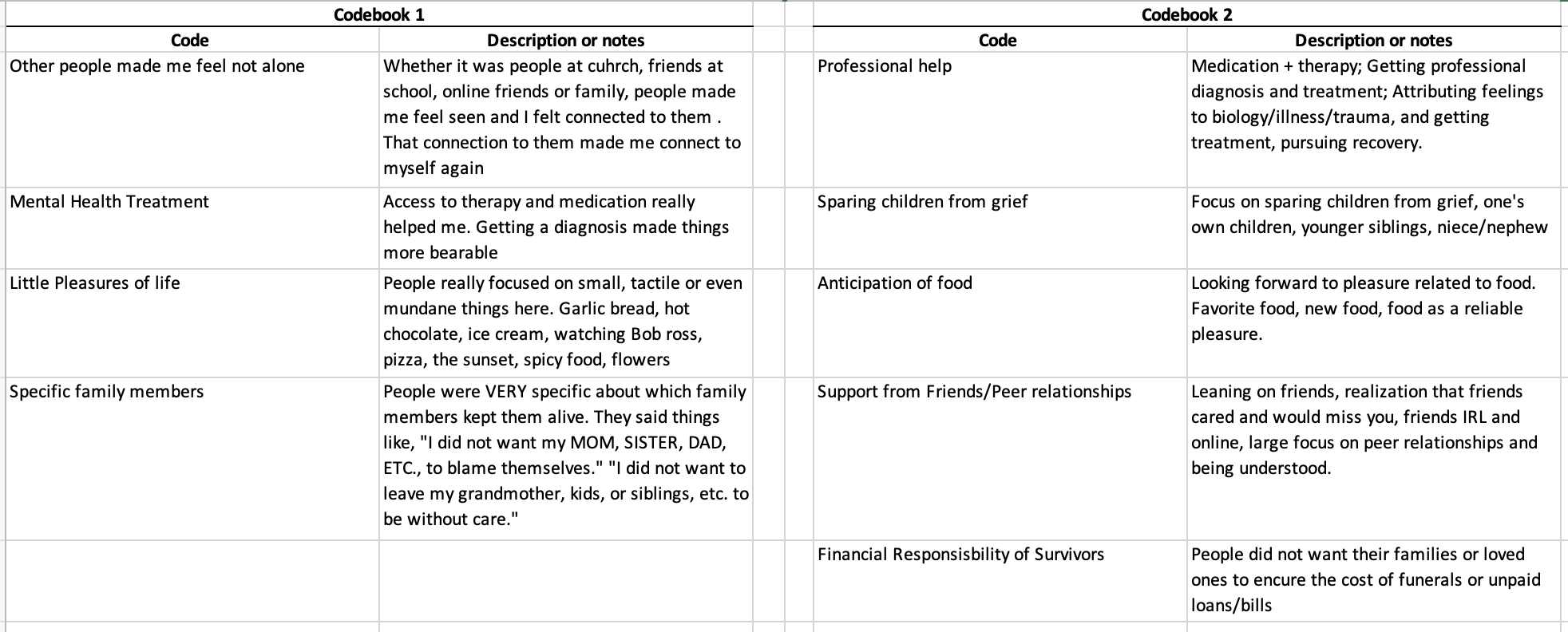
**Comparing Codebooks**

**[Version of 16 March 2024]**

Your task is to compare two different codebooks that were created for the same set of data, by identifying codes in the two codebooks that match.

This will be done using an Excel spreadsheet that contains the two codebooks, one on the left, and one on the right. Here is an example where the dataset was a sample of responses from previously suicidal people explaining what got them through their dark times.

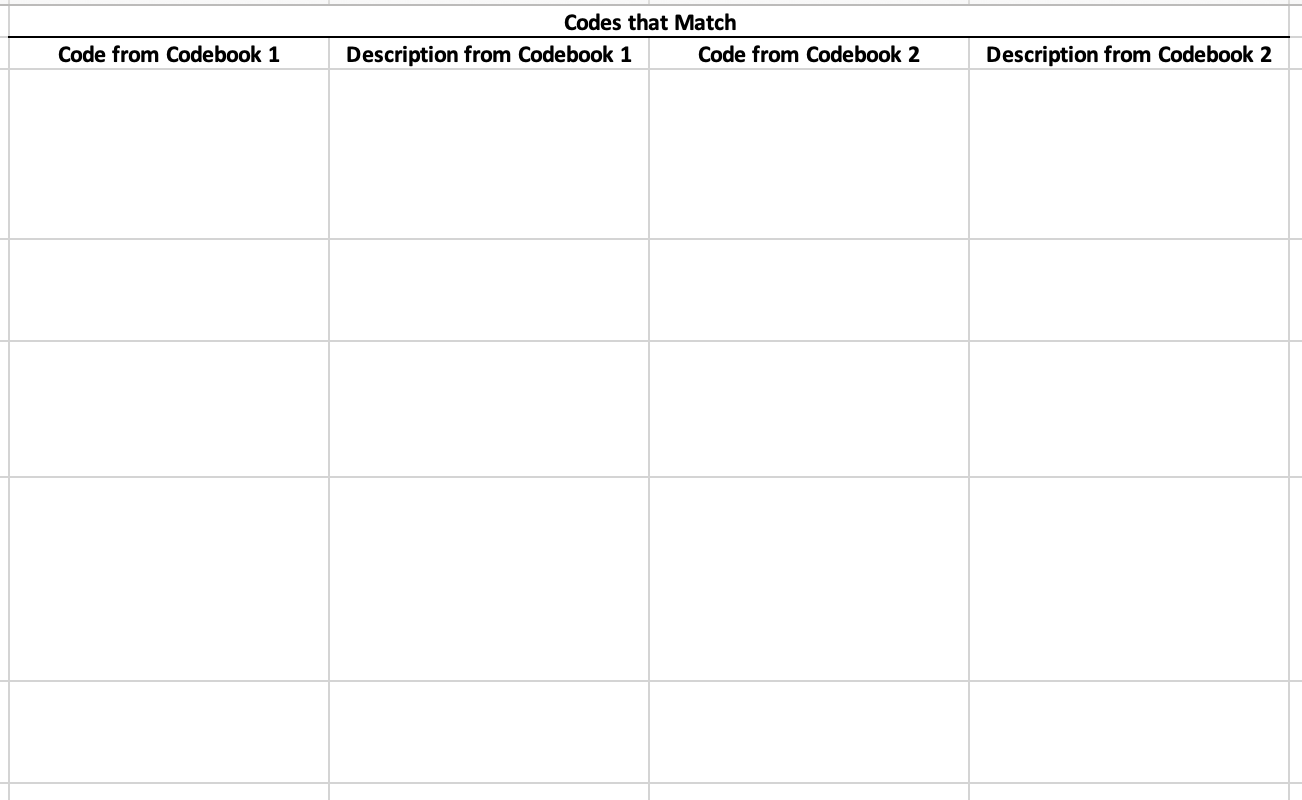


There are two different criteria that can be used for matching codes. You’ll be told which criterion to use (or you might be asked to do this twice, once each way).

**Loose match instructions:** Two codes are considered to be a loose match if they have concepts in common, even if they aren't exactly describing the same thing. This could include one being more general than the other (e.g., a code for "Lethal weapons" would be a loose match with "Kinds of firearms"), or the two having significant overlap (e.g. a code for "Zoo animals" would be a loose match for "Large mammals" -- you find snakes and birds in a zoo and you don't generally find moose or narwhals -- but "Wild animals" would *not* be a loose match for "House pets"). If these codes were each used to group documents, you would expect overlap of many (but not necessarily all) of the documents in the groups.

**Tight match instructions**: Two codes are considered to be a tight match if they seem to be rough paraphrases of one another, basically what you would expect if two people had identified the same basic concept, but were expressing it in different ways.  For example, "Faculty positions" and "University teaching jobs" would be a tight match (even though they're not *exactly* a match, e.g. research professors don't teach). If these codes were each used to group documents, you would expect more or less the same documents to be in both groups.

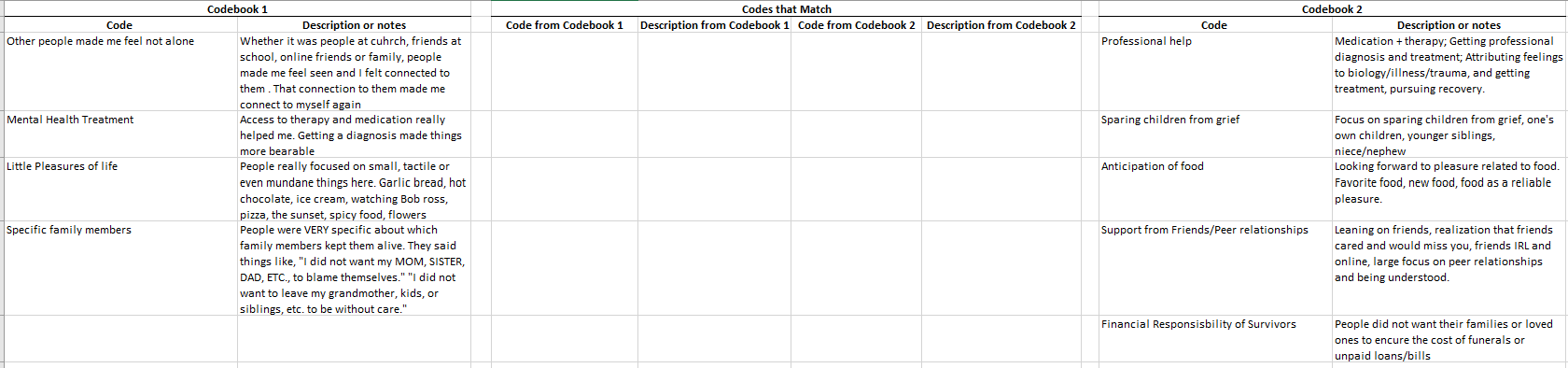
In your spreadsheet you’ll have a blank center section for codes that match. It looks like this:



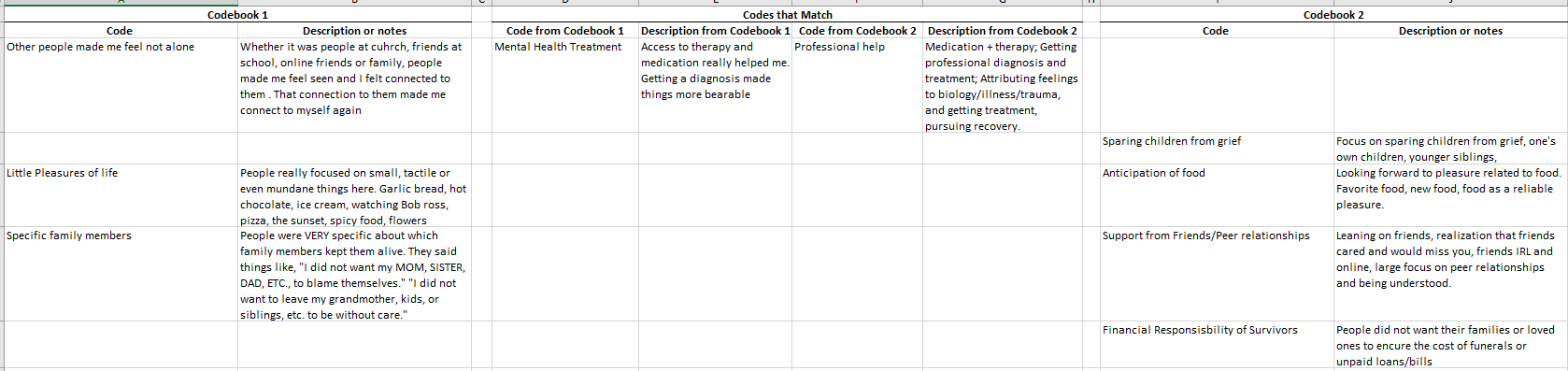
The process for creating the matching is very simple and it’s the same for either the loose or tight instructions: if you decide that two codes from the different codebooks match, **cut-and-paste** those Codes and Code Descriptions so that they’re on the same line in the center. Note that codes should be matched *one-to-one* (not one-to-many or many-to-one), which means that if one codebook has more codes than the other, some codes will *necessarily* be left over for the larger one.

Here’s a simple example workflow. We will discuss it both ways: once doing it with the *tight* criteria and once with the *loose* criteria.

1. Regardless of whether you are following the tight or loose instructions, you begin with a spreadsheet containing Codebooks 1 and 2 on each side and a blank section in the middle, which is where you will put codes that match.



1. The way you show which codes match is to cut and paste the corresponding entries from the codebooks 1 and 2 into the blank section. Please remember to *cut* and paste, not *copy* and paste, and also please remember to cut/paste both the code and it description/notes.
2. Here is what the result in this example would look like, using the *tight* criterion.



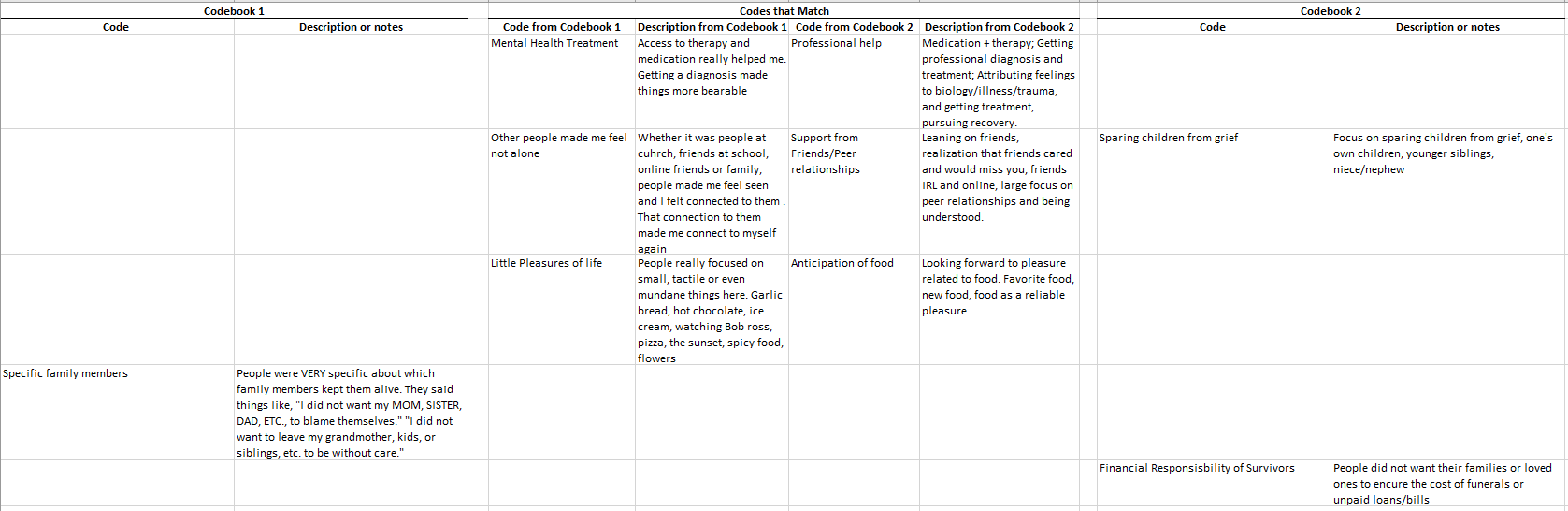
Using the *tight* match instructions,area match. Remember that according to those instructions, tight matches are what you would expect if two analysts had identified the same basic concept, just expressing it in different ways.

for the choices in this example

* For *Mental Health Treatment* and *Professional help*, it’s clear from the code names and their descriptions thatboth codebooks are talking about the same basic concept, just phrasing it in different ways. The vast majority of items you would code with one of these would also get coded with the other.

* What about ?
  + This pair fails onrather regarding the relationships that are involved: *specifically* friends/peers, which is a *subset* of the broader characterization of people who helped in Codebook 1.
* What about ?
  + Similarly, this pair fails onmore specific. It is o, while Codebook 1’s closest corresponding category includes food as a subset.

1. Here is the result one would get using the *loose* criterion:



Reasoning:

* Anything that’s a tight match is by definition also a loose match, so we have the same alignment of the codes about professional mental health help.
* Matching *Other people made me feel not alone* and *Support from Friends/Peer relationships*: On the *loose* instructions these are also a match. Recall that two codes are considered to be a *loose* match if they have concepts significantly in common, but it’s ok for one to be more general than the other, or to not match perfectly in other ways, as long as you would expect considerable overlap in the documents receiving the two codes. In this case you can see that although the subsets of people are not exactly the same, both descriptions are capturing the core idea of support from/connection with other people.

Notice that this example highlights the importance of understanding the point behind the study. In this dataset, the core question is *what* got people through their dark times, so that sense of connection and being seen/understood is central. If the research question behind the codebooks involved specifically *who* people relied on to get them through dark times, then one might make the choice *not* to match the two codes, even loosely, because in that case relying specifically on friends/peers might be considered different from relying on family.

* The reasoning for *Little Pleasures of life* and *Anticipation of food* is basically the same. Based on both the codes and their descriptions, these are capturing things that are substantially very similar, even if they’re not exactly the same thing.

Notice that this example highlights the importance of looking at the descriptions, not just the names of the codes. The language chosen as names of codes here seems pretty different. But when you also read the descriptions, you can see that the two independent coders are capturing very similar ideas.