

HAP - General LOG Book Skeleton

A GP Log Book with Time Tracking and Reporting

Alisha Awen - Harmonic Alchemy Productions*

Monday 2025-Jun-23

Contents

Contents	1
I About This Log Notebook:	2
1 References:	4
II 2025 Logs:	1
2 Lab Experiment Gone WRONG!	2
2.1 Introduction:	2
2.2 Reference:	3
2.3 Time Card:	4
2.4 Summary:	5
2.5 Details:	6

*harmonicalchemy@proton.me

Part I

About This Log Notebook:

“Everyone is a moon, and has a dark side which he never shows to anybody.” - Mark Twain

This Log Note Book SKELETON TEMPLATE was created to be used for almost anything... Anywhere you need to record events, make notes, diary entries, and keep precise time tracking of research or development or ANY of your own reasons...

Use this first section to compose a general overview of the Project of Projects that this log records data from... I HOPE this Helps you as it has me...

1. References:

Use this Section to add links to important external knowledge related to the subjects you are logging about here in this Log Book...

Part II

2025 Logs:

2. Lab Experiment Gone WRONG!

:LOG-DATE: <2022-12-20 Tue>

2.1 Introduction:

Just Kidding... No Panic... No pandemics were released and no animals were harmed... LOL

This is a dummy placeholder pretend Log Entry... Cloned from the **Meta:** Binder: **Generic Title** (*General Log Entry Template*)...

Commandeer this fake Log entry (*change it all*) to use as your already-in-place boilerplate for your VERY FIRST LOG ENTRY...

I made all this dummy content to see a real-world example exported to PDF... (*and also a sneaky way for me to test all the org-mode, org-babel, L^AT_EX, HTML etc. outputs, formats, you-name-it*)

For Me... This is a Quick Sanity Checkup Examination LOGBOOK for the publishing capabilities of Modular Emacs... All of which are conveniently part of a template so that once all of those things are looking GREAT (*quirks, and bugs all FIXED mostly*) Well then.. I will ALSO have right here in this Org-Mode Document, A fine example of *Literate Programming* using plain text as the SOURCE for EVERYTHING...

For You... The text based author / programmer / scientist / composer / filmmaker / mechanic / artist / architect / dietitian... (*huh what? Why did "dietitian" pop in my head? Do you think I look fat? LOL*) Anyway... As you can see by now... YOU IS EVERYONE! We ALL NEED This thing in our lives YESTERDAY right?

This document will contain snippets of many things USEFUL for your work-flow and creative output... Hopefully getting all the tech stuff DONE and OUT OF THE WAY so you can Do Your Thing and Create professional Docs and Web Pages that LOOK Fantastic! Yeah!

This template was used to set L^AT_EX styling for LOGBOOKS

Because of that, This DUMMY Example shows up in the exported PDF... Showing it with ample text mods etc... Which is MUCH MORE than you would want to use (*that would clutter your Final Docs and end up making them HARDER to read*) BUT good for examining how different font styles, and effects look in this example document...

Default L^AT_EX Styling For LOGBOOKS is All Set... You benefit from all my research & tweaking... So... You can NOW simply change the content to ANYTHING org-mode accepts...

Go Ahead... DO IT ;-)

BTW, If you don't like the look of the exported PDF, then do what I am doing and change it... You can start by examining:

`~/ .emacs.d/Docs/TeX/log-book-setup-3.tex`

(*as you can see by the filename, I have tried three different mods already LOL*) So... GOOD LUCK with THAT.. L^AT_EX Styling is a fxxxxxg TIME SYNC Full of ANGRY moments! Stubborn Persistence Is The Only Way to Success!

2.2 Reference:

- **ORG SRC CODE BLOCK REFS:**

[jherrlin](#) (*John Herrlin - BLOG TOP*)

[learn-awk](#) (*Learn Awk inside ORG-MODE - J.Herrlin GitHub Repo*)

[Emacs Orgmode Source Code Blocks 2](#) (*j.herrlin*)

[Inline Code Blocks](#) (*orgmode.org DOCS*)

[Results of Evaluation](#) (*orgmode.org DOCS*)

2.3 Time Card:

Weekly report starting on: *[2022-12-26 Mon]*

L	Headline	Time
	Total time	1:22
2	<i>Lab Experiment Gone WRONG!...</i>	<i>1:22</i>
3	Summary: [0/3]	0:05
3	Details:	1:17
4	First Experiment [0/3]	1:17
5	Tables:	0:59

THE ABOVE TABLE IS A DAMN LIE! I HAVE SPENT YEARS ON THIS PROJECT

(Think of the above table as how much time you will SAVE if you use this Template to LOG and Document your Next PROJECT!)

2.4 Summary:

What a MESS! It was my first attempt... Should we try to do it again right next time?

1. ☒ **First Experiment:** (*COMPLETED*)

DETAILS

This is the FIRST EXPERIMENT I did... No one died (thank God) but it was a Total Horrible MESS! I am still picking up all the pieces.. See Details for more information...

2. ☐ **Second Experiment:**

DETAILS

Write a SHORT ONE PARAGRAPH SUMMARY HERE...

This Item does not have a DETAILS doc associated yet...

Hence the BROKEN LINK ABOVE

If a tree falls in the woods and No One is there to see or hear it, will it cause a broken link found error in L^AT_EX Processing? - LOL

SO... Go Ahead and **Create your Details Doc...** Change Headings, Links, and Anchors Accordingly...

3. ☐ **New Item:** (*COPY/CLONE more of these as needed*)

[BROKEN LINK: -LOG_GENERIC-LOG-DETAILS_new-item]

Write a SHORT ONE PARAGRAPH SUMMARY HERE...

This Item does not have a DETAILS doc associated yet...

Hence the BROKEN LINK ABOVE

If a tree falls in the woods and No One is there to see or hear it, will it cause a broken link found error in L^AT_EX Processing? - LOL

SO... Go Ahead and **Create your Details Doc...** Change Headings, Links, and Anchors Accordingly...

2.5 Details:

First Experiment

CLOSED: [2022-12-28 Wed 12:42] **SUMMARY**

This is the FIRST EXPERIMENT... Not really... In truth it is a LOG NOTEBOOK Item Details Example... in the form of an ad-hoc org-mode cheat sheet... BUT I almost killed someone during my extreme frustration of trying to get tables aligned properly on export... Just kidding about the killing but NOT about the frustration... LOL. My Next Experiments will be Much Better! I promise!

Plain Lists:

The LATEX export back-end accepts the ‘environment’ and ‘options’ attributes for plain lists. Both attributes work together for customizing lists, as shown in the examples:

Some ways to say "Hello":

- **Hola** (*first level List item*)
 1. **second level list item**
 - *Third level list item*
 - a) Fourth level list item
 - b) Fourth level list item
 - *Third level list item*
 - a) Fourth Level Numbered Item
 - b) Fourth Level Numbered Item
 - c) Fourth Level Numbered Item
 2. **second level list item**
- **Bonjour**
 - **second level list item**
- **Guten Tag**
 - **second level list item**

Tables:

Org Mode Tables are a bit more complicated than Markdown Mode tables which are static tables only. Also the syntax is different (darn it!) but these org-mode tables are great because they are smart tables! You most probably won't type a table like this outside of Emacs. The manual alignment without tool-support is very tedious. You don't have to do careful tedious aligning or restructuring your tables in org-mode! You can enter them very sloppily like so:

Table 2.1: SLOPPY Org-Mode Table Input - no problem

My Column 1	My Column 2	Last Column
42	foo	bar
23	baz	abcdefg
65		

Org Tables are Cool! They even know how to justify numbers differently than text! And there's a bunch more you can do with tables... (i.e., spreadsheet stuff)... add in GNU calc or whatever its called and you can do some wild stuff with weather data.... omg! lol...

Org tables have cells of at most one line long: i.e., **There Is No Such Thing As A Multi-Line Table Cell In Org**. If you need to do that you have to play some tricks... UGH! But you know what? Table INFO needs to be terse... Its the purpose of spreadsheets.... You eyes want to see the big picture, and the RESULTS... not read through cryptic text... If you have LONG descriptions in table cells **YOU ARE USING THE WRONG DISPLAY TYPE...** Not Tables... Use Something else... lol that reminds me of the old days when we used HTML tables to format all kinds of things... it was semantic confusion for sure... HTML5 is a welcome CURSE as well... Hey.. that's why they have to pay us the BIG BUCKS right? %^)

Column Formatting: Columns are automatically aligned by default as such:

- *Number-rich Columns line up to the Right*
- *String-rich Columns line up to the Left*

Table 2.2: Default Table Column Formatting

1	2	3
xyz	xyz	xyz
4	5	6
7	8	9

- Table with Custom Aligned Columns:

If you want to override the automatic alignment Use: <r>, <c> or <l>.

Table 2.3: Table With Aligned Columns

1	222-42-84	3
Right Aligned	Center Aligned	Left Aligned
xyz	xyz	xyz
4	55	6
7	88	9

- Header row:

You can create tables with a header row by using an horizontal line of dashes to separate it from the rest of the table... You could also use a horizontal line at the bottom of your table to show a totals row or something else... In other words, horizontal lines help separate sections of your table, not just the header row...

Make a horizontal line by typing `|-` on a blank line above, below, or in the middle of a table, and then pressing `TAB` while your cursor is somewhere inside the table... The following example uses the horizontal line to create the Header Row... Org mode operations within the body of the table will not effect the header row... Sorting operations within a table will only affect the rows between horizontal rows... (*I am not totally sure about the above but so far it seems to be working that way - “RTFM!” OK I will... someday*)

Table 2.4: Table with Header Row

Name of column 1	Name of column 2	Name of column 3
Top left	Top middle	
		Middle right
Bottom left	Bottom middle	

- A very long table to test “sticky headers”:

Table 2.5: Extra LONG Table (to see how \LaTeX handles them)

Name of column 1	Name of column 2	Name of column 3
<i>Top Left</i>	<i>Top Middle</i>	<i>Top Right</i>
2		
3		
4		
5	<i>Middle</i>	<i>Right</i>
6		
7		
8		
9		
10	<i>Middle</i>	<i>Right</i>
11		
12		
13		
14		
15	<i>Middle</i>	<i>Right</i>
16		
17		
18		
19		
20	<i>Middle</i>	<i>Right</i>
21		
22		
23		
24		
25	<i>Middle</i>	<i>Right</i>
26		
27		

Continued on next page

Continued from previous page

Name of column 1	Name of column 2	Name of column 3
28		
29		
30	<i>Middle</i>	<i>Right</i>
31		
32		
33		
34		
35	<i>Middle</i>	<i>Right</i>
36		
37		
38		
39		
40	<i>Middle</i>	<i>Right</i>
41		
42		
43		
44		
45	<i>Middle</i>	<i>Right</i>
<i>Bottom left</i>	<i>Bottom middle</i>	<i>Bottom Right</i>

Table Placement:

- Table size:

Table 2.6: Full Width Table

Cell in column 1, row 1	Cell in column 2, row 1
Cell in column 1, row 2	Cell in column 2, row 2

HOPEFULLY The table above spreads out across the entire page... We shall see... I Guess NOT... How do you do that? Hmmm... ok still doing the L^AT_EX Head Scratch Game here... Stay Tuned...

- Align Table Centred: (*THE DEFAULT*)

Normal DEFAULT Tables are Centered

Table 2.7: DEFAULT TABLE (as is)

a	b	c
1	2	3
4	5	6

- Align Table Flush LEFT:

As seen ABOVE, L^AT_EX CENTERS all Images, Tables, etc. by DEFAULT... You can override this on an Item basis (*or globally in your SETUPFILE*) by setting the :center attribute to nil... easy-peasy...

Table 2.8: Table Flush LEFT

a	b	c
1	2	3
4	5	6

- Typical Use Case Table Example:

MIXOLYDIAN Mode - TRUTH TABLE: (*org-mode center*)

REM - LCS:	I	-II	II	-III	III	IV	IV+	V	V+	VI	-VII	VII
ROMAN Scale Deg:	I		II		III	IV		V		VI		VII

MIXOLYDIAN Mode - TRUTH TABLE: (*org-mode verse*)

REM - LCS:	I	-II	II	-III	III	IV	IV+	V	V+	VI	-VII	VII
ROMAN Scale Deg:	I		II		III	IV		V		VI		VII

MORE With Tables: The following example will Export an **Extra W-I-D-E Table** on a **BLANK Landscape Page** (*i.e., no page header, page number, or footer*)

Table 2.9: Afro Eurasian Jungle Jazz Orch. xStringZ

Instrument Name	MIDI	OUTS	Articulation	HAP File Name	Vendor	LIB Name
Kemence	Sm6.1	LEAD	Relaxed Preset	HAP-MID-E_Kemence_Q2-2025.nki	NI Discovery	MID EAST
Erhu	Sm6.5	LEAD	Classic Preset	HAP-E-ASIA_Erhu_Q2-2025.nki	NI Spotlight	EAST ASIA
Ajaeng	Sm6.7	LEAD	Relaxed Preset	HAP-E-ASIA_Ajaeng_Q2-2025.nki	NI Spotlight	EAST ASIA
New York Weirdness	Sm6.9	LOW	Plucked Bass	HAP-SEQUIS_NYC-Weirdness_Q2-2025.nki	NI	Sequis
Aperture Ensemble	Sm6.12	ENS	Spiccato	HAP-APE-ENS_Spiccato_Q2-2025.nki	SPITFIRE	Aperture Strs
Emotive Strings	Sm6.12	ENS	ostinatos motifs	HAP-EMOT-S_Emotive-Strings_Q2-2025.nki	NI	Emotive Strs
Ensemble Modern	Sm6.13	ENS	SUS Accented	HAP-SSSN-S_ENS-Modern_Q2-2025.nki	NI	SSN Str Pro 2
Ensemble Traditional	Sm6.14	ENS	SUS Accented	HAP-SSSN-S_ENS-Trad_Q2-2025.nki	NI	SSN Str Pro 2
Pinhole Ensemble	Sm6.13	ENS	Staccato	HAP-PIN-ENS_Spiccato_Q2-2025.nki	SPITFIRE	Aperture Strs
Refractions	Sm6.14	STfx	Tremolo	HAP-RFRAC_Tremolo_Q2-2025.nki	SPITFIRE	Aperture Strs
Violin Ens Pizz	Sm6.15	STfx	Plucked Str Ens	HAP-DEFT-L_Violin-ENS-Pizz_Q2-2025.nki	NI	DEFT LINES
Violin Ens SUSTAIN	Sm6.15	STfx	SUS Str Ens	HAP-DEFT-L_Violin-ENS-SUS_Q2-2025.nki	NI	DEFT LINES
Cello Tremolo	Sm6.15	STfx	Tremolo	HAP-DEFT-L_Cello-Trem_Q2-2025.nki	NI	DEFT LINES
Strings 3&4	Sm6.16	STfx	dynamic pads	HAP-MELT-V_Str3-4_Q2-2025.nki	NI	MELT VIBES
CLUSTERFUNK	Sm6.16	STfx	trem ambi clstr	HAP-THRIL_CLUSTERFUNK_Q2-2025.nki	Galaxy Inst	THRILL

- Create Table from SRC CODE Block:

First Show Plain Text Shell Output:

```
| echo PID: "$$"
```

PID: 44510

Next Show Shell Output as a Table:

```
| ps -al
```

UID	PID	PPID	F	CPU	PRI	NI	SZ	RSS	WCHAN	S	ADDR	TTY	TIME	CMD
503	43402	43401	4006	0	31	0	410995808	11536	-	Ss+	0	ttys000	0:01.08	zsh

The Table in the PDF will not have any row or column lines... but you may be able to add more L^AT_EX attributes to fix this... I need to go do some more Research...

- Table Formats:

org-mode has 2 table formats, the default **org-table** and the other is **table.el**... You can do a lot more by converting your tables to use Table.el... To toggle between the two different modes type:

C-c ~ (*tilde*)

TIP: ADD A BLANK LINE Above and Below the Table. The blank lines should prevent a bug when converting the table with C-c ~

– Example Table Editing Procedure:

☐ **Place Your Cursor:**

Place the cursor *i.e.*, *point* inside your updated table then type C-c ~ (*tilde*) and answer y... The table in example below is normal org-table After doing above comment it will change to a boxed in look...

Table 2.10: Raw Table with Extra Long Column RUNS OFF EDGE!!!

Term	Definition
keyword 1	Extra Long line definition that continues beyond your frame width or beyond your set Olivetti line width
keyword 2	Extra Long line definition that continues beyond your frame width or beyond your set Olivetti line width

☐ **Place Cursor Inside Updated Table:**

Then Type: C-c ' *A New Special Table Editing Window Will Appear...*

☐ **Place Cursor Inside Second Column:**

Type: C-c C-c <

Do this command repeatedly until the column is the desired width...

☐ **Save Table With:**

Type: C-c ' (*to view results of edits*)

THE EXAMPLE BELOW IS RESULT OF FIRST PASS... (*After converting back to org-table for exporting*)

Tip: Use the universal argument key binding C-u to quickly repeat this step. For example, If your right column is 106 characters as in the above example, and you want to reduce the right column width to 50 characters do: C-u 56 C-c C-c < which will repeat: C-c C-c < command 56 times.

After typing: C-c ' (*to save and exit the table editor*) you will see your table resized as the above table now looks below...

Table 2.11: Resized Table (try #1)

Term	Definition
keyword 1	Extra Long line definition that continues beyond your frame width or beyond your set Olivetti line width
keyword 2	Extra Long line definition that continues beyond your frame width or beyond your set Olivetti line width

Table 2.12: Resized Table (Final)

Term	Definition
keyword 1	Extra Long line definition that continues beyond your frame width or beyond your set Olivetti line width
keyword 2	Extra Long line definition that continues beyond your frame width or beyond your set Olivetti line width

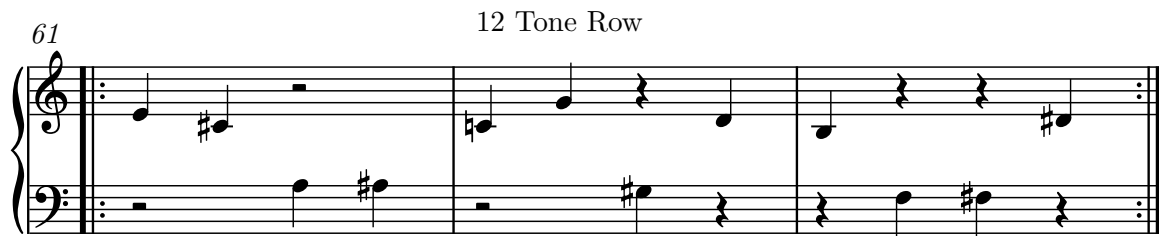
Second Experiment:**SUMMARY****LiliPond Music Notation:****Example 1:**

Figure 2.1: 12 Tone Row

Example 2:**2 Bar Example Boilerplate**

$\text{♩} = 80$

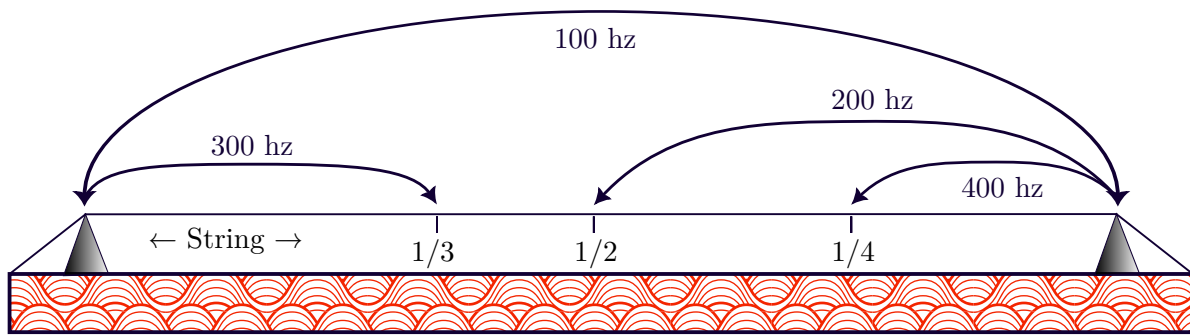
Violin I

Violin II

Viola

Violoncello

Figure 2.2: Example 2 Bar String Quartet

SVG & Graphviz:**Display SVG File Created in Inkscape:**

Monochord side view

Figure 2.3: Monochord Example showing Harmonic Nodes

NOTE: This took a bit of fiddling in Inkscape to get the labels positioned... My guess is the fonts are changed or something and that throws positions off... but by comparing the Inkscape edit window with the final PDF I was able to nudge things to get it looking good... THERE MUST BE A BETTER WAY or It SHOULD JUST WORK as is... right?

Automate Graphviz:

Generate DOT code and RENDER results

A typical use of a dot source code block is to produce a graph visualization of a data set. In this example, the following input table...

Table 2.13: Hello World Table for DOT

b	World!
---	--------

is passed to a source code block that is responsible for producing valid dot code...

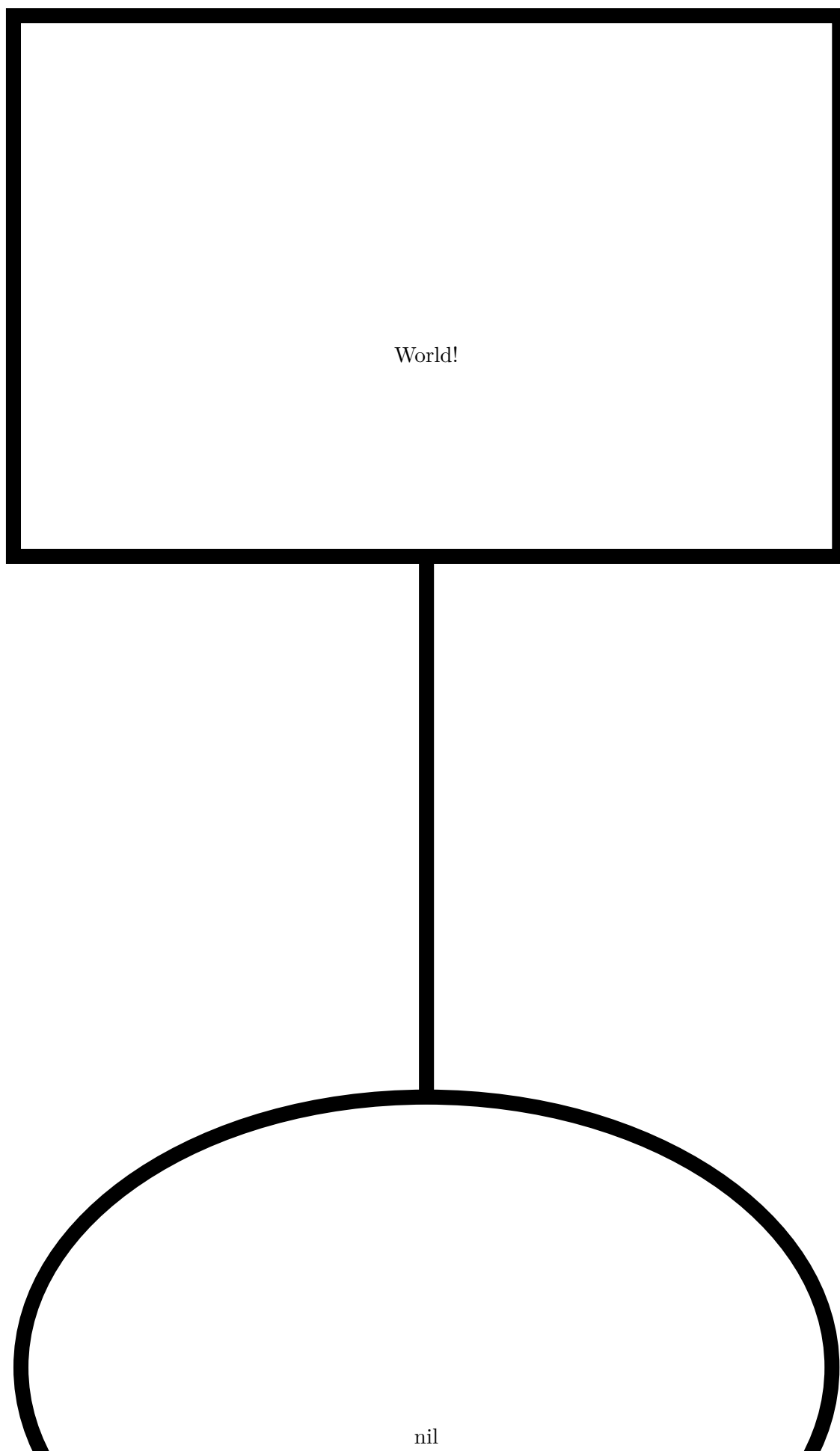
```
(mapcar #'(lambda
            (x)
            (princ
             (format "%s [label = \"%s\", shape = \"box\"];\\n"
                     (cl-first x) (cl-second x))))
 table)

(princ
 (format "%s -- %s;\\n"
         (cl-first (cl-first table))
         (cl-first (cl-second table)))))
```

```
b [label = "World!", shape = "box"];
b -- nil;
```

and this is chained to a dot source code block...

```
graph {
  $input
}
```



Nice! Simple... but it was a powerful process... COPY/CLONE this to do Fancier Things not just simple blocks like above...

Nested Clusters:

```

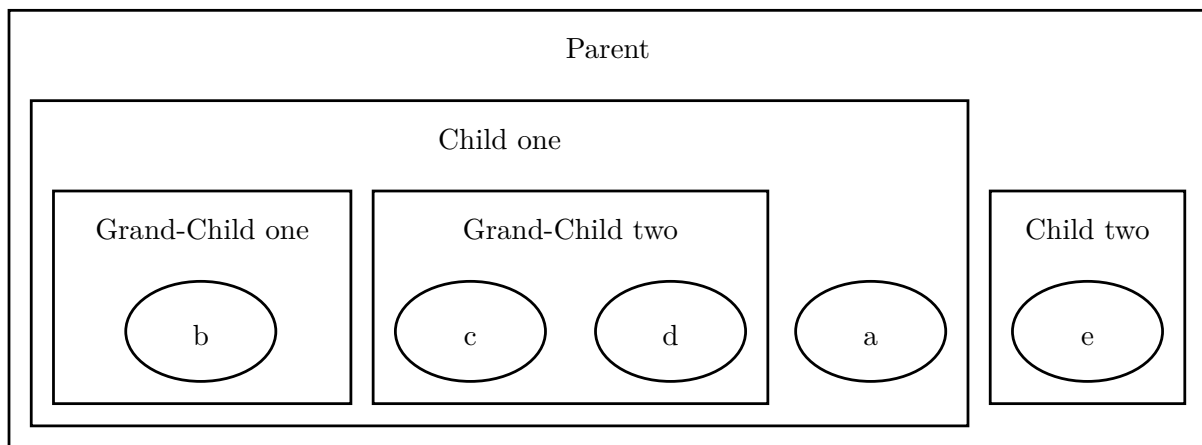
digraph D {
  subgraph cluster_p {
    label = "Parent";

    subgraph cluster_c1 {
      label = "Child one";
      a;

      subgraph cluster_gc_1 {
        label = "Grand-Child one";
        b;
      }
      subgraph cluster_gc_2 {
        label = "Grand-Child two";
        c;
        d;
      }
    }
  }

  subgraph cluster_c2 {
    label = "Child two";
    e;
  }
}

```



END: Experiments

What Shell Am I Using?

```
| echo $SHELL
```

Listing 1: SHELL Query?

```
/bin/zsh
```

- INLINE EMACS VERSION:
29.4
- INLINE ORG VERSION:
9.6.15
- EMACS VERSION WITH EXPORT Settings:

```
| (emacs-version)
```

Listing 2: Which EMACS Version?

```
GNU Emacs 29.4 (build 1, aarch64-apple-darwin23.6.0, Carbon Version 170 AppKit 2487.7)
of 2025-04-14
```

- ORG VERSION WITH EXPORT Settings:

```
| (org-version)
```

Listing 3: Which ORG-MODE Version?

```
9.6.15
```

Emacs 29.4 (Org mode 9.6.15)