Boolean Queries

Queries · https://namazustudios.com/docs/namazu-elements-core/queries/boolean-queries/

Combining Base Queries to Form Complex Queries

In addition to Base Query Syntax, it is possible to mix and match multiple queries to perform complex lookups of objects in the database. The following operators allow for greater complexity of finding objects within Elements:

- `AND` - Logical all-inclusive operation. Both sides of the AND operation must be true for the object to match.
- `OR` - Logical operation which will match if the object matches either the left or right side of the operator
- `NOT` - Logical negation. The operator following NOT must be false to match the object in the query.
- `( )` - Grouping Operator. Groups operations such that they are evaluated before the groups on the outside. Used to force order of operations or provide clarity where the syntax may be ambiguous.

Example: Finding Legendary Swords

With complex Boolean queries, it is possible to perform advanced grouping of data. For example, it may be possible to search for all distinct items with certain metadata fields.

In this case, we can query the whole database for legendary swords:

```
metadata.type:Sword AND metadata.rarity:Legendary
```

This will match all objects which match the following criteria:

- Have a metadata string property named "Type"
- Have a metadata string property named "Rarity"

- Type equals the word "Sword"
- Rarity equals the word, "Legendary"

Example: Finding Legendary Low-Level Swords

We can further refine the query to include stats on the item itself:

```
metadata.type:Sword AND metadata.rarity:Legendary AND metadata.level:10 TO 20
```

This will match all objects which match the following criteria:

- Have a metadata string property named "Type"
- Have a metadata string property named "Rarity"
- Type equals the word "Sword"
- Rarity equals the word, "Legendary"
- The object's level is between Level 10 and Level 20.