



A Python-powered logical system and GUI application designed to solve stoichiometric equations, by breaking down each step as you would write them on a piece of paper. Thus building the needed knowledge — all for free. Stoichify is open-source and aimed at chemistry students and teachers to learn and better their experience with the daunting complexity of stoichiometry.

GitHub: <https://github.com/KingPr0o7/Stoichify>

USER MANUAL

Built and Designed by **Nathan Parker**
... with help from other contributors

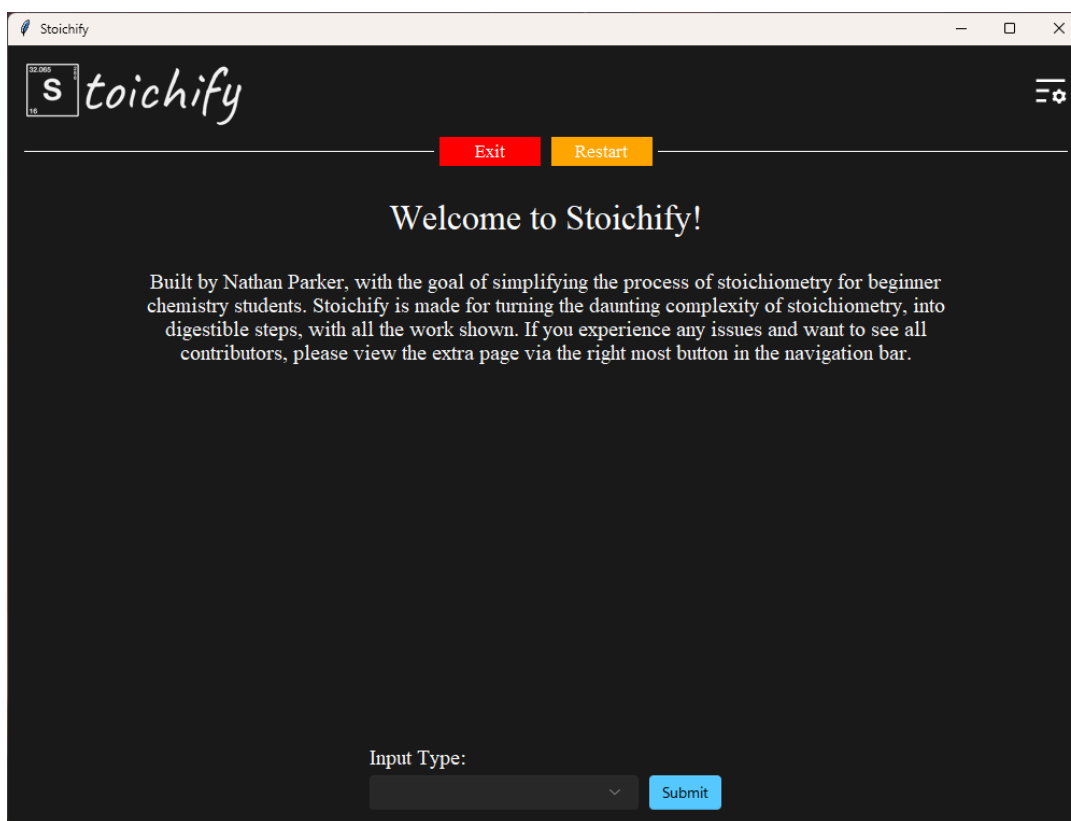
Table of Contents

How to Use Stoichify	3
Stage 1 (Choosing Type — Substance / Equation)	4
Stage 2 (Inputting Substance/Equation)	7
Stage 3 (Stoichiometric Calculations)	10
Stage 4 (Results + Loop Back)	13
Testing Stoichify	14
Contributors	21

How to Use Stoichify

The main goal of this project was to make things as simple as possible. Thus, all aspects of the GUI and most methods if called-upon in a CLI-type of way have standard explanations (doc-strings, comments, and descriptions) to show exactly what's needed and being done. However, this user manual will only show how to use the GUI that I have built¹.

[That being said, after you install Stoichify](#) and have it running on your desktop, you'll be greeted with a window that looks like **Figure 1**:



(Figure 1)

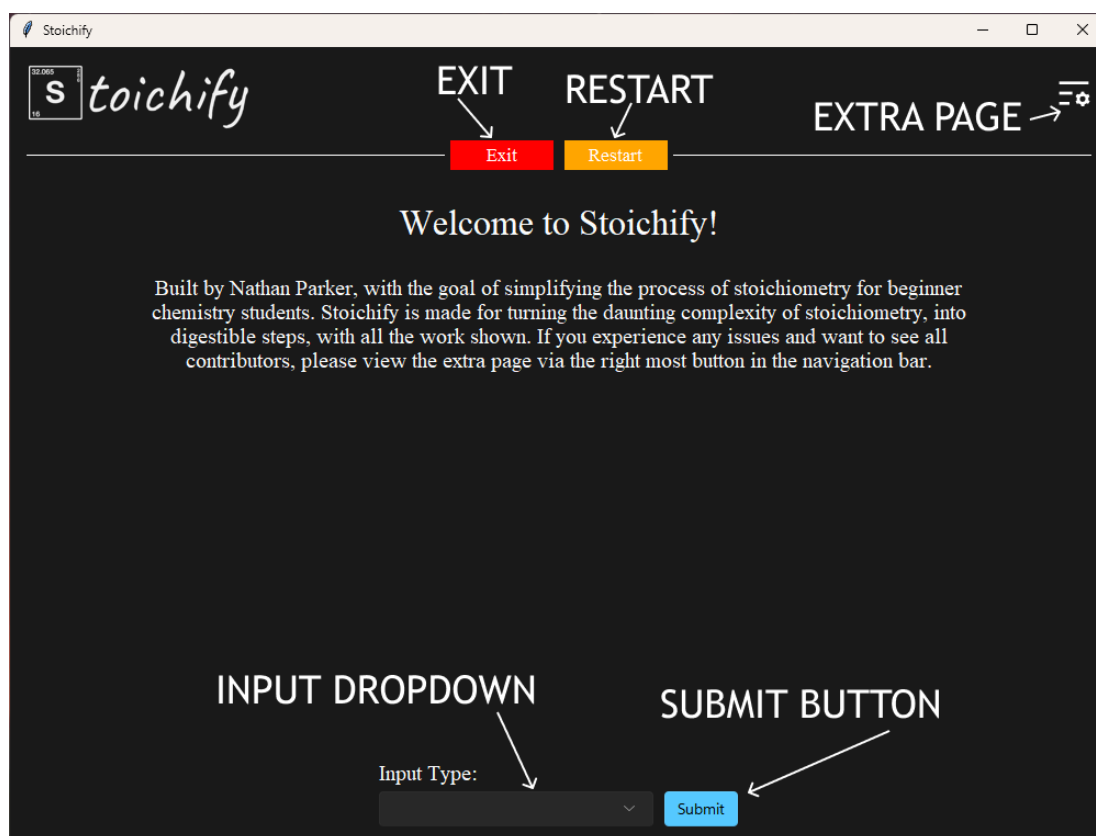
¹ Might change if more community interest is accumulated.

Stage 1 (Choosing Type – Substance / Equation)

With Stage 1, you're presented with 5 intractable elements, which are:

1. Extra Page Button (Navigation Bar)
2. Exit Button (Navigation Bar)
3. Restart Button (Navigation Bar)
4. Input Selection Box (Input Panel)
5. Input Submit Button (Input Panel)

Below is **Figure 2** which highlights the above listed elements in their exact positions:

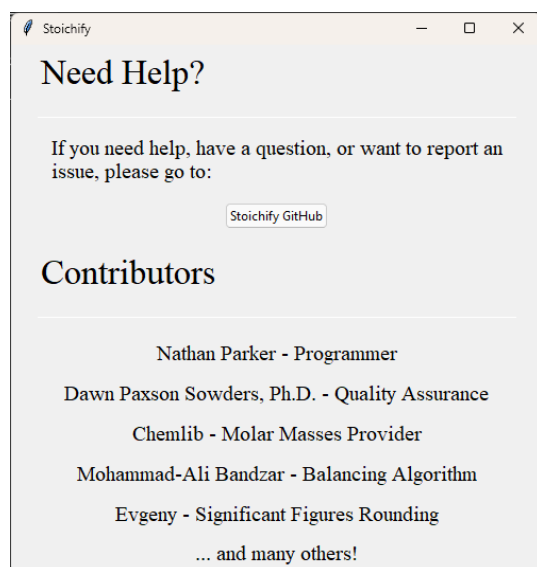


(Figure 2)

Any element located in the *Navigation Bar* will never **disappear** on you. The *Navigation Bar* is distinguished by lines next to the *Exit Button* and *Restart Button*. Anything after those lines are considered *Content Elements* and will change on you after each stage (called upon clicking the *Submit Button*). Unfortunately, I had no time to account for tabbing to elements, which will be added *eventually*.

Starting off with the two *Navigation buttons*, they consist of the *Exit Button* and *Restart Button*. In which, the *Exit Button* will simply close Stoichify, leading you back to your preferred coding editor². With the *Restart Button*, it'll take you back to *Stage 1*, regardless of what stage you're currently at. It'll **never close or kill variable instances**.

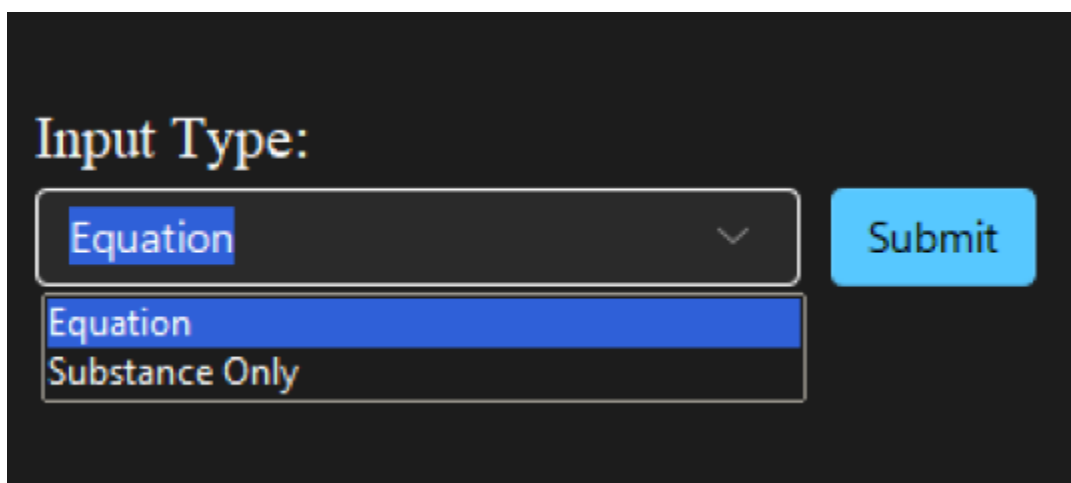
The '*Extra*' *Page Button* opens another window with a link to the Stoichify GitHub (in case of any issues) and a list of some top contributors, shown in **Figure 3**:



(Figure 3)

² May change if Stoichify is bundled differently.

Getting onto the *Input Panel*, this is where you'll be interacting with the most. All required inputs are always located at the bottom of the screen. They change based on stage and this input type presented on *Stage 1*. There are two types of ways that Stoichify can handle, being *Equations* or a *Substance*. If you decide to choose an *Equation*, you'll be required to put in an equation in the desired format, whereas if you chose *Substance Only*, you'll need a substance to move on. This choice is presented in the dropdown (**Figure 4**) and upon submitting you'll be moved into *Stage 2* for that type:



The image shows a dark-themed user interface. At the top left, the text "Input Type:" is displayed in a light blue font. Below it is a dropdown menu with a white border. The menu is currently open, showing two options: "Equation" and "Substance Only". The "Equation" option is highlighted with a blue background. To the right of the dropdown menu is a red button with the word "Submit" in white text.

(Figure 4)

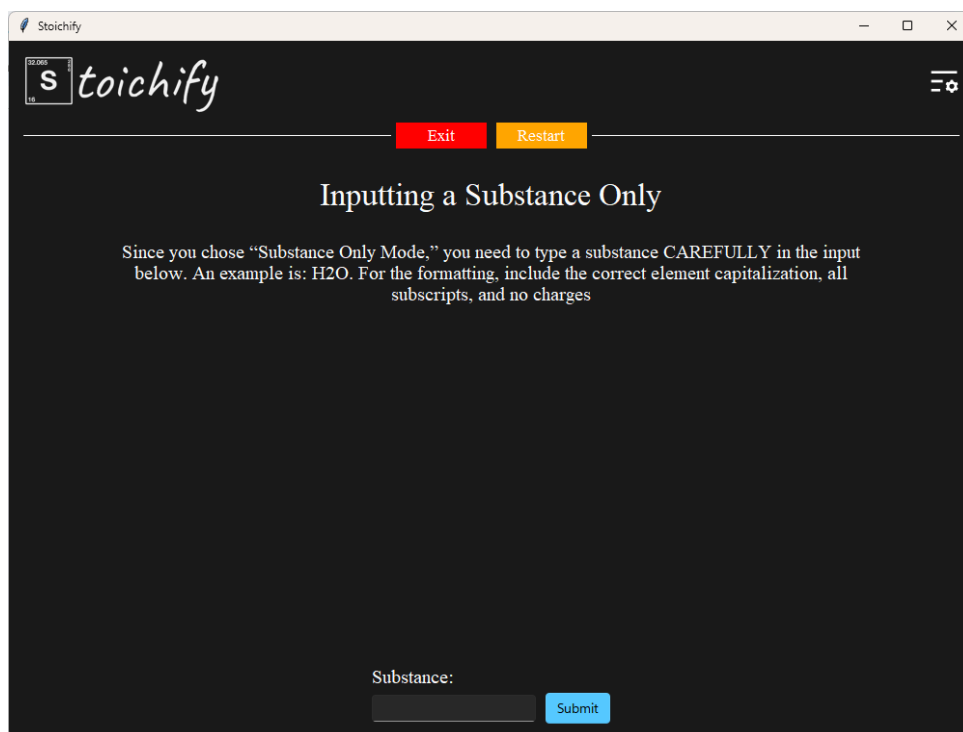
Stage 2 (Inputting Substance/Equation)

After selecting one of the types, either *Equation* (**Figure 5**) or *Substance Only* (**Figure 6**), you'll move on to *Stage 2*:



The screenshot shows a web browser window with the title "Stoichify". The page has a dark background with a logo in the top left corner consisting of a periodic table element box with the letter 'S' and the word "toichify" in a script font. In the top right corner, there is a settings icon. Below the header, there are two buttons: "Exit" (red) and "Restart" (orange). The main heading is "Inputting a Chemical Equation". Below this, there is a paragraph of instructions: "Since you chose 'Equation Mode,' you need to type an equation CAREFULLY in the input below. Any mistyping will result in incorrect balancing, affecting the rest of the calculation. An example is: $\text{Fe} + \text{O}_2 \rightarrow \text{Fe}_2\text{O}_3$. For the formatting, include pluses between substances, correct element capitalization, all subscripts, an arrow of some sort (UNICODE or a dash and greater than ">"), and no charges (oxidation-reactions aren't supported)." At the bottom, there is a label "Equation:" followed by a text input field and a blue "Submit" button.

(Figure 5 — Equation Type)



(Figure 6 – Substance Type)

Follow the instructions and input into the *Input Entries*. Stoichify doesn't support charges which lead to oxidation-reactions that have a different balancing method. *Equations* allow for pasting of:

- Any arrow type from UNICODE in case your assignment/website uses them (99% likely).
- Any subscript included
- Any substance state (s, l, g, aq)
- Substances surrounded by: [] (complex; ignored)

Which covers most bases; however, if you're typing raw into Stoichify you can type regular numbers for subscripts and use the arrow "->" (dash and greater than sign) to represent the yield arrow (it'll be replaced). The same general rules **also apply to *Substances***. Just then you don't have to type arrows, pluses, reactants, or products as you're only dealing with

one substance. If you mess up, errors will appear, allowing you to fix your mistakes³. Here are examples of equations and substances that'll pass the checks:

- $\text{SO}_2 + \text{O}_2 \rightarrow \text{SO}_3$
- $\text{Fe} + \text{O}_2 \rightarrow \text{Fe}_2\text{O}_3$
- $\text{Fe}_2 + \text{O}_2 \rightarrow \text{Fe}_2\text{O}_3$
- $\text{F}_2(\text{g})$
- F_2
- $\text{H}_2(\text{g}) + \text{O}(\text{g}) \rightarrow \text{H}_2\text{O}(\text{l})$
- $\text{K}_4[\text{Fe}(\text{SCN})_6] + \text{K}_2\text{Cr}_2\text{O}_7 + \text{H}_2\text{SO}_4 \rightarrow \text{Fe}_2(\text{SO}_4)_3 + \text{Cr}_2(\text{SO}_4)_3 + \text{CO}_2 + \text{H}_2\text{O} + \text{K}_2\text{SO}_4 + \text{KNO}_3$
- Co_2
- CO_3

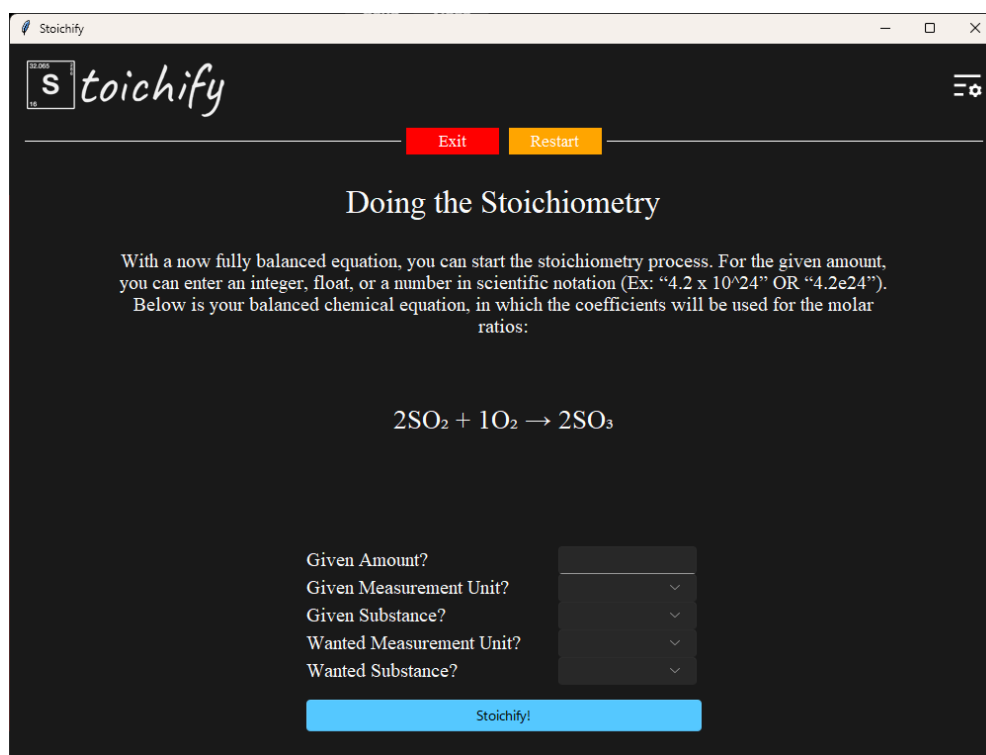
And more. Stoichify's type checking is quite strict, and should⁴ catch most errors (if you do encounter an error, and can't fix it, report it to GitHub). After submission of your *Equation* or *Substance*, you'll move on to a unified *Stage 3*.

³ Stoichify isn't perfect, and might not catch every error.

⁴ Stoichify isn't perfect, and might not catch every error.

Stage 3 (Stoichiometric Calculations)

Now with the *Equation* or *Substance* is saved, Stoichify will automatically format and balance equations (**Figure 7**), or show your formatted substance. If you choose a substance, the fields of the *Given Substance?* and *Wanted Substance?* will already be filled out (**Figure 8**).



The screenshot shows the Stoichify web application interface. At the top, there is a header with the logo "stoichify" and a settings icon. Below the header, there are two buttons: "Exit" (red) and "Restart" (yellow). The main section is titled "Doing the Stoichiometry". It contains a paragraph explaining the process: "With a now fully balanced equation, you can start the stoichiometry process. For the given amount, you can enter an integer, float, or a number in scientific notation (Ex: '4.2 x 10^24' OR '4.2e24'). Below is your balanced chemical equation, in which the coefficients will be used for the molar ratios:". The balanced chemical equation is displayed as $2\text{SO}_2 + 1\text{O}_2 \rightarrow 2\text{SO}_3$. Below the equation, there are five input fields with labels: "Given Amount?", "Given Measurement Unit?", "Given Substance?", "Wanted Measurement Unit?", and "Wanted Substance?". The "Given Substance?" and "Wanted Substance?" fields are pre-filled with "SO2" and "SO3" respectively. At the bottom, there is a blue button labeled "Stoichify!".

(Figure 7)

Stoichify

stoichify

Exit Restart

Doing the Stoichiometry

With a now fully balanced equation, you can start the stoichiometry process. For the given amount, you can enter an integer, float, or a number in scientific notation (Ex: "4.2 x 10²⁴" OR "4.2e24").

Below is your substance:

F₂

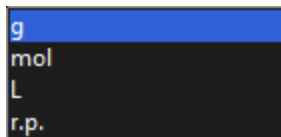
Given Amount?
 Given Measurement Unit?
 Given Substance?
 Wanted Measurement Unit?
 Wanted Substance?

Stoichify!

(Figure 8)

The first input (*Given Amount?*), an *Entry* requires you to know the given amount, which can either be a float, an integer, or a number in scientific notation (4.2 x 10²⁴, 4.2 * 10²⁴, 4.2e24, 4.2e-24, etc.). This entry, upon submission, will count your significant figures for you (even in scientific notation (if parsed correctly)). *Unfortunately*, if you adapt Stoichify into a CLI type application, it requires you to know the given significant figures... As Python and other languages strip zeros, so you'll have to find a way to keep them via keystrokes (like tkinter) and build a string representation.

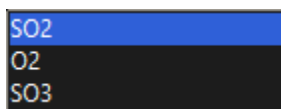
For the next inputs, they'll always be dropdowns... Which consist of the *given measurement unit*, *given substance*, *wanted measurement unit*, and *wanted substance*. Both measurement unit dropdown consists of (*Figure 9*):



(Figure 9)

With g (grams), mol (moles), L (liters), and r.p. (representative particles, like atoms, molecules, etc.) In terms of *r.p.* it's a general representation of particles.

For the *Given Substance?* and *Wanted Substance?* In substance mode, as already mentioned, will already be filled out (as there are no other substances). However, in an equation there are multiple substances, in which will be presented for you to choose from (don't worry, their coefficients are saved) and, as an example, would look like (**Figure 10**):

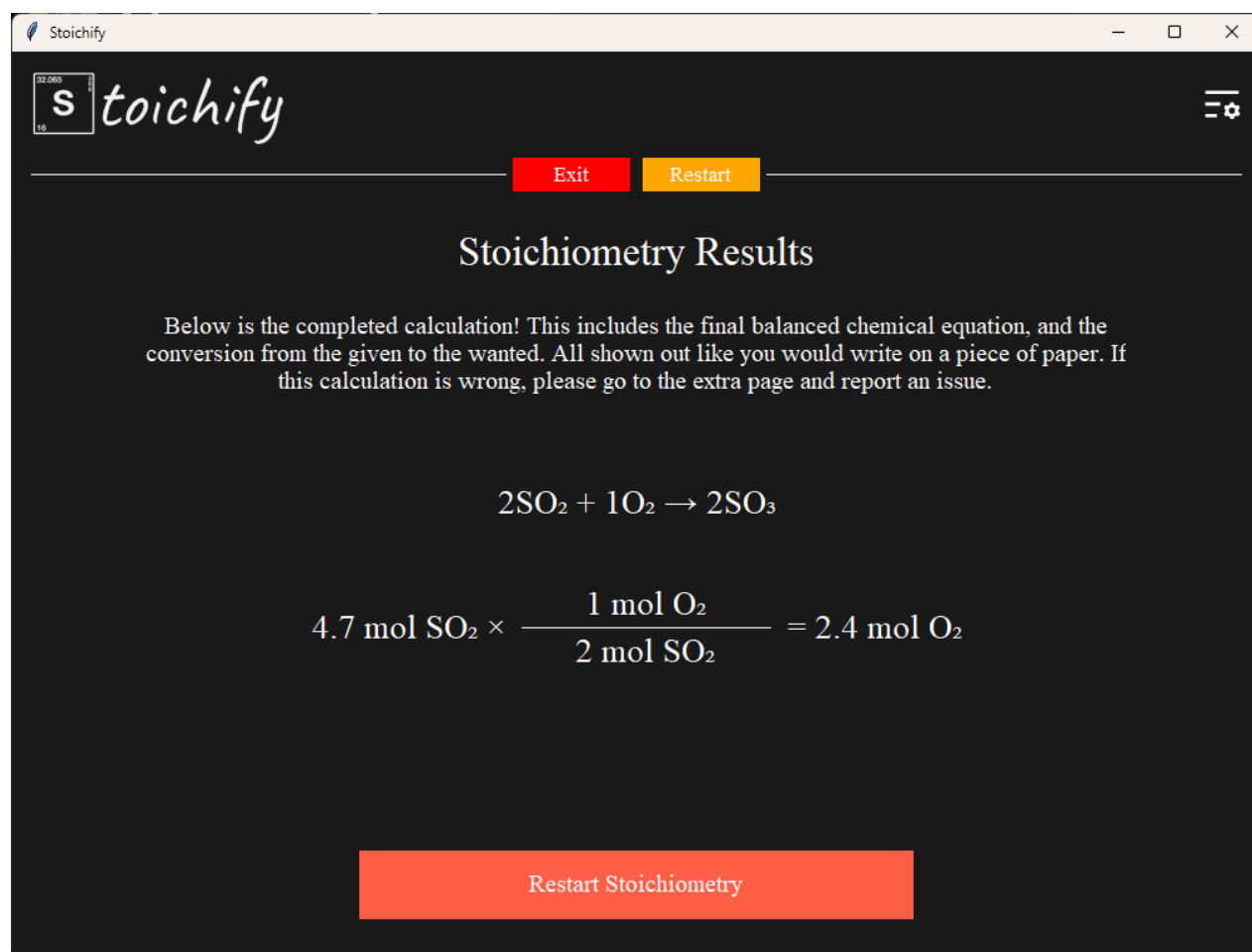


(Figure 10)

After all values are inputted (with none blank (it'll throw an error)) you can click on the submit button (appropriately named "Stoichify!") you'll move on to *Stage 4*, the final stage with the results!

Stage 4 (Results + Loop Back)

Upon clicking the “Stoichify!” button from *Stage 3*, the GUI (*main.py*) will call the **Stoichify** class (*stoichiometry.py*) which will water fall calls of other classes to calculate the results. During this process, it’ll build a list of values, that show the program saving its work along the way. Upon completion of this, you’ll be brought to this screen (**Figure 11**):



(Figure 11)

Which will show your final balanced equation / substance and the work shown as a student/teacher would write on a piece of paper! After you’re

satisfied with the results, you can click the “Restart Stoichiometry” button to go back to the saved balanced equation / substance and do many other stoichiometric calculations. Saving typing time and bringing a quicker experience!

Testing Stoichify

Stoichify was heavily tested during the development and somewhat after development. Comments in source code were some tests and hurdles I had to jump over... However, I made the critical mistake of starting the *testing.py* file, until after development... Regardless, there are **60** tests against Stoichify ensuring type security and consistency with answers across the board. Such test data was gathered from the following sources:

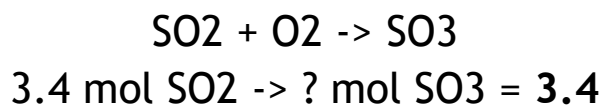
- <https://www.youtube.com/watch?v=7Cfq0ilw7ps>
- <https://www.mcmsnj.net/cms/lib07/NJ01911694/Centricity/Domain/540/Extra%20Practice%20mole%20mass%20rp%20practice%20problems%20answers.pdf>
- <https://www.youtube.com/watch?v=74-X94OP2XI&t=596s>
- <https://www.youtube.com/watch?v=XnfATaoubzA>
- <https://www.youtube.com/watch?v=VJWTnwl1V0g>
- ... and many more!

However, it's important to note that some of these sources may not *consider some variables*. Like significant figures (might be an introductory video), have the same molar mass (might be using an out-of-date periodic

table), and different rounding. Stoichify uses a newer periodic table and has strict significant figure counting and rounding (rounds up). Here are some examples of Stoichify in action:

The image displays two side-by-side windows. The left window is a YouTube video player showing a chemistry tutorial. The video title is "Stoichiometry Basic Introduction, Mole to Mole, Grams to Grams, Mole Ratio Practice Problems" by "The Organic Chemistry Tutor". The video content shows handwritten chemical equations and calculations. The right window is the "stoichify" web application. It shows the chemical equation $2\text{SO}_2 + 1\text{O}_2 \rightarrow 2\text{SO}_3$ and the calculation $3.4 \text{ mol SO}_2 \times \frac{2 \text{ mol SO}_3}{2 \text{ mol SO}_2} = 3.4 \text{ mol SO}_3$. The application also has buttons for "Exit", "Restart", and "Restart Stoichiometry".

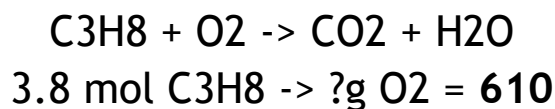
(Figure 12)



(taking given * molar ratio of given/wanted coefficients)

The image shows two side-by-side windows. The left window is a YouTube video titled 'Stoichiometry Basic Introduction, Mole to Mole, Grams to Grams, Mole Ratio Practice Problems' by 'The Organic Chemistry Tutor'. It displays a handwritten chemical equation: $1\text{C}_3\text{H}_8 + 5\text{O}_2 \rightarrow 3\text{CO}_2 + 4\text{H}_2\text{O}$. Below the equation, a calculation is shown: $\frac{3.8\text{ mol C}_3\text{H}_8}{1} \times \frac{5\text{ mol O}_2}{1\text{ mol C}_3\text{H}_8} \times \frac{32\text{ g O}_2}{1\text{ mol O}_2} = 608\text{ g O}_2$. The right window is a web application called 'stoichify'. It shows the same chemical equation: $1\text{C}_3\text{H}_8 + 5\text{O}_2 \rightarrow 3\text{CO}_2 + 4\text{H}_2\text{O}$. Below it, the calculation is presented in a digital format: $3.8\text{ mol C}_3\text{H}_8 \times \frac{5\text{ mol O}_2}{1\text{ mol C}_3\text{H}_8} \times \frac{31.998\text{ g O}_2}{1\text{ mol O}_2} = 610\text{ g O}_2$. The application includes buttons for 'Exit', 'Restart', and 'Restart Stoichiometry'.

(Figure 13)

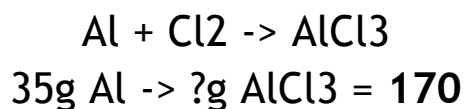


(Take the given * molar ratio of given/wanted coefficients * a conversion to wanted measurement unit)

(The Organic Chemistry Teacher didn't account for Significant Figures, and the defined user measurement is two significant figures. So, Stoichify rounded up.)

The image shows two side-by-side windows. The left window is a YouTube video titled 'Stoichiometry Basic Introduction, Mole to Mole, Grams to Grams, Mole Ratio Practice Problems' by 'The Organic Chemistry Tutor'. It features handwritten calculations for the reaction $2\text{Al} + 3\text{Cl}_2 \rightarrow 2\text{AlCl}_3$. The calculation starts with 35g Al, converts it to moles (35g / 26.98g/mol = 1.299 mol Al), uses the molar ratio (2 mol Al : 2 mol AlCl₃) to find moles of AlCl₃ (1.299 mol AlCl₃), and finally converts it to grams (1.299 mol AlCl₃ * 133.33g/mol = 172.96g AlCl₃). The right window is the 'stoichify' web application. It shows the same reaction $2\text{Al} + 3\text{Cl}_2 \rightarrow 2\text{AlCl}_3$ and a calculation: $35\text{g Al} \times \frac{1 \text{ mol Al}}{26.982\text{g Al}} \times \frac{2 \text{ mol AlCl}_3}{2 \text{ mol Al}} \times \frac{133.341 \text{ g AlCl}_3}{1 \text{ mol AlCl}_3} = 170\text{g AlCl}_3$. The application has buttons for 'Exit', 'Restart', and 'Restart Stoichiometry'.

(Figure 14)



(Take the given measurement unit, convert it to moles, * by molar ratio of given/wanted coefficients, and convert back to grams)

(Same thing, Stoichify noticed 2 significant figures, and rounded to it)

Extra Practice mole, mass, rp...

1 / 1

100%

Stoichify

Exit Restart

Stoichiometry Results

Below is the completed calculation! This includes the final substance, and the conversion from the given to the wanted. All shown out like you would write on a piece of paper. If this calculation is wrong, please go to the extra page and report an issue.

OCl₂

$$392.1 \text{ g OCl}_2 \times \frac{1 \text{ mol OCl}_2}{86.905 \text{ g OCl}_2} \times \frac{6.02 \times 10^{23} \text{ r.p. OCl}_2}{1 \text{ mol OCl}_2} = 2.716 \times 10^{24} \text{ r.p. OCl}_2$$

Restart Stoichiometry

Honors Chemistry

Extra Practice Problems - The Mole and Conversions with Mass and Representative Particles

Complete the following problems on a separate sheet of paper.

- Determine the mass of Lithium Carbonate in 4.020 moles of Lithium Carbonate?

$$\frac{4.020 \text{ moles Li}_2\text{CO}_3}{1} \times \frac{73.89 \text{ g Li}_2\text{CO}_3}{1 \text{ mole Li}_2\text{CO}_3} = 297.0 \text{ g Li}_2\text{CO}_3$$
- Determine the number of representative particles and the type of representative particles in 392.1 g of Oxygen Dichloride.

$$\frac{392.1 \text{ g OCl}_2}{1} \times \frac{1 \text{ mole OCl}_2}{86.905 \text{ g OCl}_2} \times \frac{6.02 \times 10^{23} \text{ molecules OCl}_2}{1 \text{ mole OCl}_2} = 2.716 \times 10^{24} \text{ molecules OCl}_2$$
- Determine the number of moles in 93.20 g of Carbon tetrahydride.

$$\frac{93.20 \text{ g CH}_4}{1} \times \frac{1 \text{ mole CH}_4}{16.043 \text{ g CH}_4} = 5.809 \text{ mole CH}_4$$
- Determine the mass of 9.3021 x 10²⁷ representative particles of Fluorine. What is the representative particle?

$$\frac{9.3021 \times 10^{27} \text{ molecules F}_2}{1} \times \frac{1 \text{ mole F}_2}{6.02 \times 10^{23} \text{ molecules F}_2} \times \frac{37.996 \text{ g F}_2}{1 \text{ mole F}_2} = 587.110 \text{ g F}_2 \text{ OR } 5.8711 \times 10^5 \text{ g F}_2$$
- Determine the mass of 3.902 x 10²⁶ representative particles of Ammonium Chloride. What is the representative particle?

$$\frac{3.902 \times 10^{26} \text{ f.u. NH}_4\text{Cl}}{1} \times \frac{1 \text{ mole NH}_4\text{Cl}}{6.02 \times 10^{23} \text{ f.u. NH}_4\text{Cl}} \times \frac{53.492 \text{ g NH}_4\text{Cl}}{1 \text{ mole NH}_4\text{Cl}} = 3,467,000 \text{ g NH}_4\text{Cl OR } 3.467 \times 10^6 \text{ g NH}_4\text{Cl}$$
- How many moles of Iron III Carbonate are there in 9.032 x 10²⁰ representative particles of Iron III Carbonate?

$$\frac{9.032 \times 10^{20} \text{ f.u. Fe}_2(\text{CO}_3)_3}{1} \times \frac{1 \text{ mole Fe}_2(\text{CO}_3)_3}{6.02 \times 10^{23} \text{ f.u. Fe}_2(\text{CO}_3)_3} = 1.500 \times 10^{-3} \text{ mole Fe}_2(\text{CO}_3)_3$$
- How many representative particles are there in 490.2 grams of Copper I Oxide?

$$\frac{490.2 \text{ g Cu}_2\text{O}}{1} \times \frac{1 \text{ mole Cu}_2\text{O}}{143.091 \text{ g Cu}_2\text{O}} \times \frac{6.02 \times 10^{23} \text{ f.u. Cu}_2\text{O}}{1 \text{ mole Cu}_2\text{O}} = 2.062 \times 10^{24} \text{ f.u. Cu}_2\text{O}$$

(Figure 15)



$$392.1 \text{ g OCl}_2 \rightarrow ? \text{ r.p. OCl}_2 = 2.716 \times 10^{24}$$

(Take given into moles, then to r.p. (molar ratios don't exist in just substances))

Stoichify

Extra Practice mole, mass, r.p. pra... 1 / 1 100% +

3. Determine the number of moles in 93.20 g of Carbon tetrahydride.

$$\frac{93.20 \text{ g } CH_4}{1} \times \frac{1 \text{ mole } CH_4}{16.043 \text{ g } CH_4} = 5.809 \text{ mole } CH_4$$

4. Determine the mass of 9.3021×10^{27} representative particles of Fluorine. What is the representative particle?

$$\frac{9.3021 \times 10^{27} \text{ molecules } F_2}{1} \times \frac{1 \text{ mole } F_2}{6.02 \times 10^{23} \text{ molecules } F_2} \times \frac{37.996 \text{ g } F_2}{1 \text{ mole } F_2} = 587,110 \text{ g } F_2 \text{ OR } 5.8711 \times 10^5 \text{ g } F_2$$

5. Determine the mass of 3.902×10^{28} representative particles of Ammonium Chloride. What is the representative particle?

$$\frac{3.902 \times 10^{28} \text{ f.u. } NH_4Cl}{1} \times \frac{1 \text{ mole } NH_4Cl}{6.02 \times 10^{23} \text{ f.u. } NH_4Cl} \times \frac{53.492 \text{ g } NH_4Cl}{1 \text{ mole } NH_4Cl} = 3,467,000 \text{ g } NH_4Cl \text{ OR } 3.467 \times 10^6 \text{ g } NH_4Cl$$

6. How many moles of Iron III Carbonate are there in 9.032×10^{30} representative particles of Iron III Carbonate?

$$\frac{9.032 \times 10^{30} \text{ f.u. } Fe_2(CO_3)_3}{1} \times \frac{1 \text{ mole } Fe_2(CO_3)_3}{6.02 \times 10^{23} \text{ f.u. } Fe_2(CO_3)_3} = 1.500 \times 10^7 \text{ mole } Fe_2(CO_3)_3$$

7. How many representative particles are there in 490.2 grams of Copper I Oxide?

$$\frac{490.2 \text{ g } Cu_2O}{1} \times \frac{1 \text{ mole } Cu_2O}{143.091 \text{ g } Cu_2O} \times \frac{6.02 \times 10^{23} \text{ f.u. } Cu_2O}{1 \text{ mole } Cu_2O} = 2.062 \times 10^{24} \text{ f.u. } Cu_2O$$

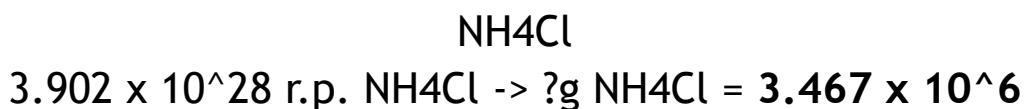
Stoichiometry Results

alculation! This includes the final substance, and the conversion from the hown out like you would write on a piece of paper. If this calculation is ng, please go to the extra page and report an issue.

NH_4Cl

Restart Stoichiometry

(Figure 16)



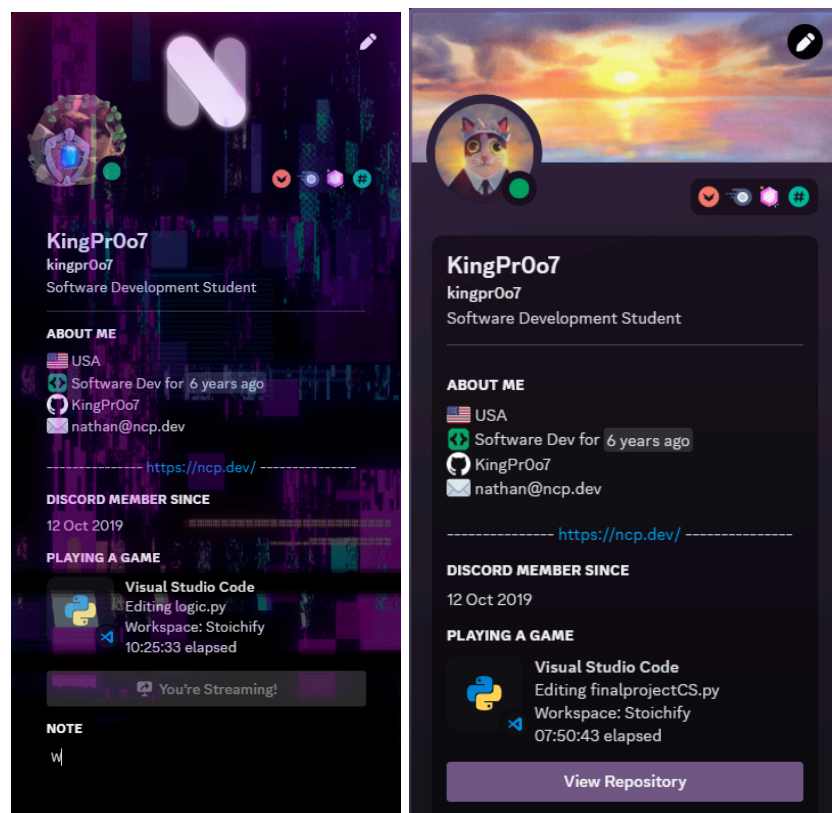
(Go from r.p. to moles, then to grams)

... and many more! Try out Stoichify for yourself to see the results! Please! If you find any errors, want to suggest features, and more, please go to the GitHub: <https://github.com/KingPr0o7/Stoichify>

The work shown by these people brought up many errors that I had to fix along the way. For example, scientific numbers were a real pain. Because I also had to find the significant figures in them as well. Along with parsing unique ways to format them, and even the float representation of them.

Other things I had to fix were correct balancing, formatting, GUI bugs, and many more. However, the biggest pain, by far, was parsing significant figures. I had gone to many people with my frustrations of Python cutting off my precious significant figures (it haunts me still to this day) until I realized that tkinter saved me, by building a string representation of all characters entered.

All these contributed to a lot of time debugging, asking for help, and hours and hours of Googling my way through issues. But nonetheless, I pushed forward, through the hours (sometimes 10+ hours at a time) (**Figure 16-7**) to create something for the public to enjoy. I want to clarify it, that this project is open to the public, and anyone can report issues to better Stoichify.



(Figures 16 + 17)

Contributors

Even though this project was meant to be a single-person project for Ivy Tech, Stoichify cannot be what it is today without the help from these people below, please give them the credit they deserve. They truly made Stoichify one-of-a-kind.

Dawn Paxson Sowders, Ph.D. — Quality Assurance

[Chemlib](#) Creators — Molar Masses Provider

Mohammad-Ali Bandzar — [Balancing Algorithm](#)

Evgeny — [Significant Figures Rounding \(basis\)](#)

Benedek Dévényi (rdbende) et al. — [Tkinter Theme \(sv_ttk\)](#)

Jeffrey A. Clark et al. — [Pillow \(PIL\) Image Library](#)

SymPy Development Team — [Symbolic Mathematics](#)

Thank you!

This project was in development and **will** be maintained by Nathan Parker (KingPr0o7). I'm a software development student in Indiana, trying to pursue my dreams of making software for the public to enjoy. Follow me on [X](#) to get updates about myself.

