No time for stress.
The first part, the density sorter works by receiving a chemical system from the user in the form of a string
Ex: Li-Fe-O
And outputs a table of the materials with that chemical structure sorted from lowest to highest density along with a bar graph of the 3 highest density and 3 lowest density materials and a scatter plot with the highest and lowest density points labeled and highlighted.
You can use the search multiple times in a row and it has a few features to ensure that results are at least reasonably easy to read (unless it spits out 2 billion (or zero) materials I guess).

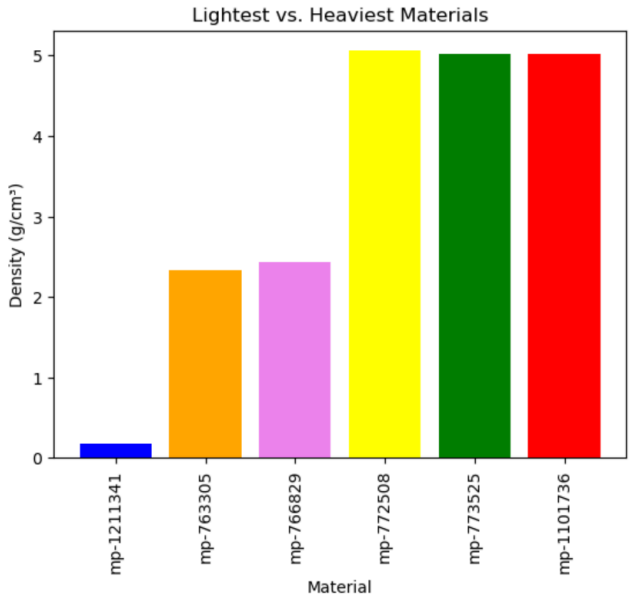
Here's some examples:

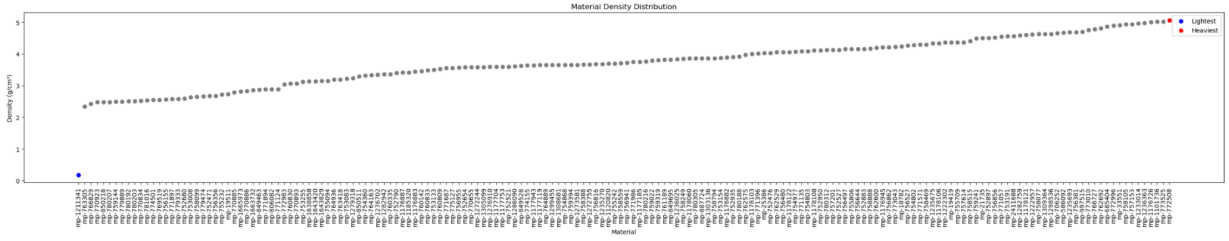
Li-Fe-O

```
Li-Fe-O
Materials Property Explorer
==========================

Sorting by density...
Retrieving SummaryDoc documents: 100% 176/176 [00:00<00:00, 21094.37it/s]

Exploration complete!
```





```
show table? y/n:
 y
```

	material_id	formula	density	formation_energy	volume	crystal_system	structure
88	mp-1211341	Li3FeO6	0.181168	-0.146007	1582.600705	Orthorhombic	[[0. 0. -17.12851736] Li, ...
141	mp-763305	Li8FeO6	2.334590	-1.968938	294.994036	Orthorhombic	[[1.94553673 3.36024988 2.52277082] Li, [1.692...
28	mp-766829	Li15(FeO4)4	2.429376	-1.798949	797.653579	Monoclinic	[[5.88791252 8.22165258 1.34682594] Li, [3.382...
112	mp-770923	Li5FeO4	2.480480	-1.970647	413.843014	Monoclinic	[[9.12830861 3.50965654 7.92487594] Li, [5.215...
62	mp-850218	Li35(FeO4)8	2.483035	-1.935224	803.625342	Triclinic	[[3.45709162 1.29920938 8.48878052] Li, [1.168...
...	...	...	...	...	...	...	...
4	mp-1236363	Li(Fe2O3)4	4.970073	-1.527047	215.731202	Triclinic	[[6.20716052 3.58545161 3.94078056] Li, [0.784...
154	mp-1176726	LiFe3O4	4.993785	-1.673904	158.595024	Monoclinic	[[2.0915155 2.172567 0.] Li, [-2.09151...
153	mp-1101736	LiFe3O4	5.018470	-1.673856	78.907460	Triclinic	[[-1.48982402 2.19943537 -1.43482232] Li, [-1...
156	mp-773525	LiFe3O4	5.018471	-1.672052	78.907448	Monoclinic	[[-1.532067 -2.588143 0.] Li, [0. 0. 0....
155	mp-772508	LiFe3O4	5.058357	-1.626865	78.285252	Orthorhombic	[[3.05565 0. 0.] Li, [1.527825 2.10...

176 rows × 7 columns

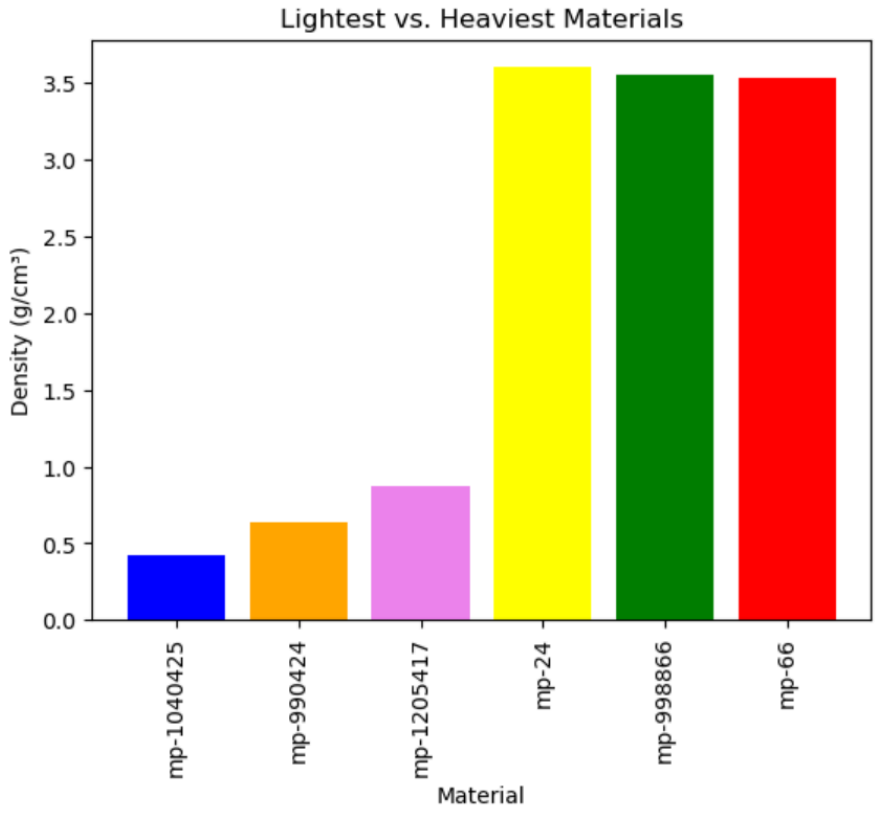
```
search again? y/n:
 y
```

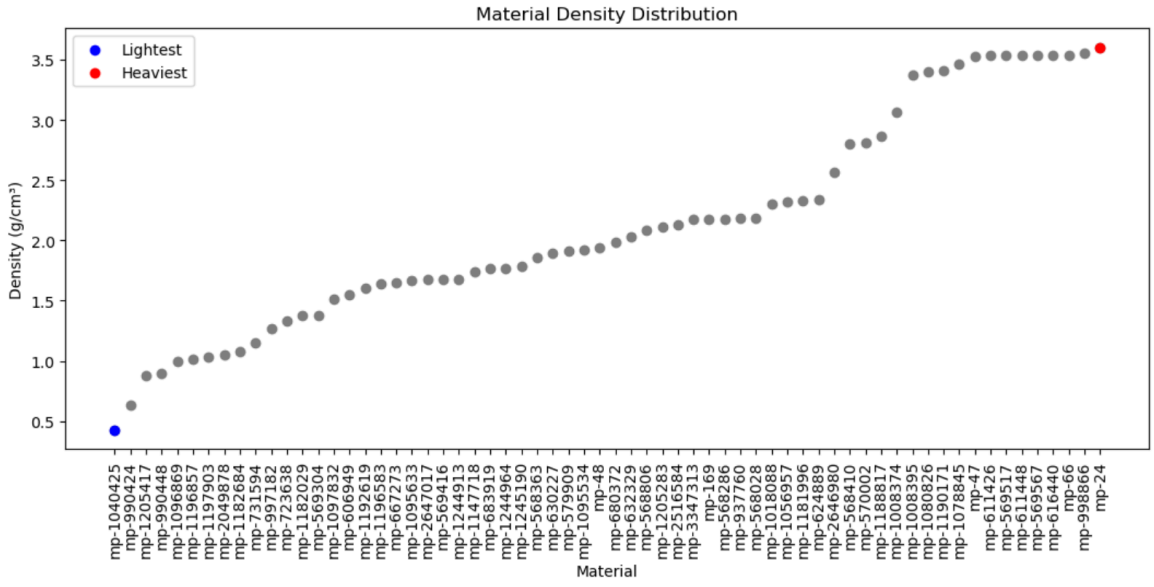
C

```
enter new chemical system:
 C
Materials Property Explorer
==========================

Sorting by density...
Retrieving SummaryDoc documents: 100% 64/64 [00:00<00:00, 9016.37it/s]

Exploration complete!
```

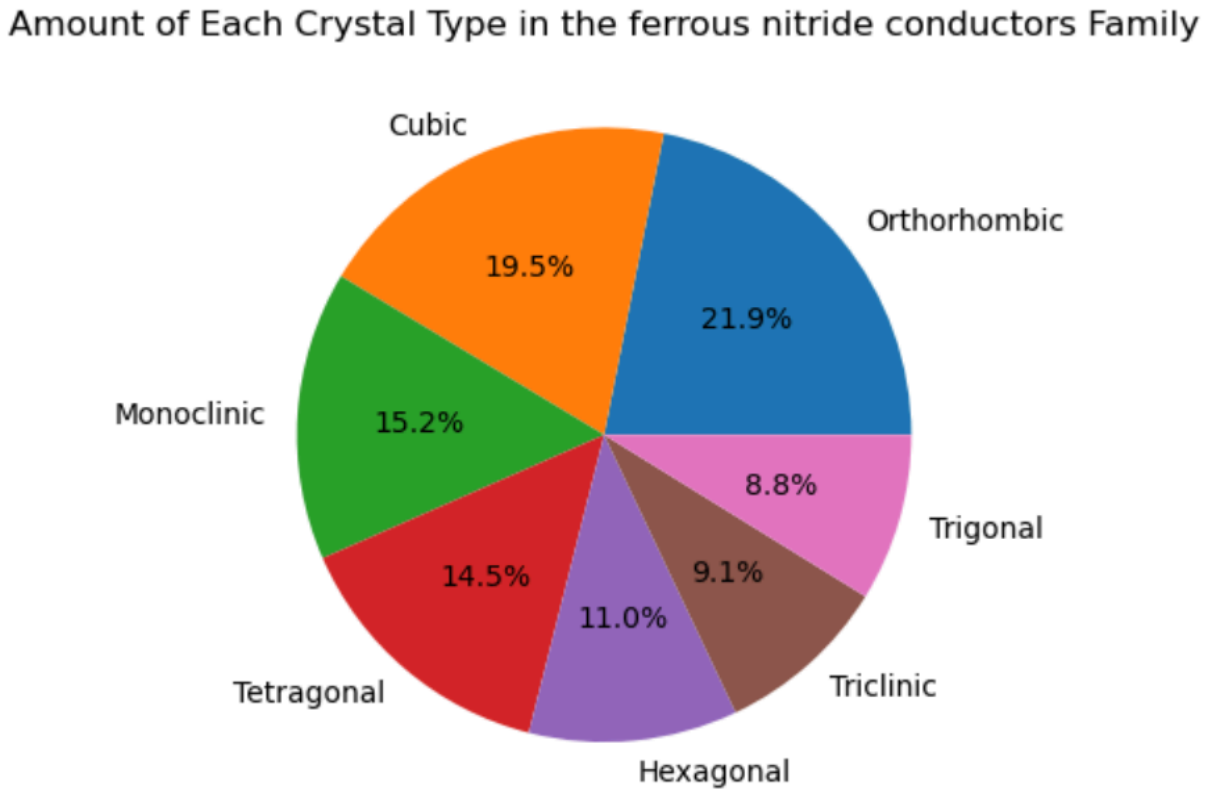


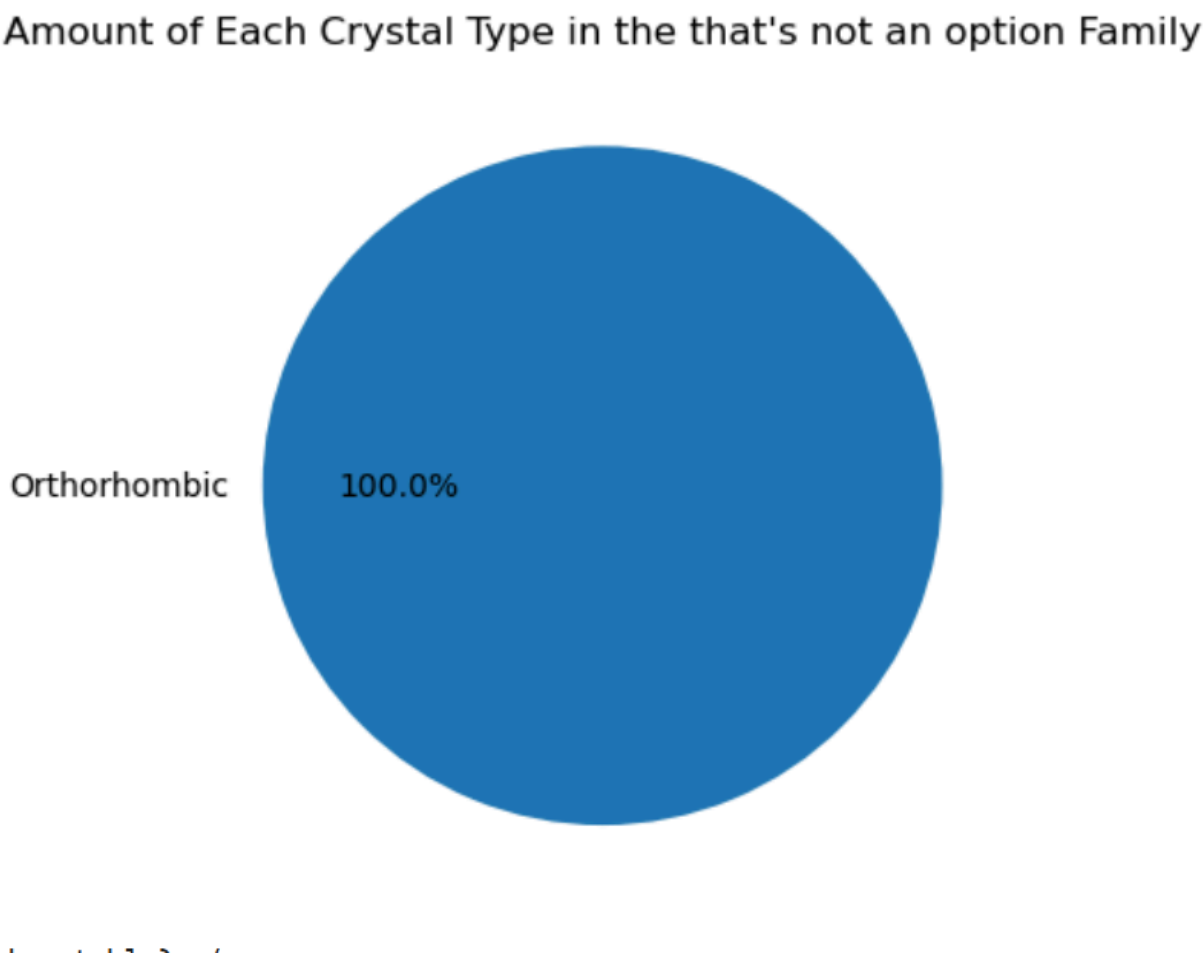


The comments in the code should explain what's going on pretty well.

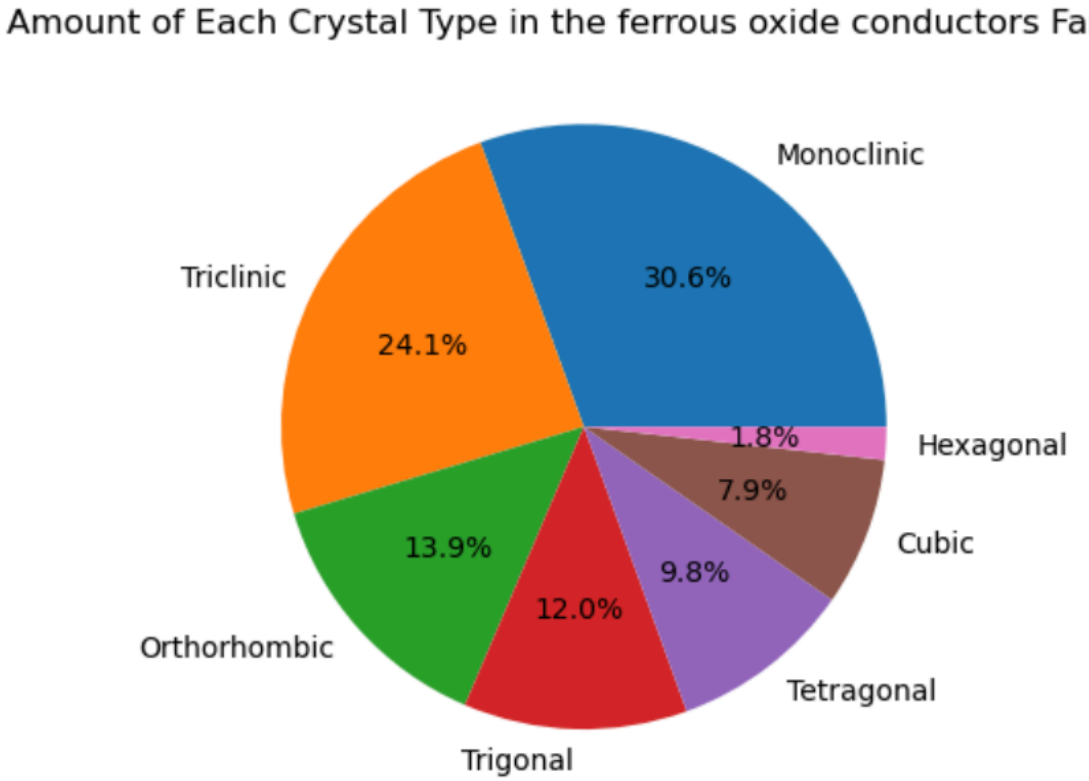
Second script:
Visualizes the distribution of crystal systems within a given material family using pie charts
Using the keyword system and also the chemical system selection tool you can get pretty broad or specific here. I would recommend being specific for faster loading times though. If you don't put any filters in the system will stop you from loading all of the materials. You will instead load one very specific material. It runs off of the same (or similar) systems to the last one but just with a lot more choice in how you sort through things and also obviously returning different information. I think the other one was a lot easier to make sure you didn't get like 80000 results or something though. Also, the title of the graph changes based on what you searched.

```
Exploration complete!
```





I wanted to put more loading things at some big numbers incase you accidentally did a combination that resulted in a really big number of materials but it ended up slowing it down a lot and just being a pain to deal with so I had to get rid of it.



I was going to put the world's shortest poem in the actual code for the first one but Jupyter lab disliked my method of putting an image in so it's going here: