How to: An ELN+LIMS built in Notion + Airtable

This document provides more details on how to set up this sort of lab notebook. Check out the accompanying blog post: An AI-ready lab notebook for life science doesn't need to be complex or expensive

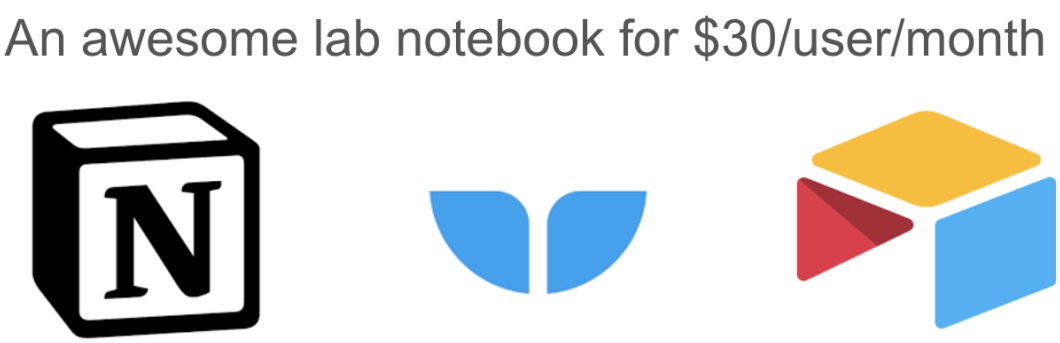


Table of contents

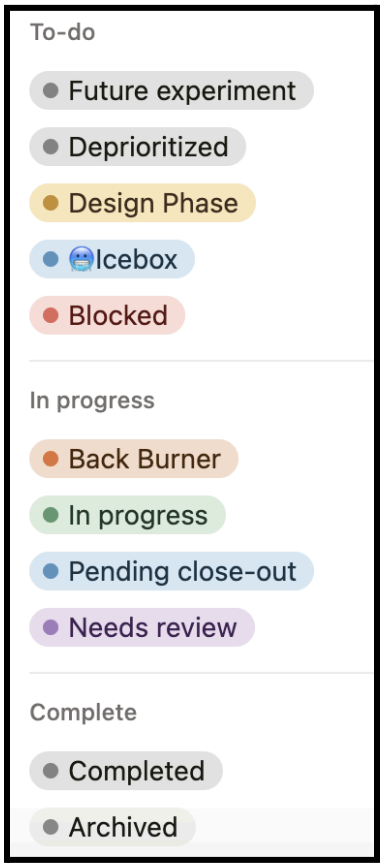
Notion updates

We started with ELeveNote's ELN template and made customizations. Here's a couple we'd recommend to others:

The QC expression we use is a little different than the default one in ELevenNote

```
ifs(
  empty( Aa Name ), "❌ no name specified",
  empty( ↗ Linked Project ), "❌ no project specified",
  empty( Experimenters ), "❌ no experimenters specified",
  empty( Objective ), "❌ no objective specified",
  empty( Measure of success ), "❌ no measure of success specified",
  empty( Conclusions ), "⏳🧪 no conclusions specified",
  empty( Reviewer ), "❌ no reviewer specified",
  "✅"
  )
```

Our current set of expanded status tags:

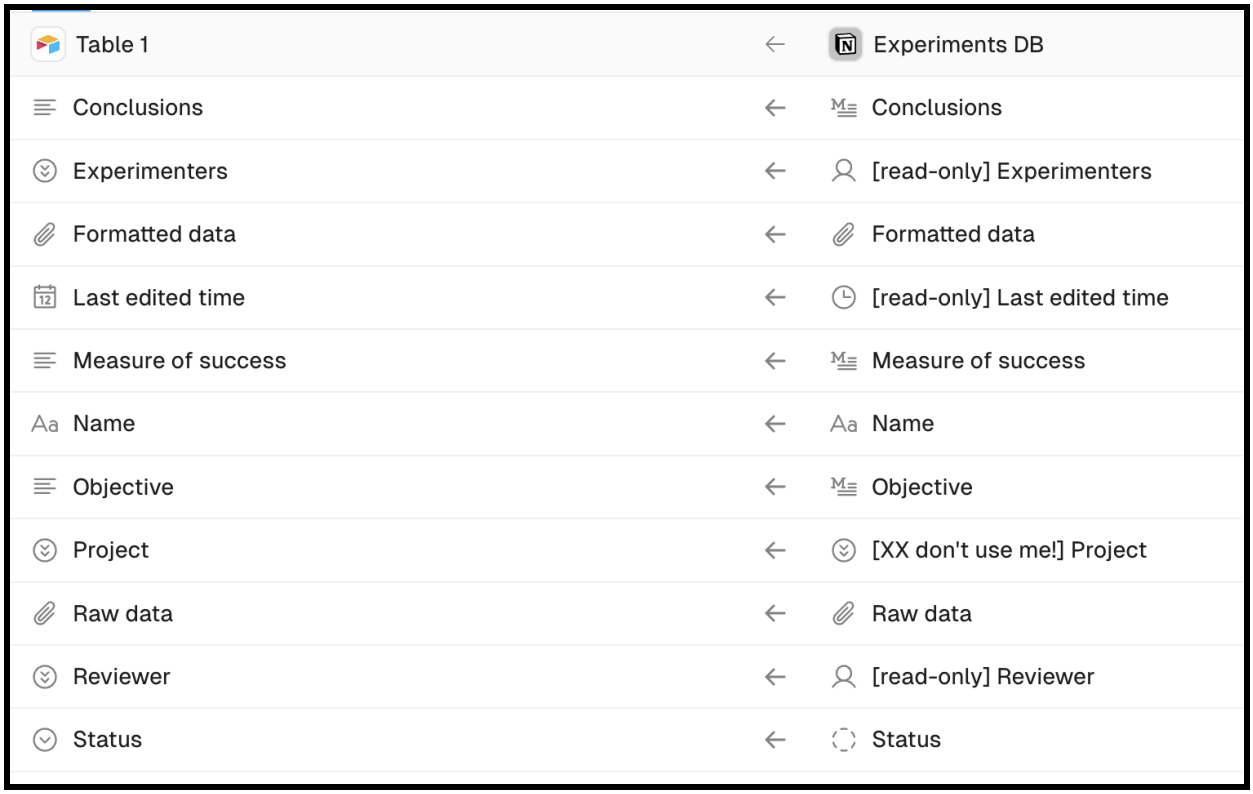


Whalesync

Go to https://www.whalesync.com/ and create an account.

To sync Notion → Airtable, first create a new Airtable database and enter all of the fields that you want to sync. You will need to properly select a reasonable field type for each one (you can see the icons in ours below). Once you've created the destination airtable, setting up Whalesync is pretty self-explanatory and the software will walk you through authenticating on both sides.

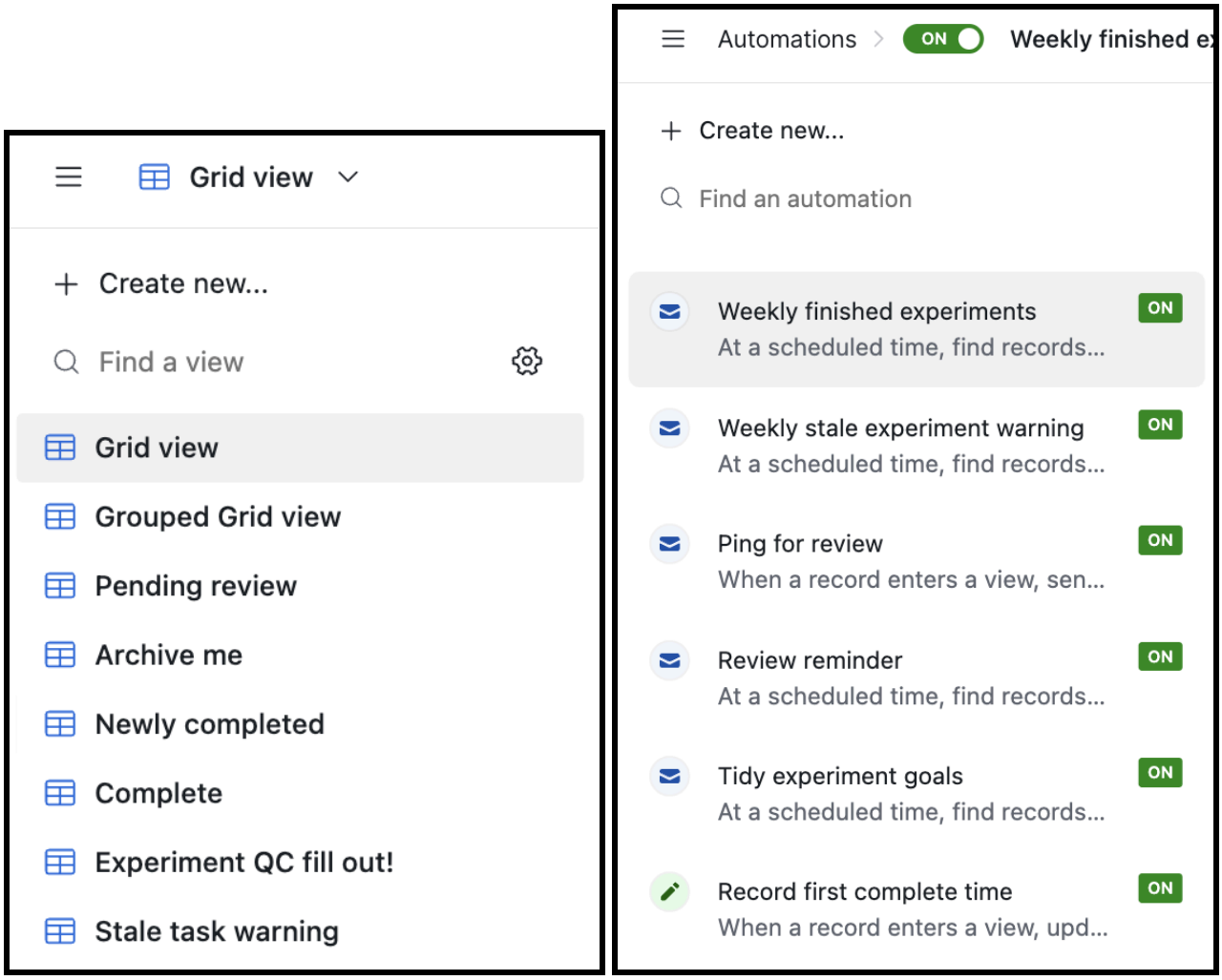
You'll end up with something that looks something like this:



NB: if you want to duplicate our entire setup, you're going to end up with TWO airtable bases. One is an airtable base of Experiments, linked with Whalesync, just for automated emails. The other is the LIMS (template provided below).

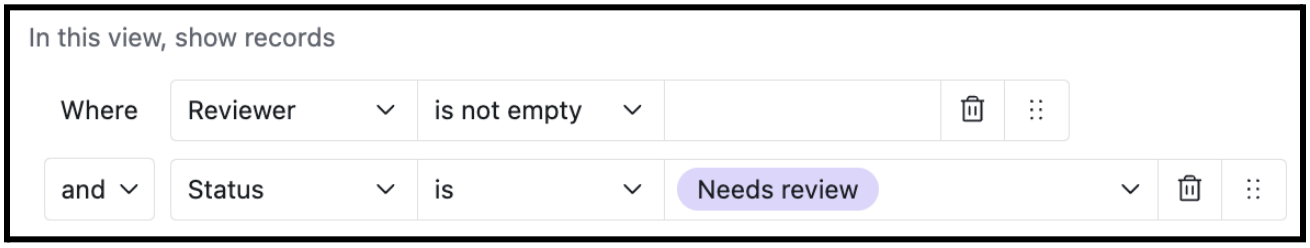
Airtable Automations for Experiments

To enable airtable automations, you'll be setting up a number of views and automations. Here are all the views and automations we have set up. We'll highlight just a few of them below, the others are pretty self-explantory.

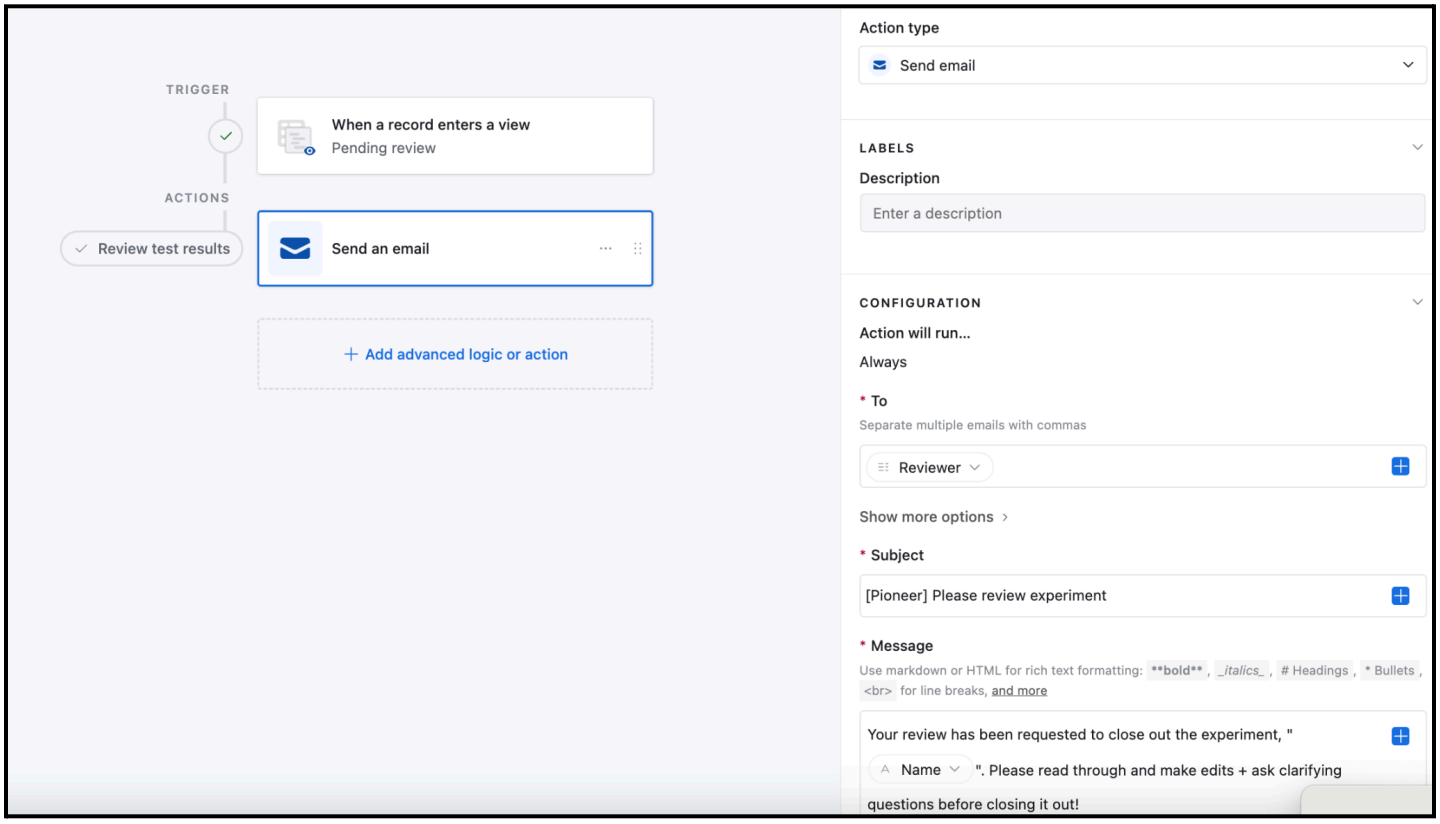


Ping for review

Begin by setting up a view called “Pending review” which is filtered by the status tag “Needs review.” Also ensure at least one reviewer is listed; this makes the automation robust to the order in which people change status and tag reviewers.

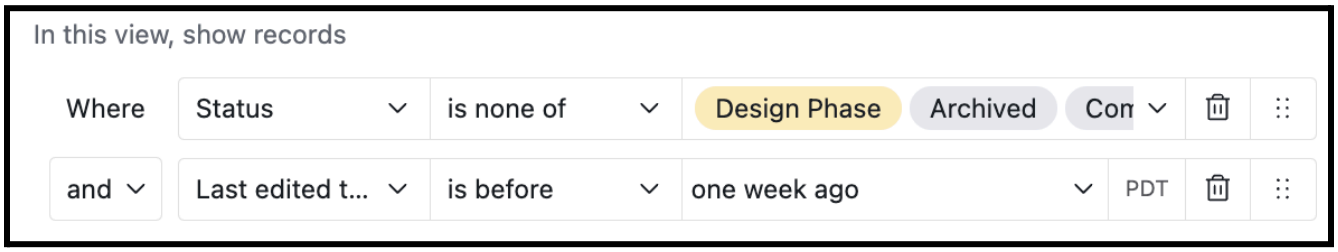


In our case, the usernames of each person are their email, so all this automation does is send an email to the reviewer.

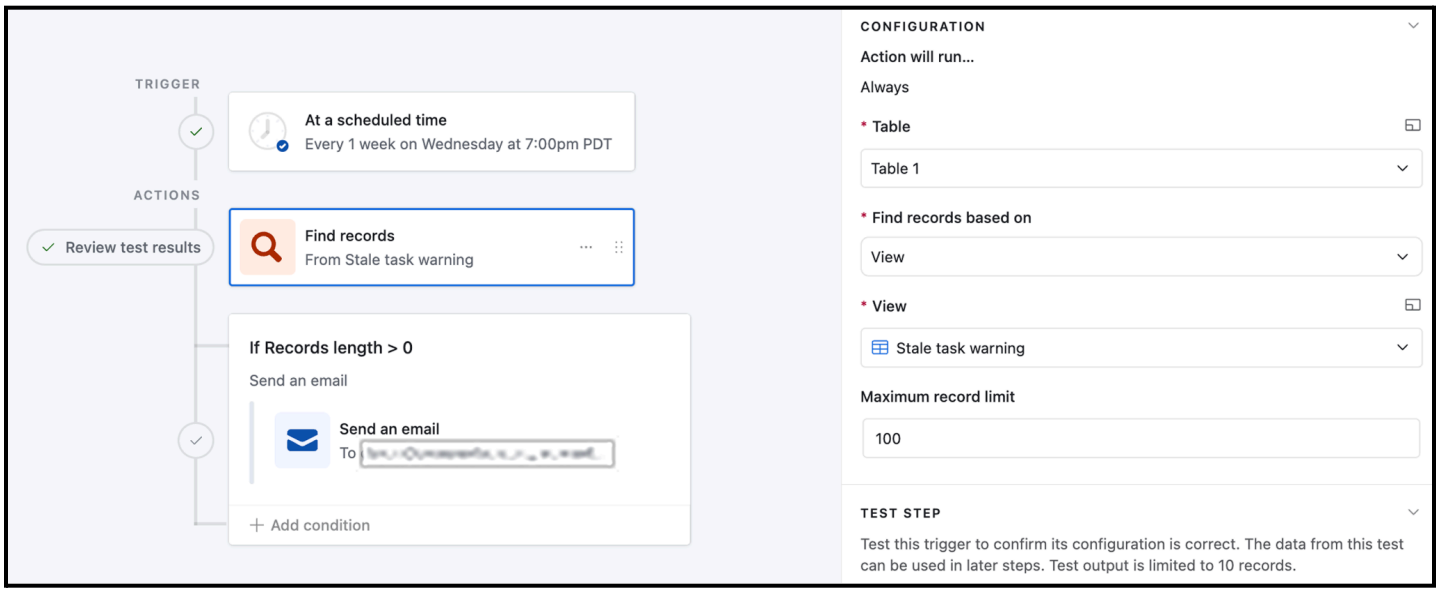


Stale task warnings

Begin by setting up a Stale task warning view. Filter out all of your statuses that indicate something is inactive.

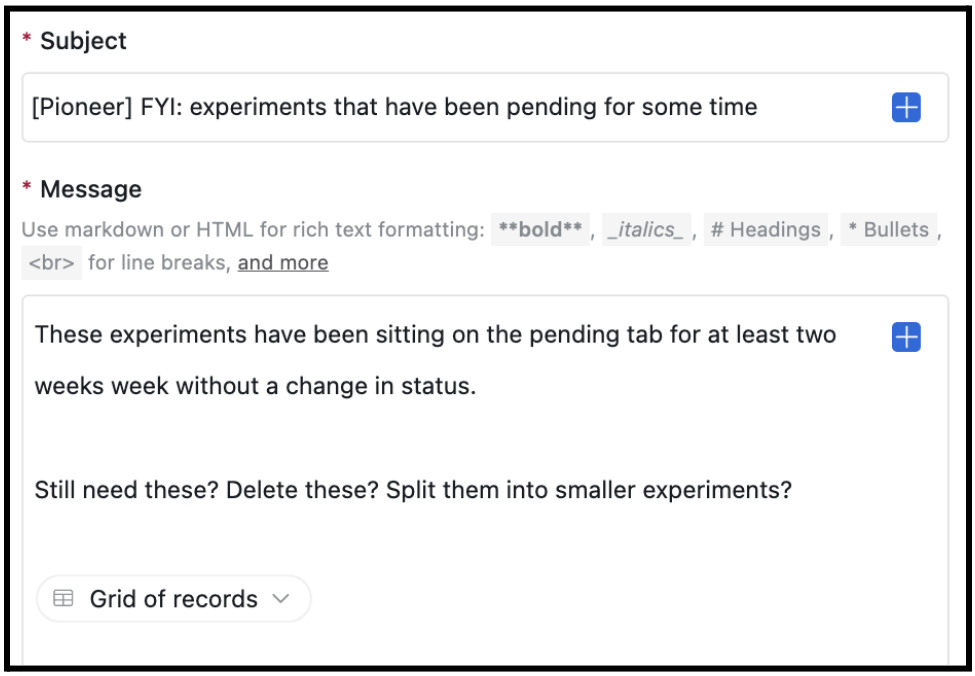


At a fixed time of week, query this view for all stale tasks.

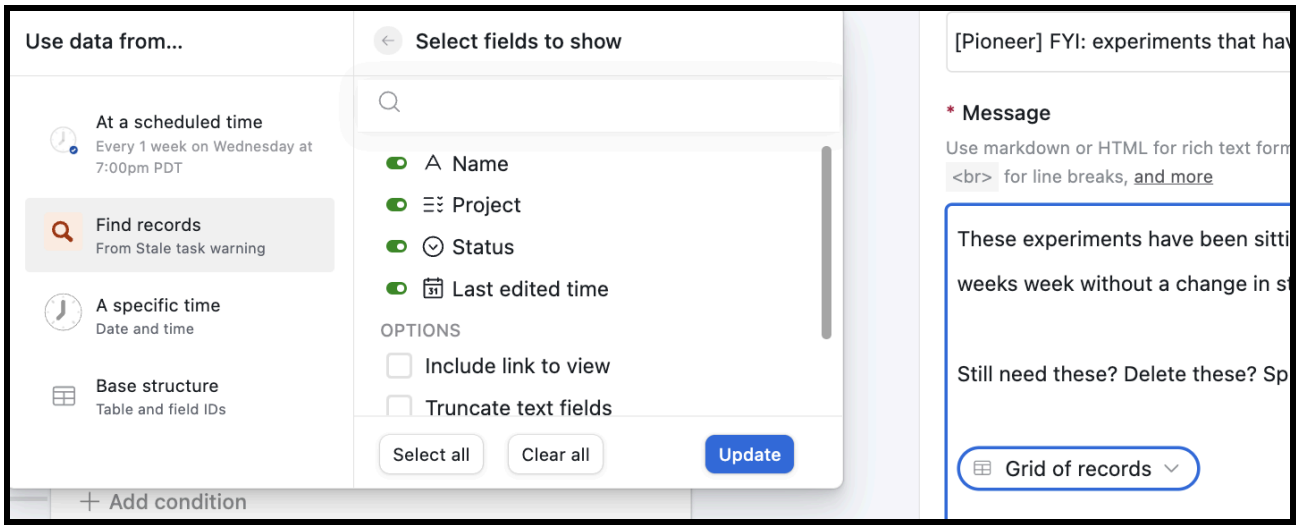


To avoid sending blank emails if there are no stale tasks, wrap the send email in an `if` that checks for the existence of stale records.

The list of records can be sent with the email as a grid:



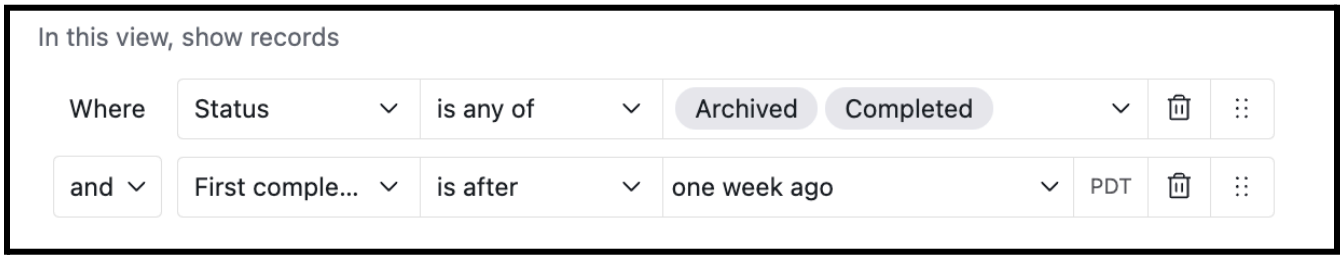
You can configure what information you want to display in the grid by “Edit token” on the “Grid of records”.



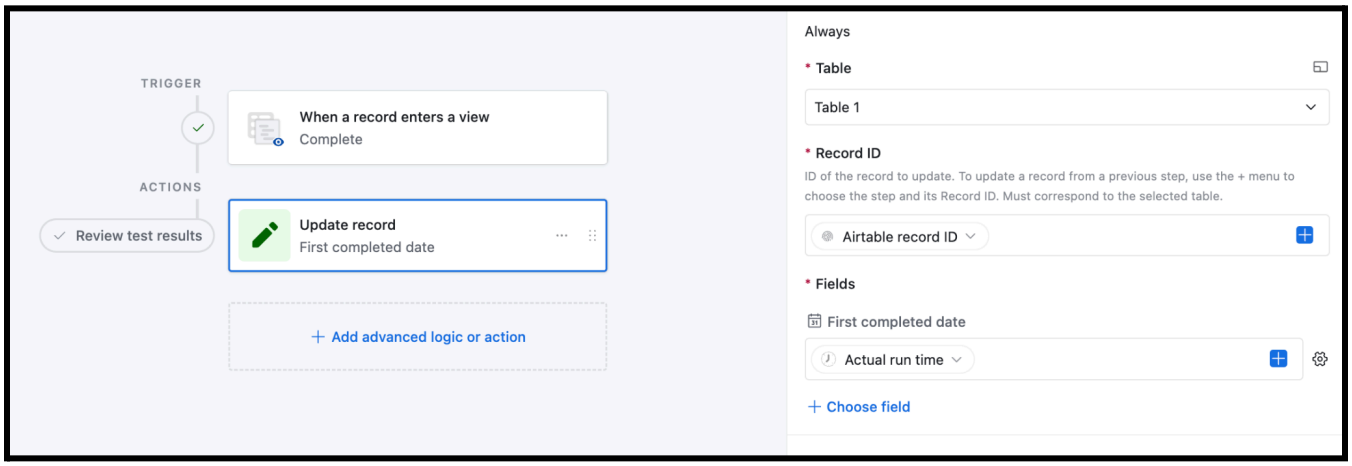
Record first complete time

In order to send out the ‘Weekly finished experiments’ email, we want to record the first time that an experiment is marked as complete. We used to use the ‘last edited by’ field, but sometimes experiments get edited after they’re completed or updated with retrospective notes, so this isn’t quite the same thing.

Add a column called ‘First complete date.’ To record the first complete time, set up a view called “Newly completed” that simply filters for complete experiments that have a recent ‘First complete date’.



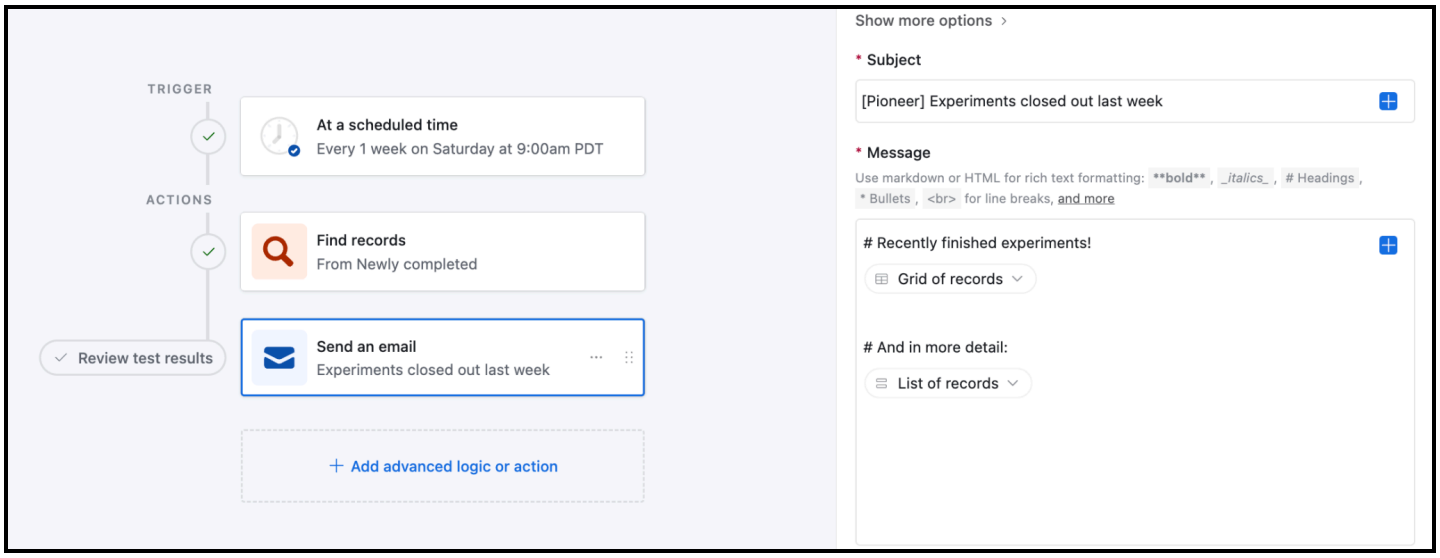
Next, set up an automation that writes the date into the ‘First complete date’ field when a record enters the “Newly completed” view.



Weekly finished experiments email

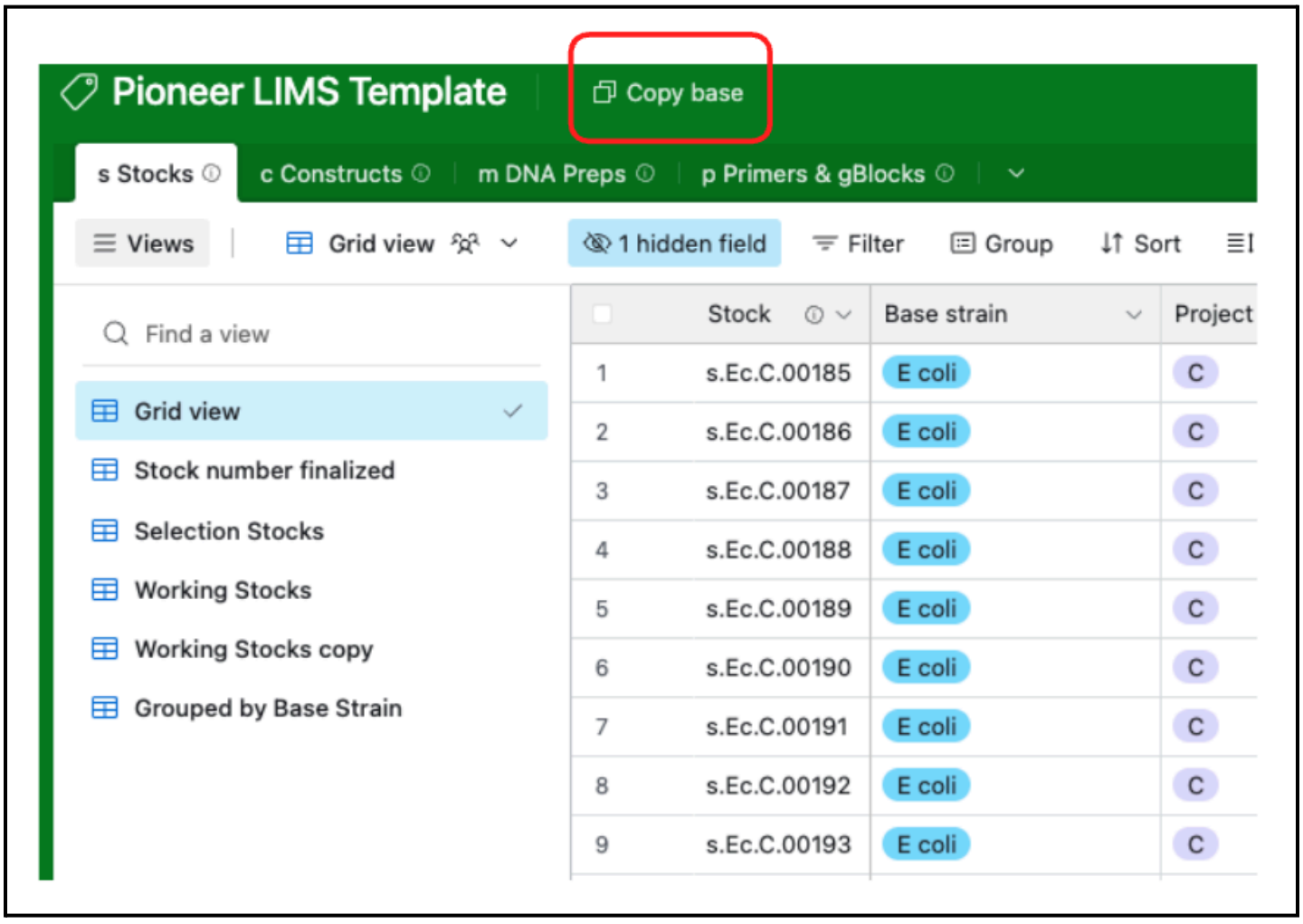
Begin by setting up the “Record first complete time” automation above.

At a scheduled time each week, grab all the records in the “Newly completed” view and send them out in an email.



Airtable LIMS

We’ve prepared a [template of our airtable LIMS](#) that you can copy and modify. To use it, press Copy Base at the top.



To enable custom unique numbering, first create an autonumber field. Then have the primary key for the table be a formula. The formula can be customized to include human-readable letters etc based on other fields in your table.

For example, we wanted human-readable stocks with both the base strain and project code included in a short name.

	Stock	Autonumber	Base strain	Project Code
1	s.Em.D.00320	320	E coli MFDpir	D
2	s.Em.D.00339	339	E coli MFDpir	D
3	s.Em.D.00340	340	E coli MFDpir	D
4	s.Em.D.00364	364	E coli MFDpir	D
5	s.Em.D.00365	365	E coli MFDpir	D
6	s.Em.D.00366	366	E coli MFDpir	D

Here is our Stock formula:

```
CONCATENATE("s.",
SWITCH(
{Base strain},
  "Selection", "Sl",
  "E coli", 'Ec',
  "S cerevisiae", 'SC',
  "S elongatus", "SE",
  "B subtilus", "BS",
  "C psychrerythraea", "CP",
  "D radiodurans", "DR",
  "R radiotolerans", "RR",
  "H elongata", "HE",
  "G obscurus", "GO",
  "P halocryophilus", "PH",
  "E coli WM6026", "Ed",
  "E coli K12", "Ek",
  "E coli K-12 MG1655", "Ek",
  "E coli MFDpir", "Em",
  "Cheese", "Ch",
  "Mold","Mo",
  "K variicola","Kv",
  "C necator","Cn",
  "A oryzae","Ao",
  "P stuzeri","Ps",
  "P putida","Pp",
```

```
  "D agitata","Da",
  "X autotrophicus","Xa",
  "A baylyi","Ab",
  "H campaniensis","Hc",
  " A ferrooxidans", "Af",
  "O polymorpha","Op",
       '*'
),
".",
{Project Code},
".",
RIGHT('00000' & Autonumber,5)
)
```

Similarly, we like for our constructs to have an optional ‘short name’ that make them more human readable. They end up looking like this:

Construct	Short name
c.00096.pORTMAGE-1	pORTMAGE-1
c.00097.pORTMAGE-2	pORTMAGE-2
c.00098.pORTMAGE-3	pORTMAGE-3
c.00099.pORTMAGE-4	pORTMAGE-4
c.00100.pZT247-pANS-RhamsfGFP	pZT247-pANS-RhamsfGFP
c.00101.pZT266-pANS-GFP	pZT266-pANS-GFP

And it’s enabled by this formula:

```
CONCATENATE("c.", RIGHT('00000' & Autonumber,5), ".", {Short name})
```