

# Handy Grid Filters

Below is a list of Regular Expressions (a.k.a. Regex) that allow you to dynamically query for data in any Onit grid.

## Filtering for Multiple Values

To search for multiple values use the following syntax:

```
# (Pending|Paid)
```

This filter would find all records that have values like "**Pending Review**", "**Pending SAP**", "**Paid**", etc.

## Excluding Records

To exclude records you can prepend your filter with "!".

```
!Pending
```

This filter would find all records except for those containing the word "**Pending**".

To exclude *multiple* values use:

```
#! (Pending|Paid)
```

This filter would find all records except for those containing the words "**Pending**" and "**Paid**".

## Filtering for records where a Field's value is blank

To find records whose Field value is blank, missing, or empty, use the following:

```
#! .
```

If you used this filter on an **Expiration Date** column, you would get all the records that didn't have an **Expiration Date** value.

## Filtering for records where a Field isn't blank

Alternatively, you may want to find all the records where a Field's value *isn't* blank.

```
#! ^$
```

If you used this filter on an **Expiration Date** column, you would get only the records that *did* have an **Expiration Date** value.

## Find Exact Match

To find an exact match of the search criteria use the following:

```
#"Closed"
```

If you used this filter on a **Phase** column, you would get only the Records in the Closed phase.

## Matching Email Addresses

Use the following Regex expression to match email addresses for validation purposes:

```
/^([a-z0-9_\. -]+)@([\da-z_\. -]+\.)([a-z_\.]{2,5})$/
```

If you used this Regex as a validator it would match all popular email address formats such as `name@domain.com` or `firstname.lastname@domain.com`.

## Matching a Phone Number

Use the following Regex expressions to match phone numbers for validation purposes:

```
/^\b\d{3}[-.]?\d{3}[-.]?\d{4}\b$/
```

If you used this Regex as a validator it would match phone numbers with the format `123-123-1234`, `123.123.1234`, or `1231231234`.

## Matching a Date

Use the following Regex expression to match a date:

```
^\d{1,2}\/\d{1,2}\/\d{4}$
```

If you used this Regex as a validator it would match dates in either the `dd/mm/yyyy`, `d/m/yyyy`, or `mm/dd/yyyy` format.

## Matching a Positive Number

Use the following Regex to match any positive number:

```
/^\d*\.\?\d+$/
```

For example, this Regex could be used as a validation on Budget Records to make sure all Budgets are positive numbers.

## Matching a Zip Code

Use this Regex to match zip codes:

```
/ (^\\d{5} (-\\d{4})? $) | (^ [ABCEGHJKLMNPRSTVXY] {1} \\d{1} [A-Z] {1} * \\d{1} [A-Z] {1} \\d{1} $) /
```

This Regex could be used to validate zip codes entered into employee profiles.