Inputs:
user_ id
request _language
n_items
oldest _ item
latest- item

What does the code need to do?
-take in the inputs
-return an object
-create a class which sets the arguments

Task1:
-return the lastest n_items (upto latest-item)
-Things to check:
- Are the the items in order or do they need ordering?
- How to deal with items at the same time especially the last term ?

↳ Ordered to millisecond so this is not a problem.

Yes

update: I was returning the newest n_items not the items before the latest-item.
Data appeared ordered but sort to check, reverse order to get latest item above newer ones.

To verify terms aren't missed check the number of content

terms between the latest item and the latest output time. There should be no Approved terms. (As long as the userfollows no one)

Task 2:
-return the n_items older than the oldest_item
- Things to check:
-There are enough stored terms
- all items should be approved
-amend Task 1function to a range

General checks:
-oldest_ item or latest-item do not have an id at the some time. Reassign the other when one term is set.

Optional tasks:
- language filter: Whether checking for approval check the language is available. To do this we need to read the language string as a list of terms and check that the content language is in the user language list.
- item. id for followed users should be stored in separate list to unfollowed. Before returning content stack the lists so followed users are first.
- Task3 if latest-item is more than a day old then n_items are found from now rather than before latest.

Improvements:

created separate functions to determine pivot in search and list of understood languages. This means the some functions can be used in finding new/old content as is used to find all valid content. Also makes code more readible..

-Raise errors if user_ id or item_ id are invalid.
- ignore items with no timestamp.
- If language is being checked then ignore iterms with no language set.
-Options for user following: Show content further in time range from followers or find the closest time items then show priority first. I have chosen to prioritise following over time. So n_items from followers then non- following if n_items are not sound.

The first item is dated as being created after today this doesn't cause a problem but I'll create a time check to remove impossible content.

- The priority for followed users is failing. - No priority terms are found. Error in time filtering the first time is aster the last-tim. When the latest item is given the first time should be the time of this item and the last time is now. When the oldest item is given the time of this should be the last time and the first time is the earliest time in the content. Reviewed all the timing comparisons in pivot terms.

- Tested the validity checks, all tests are now passing.