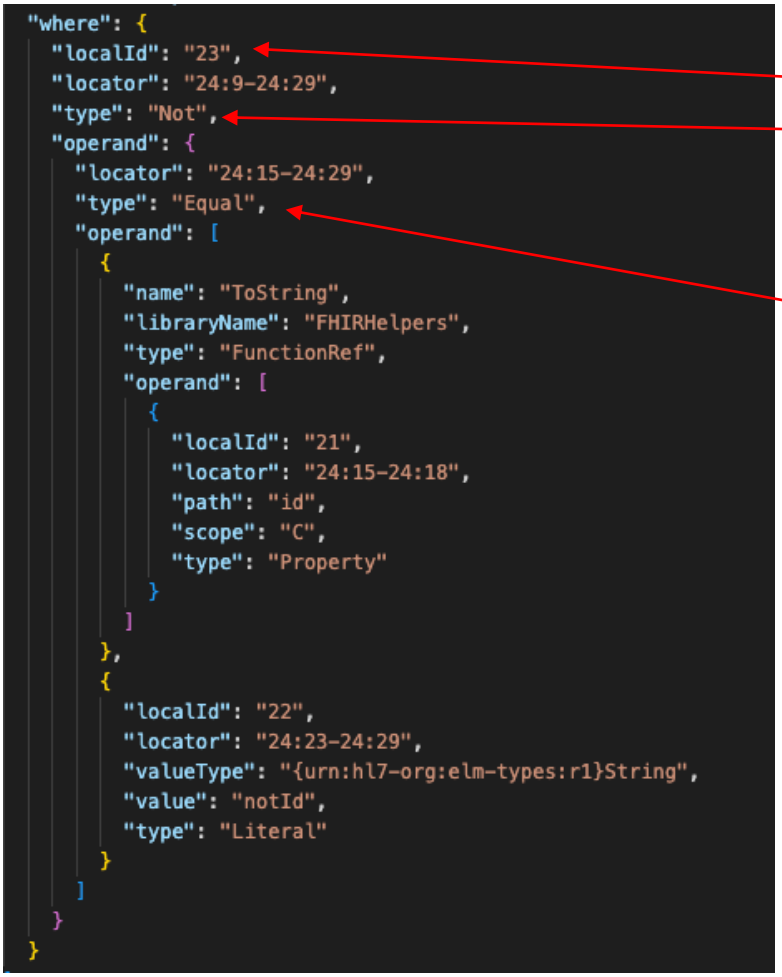
In fqm-execution 1.3.3:

- Highlighting does **NOT** change
- Calculation percentage **DOES** change
    - The Equivalent clause is not counted in the coverage percentage calculation

Not Equal (!=)

We were not seeing any issues with coverage percentage calculation for **Not Equal** because **Equal** does not receive a localId in the current version of the cql-to-elm translator:

```
C.id != 'notId'
```



Not clause gets a localId of 23 that will evaluate to **true** if there is a test case where **C.id is NOT EQUAL** to **"notId"**

Equal clause does **NOT** get a localId; therefore, the clause is not included in coverage percentage calculation

There is currently no issue with **!=** in fqm-execution. However, the lack of a localId on the Equal clause is an issue in the translator that we are anticipating a fix for, so **fqm-execution 1.3.3** accounts for that change whenever it happens.

In fqm-execution 1.3.3 AND with current translator:

- Highlighting does **NOT** change
- Calculation percentage does **NOT** change