Code Documentation for Chem 1 Creative Output Midyear 2025

Jezic Abueg Chem 1 X1

https://jfabuchem1creativeoutput.on.drv.tw/www.chem1.com/

Main logic (and chem concepts):

stoich.js

(This also uses <u>chemcalc.js</u> which is a javascript library that allows getting information about molecules and toms)

Converting within the same molecule/atom:

```
// mol to grams
function convertMolToGrams(mols, molar_mass) {
    return mols * molar_mass;
}

// mol to amount
function convertMolToAmount(mols) {
    return mols * AVOGADRO;
}

// grams to mol
function convertGramsToMol(grams, molar_mass) {
    return grams * 1/(molar_mass);
}

// grams to amount
function convertGramsToAmount(grams, molar_mass) {
    return convertMolToGrams(convertAmountToMol(amount), molar_mass);
}

// grams to amount
function convertGramsToAmount(grams, molar_mass) {
    return convertMolToGrams(convertAmountToMol(amount), molar_mass);
}

// molecule to its atoms
function convertMolEcToItsAtoms(molec, ratio) {
    return molec * ratio;
}
```

The functions above are used to convert from one unit to another. From lessons, we know that the general way to convert is "grams->mol->amount". This is reflected in the functions like how "convertGramsToAmount" needs to call "convertGramsToMoll" then "convertMolToAmount". This is also seen in "convertAmountToGrams".

As such, we apply these functions as seen in the following code snippets, wherein we check the starting and ending unit, and convert as such.

```
} else if (starting_unit == "mol") {
    if (ending_unit == "grams") {...
    } else if (ending_unit == "atoms/molecules") {
        result_ = convertToLatexSciNotation(convertMolToAmount(rawInput));
        ansLatex = `Answer: ${result_}~atoms/molecules~${element}`;

else if (starting_unit == "atoms/molecules") {
    if (ending_unit == "grams") {
        result_ = convertToLatexSciNotation(convertAmountToGrams(rawInput, result.mw));
        ansLatex = `Answer: ${result_}~g~${element}`;

else if (starting_unit == "atoms/molecules") {
    if (ending_unit == "grams") {...
    } else if (ending_unit == "mol") {
        result_ = convertToLatexSciNotation(convertAmountToMol(rawInput));
        ansLatex = `Answer: ${result_}~mol~${element}`;
    }
}
```

Converting from a molecule to one of its atoms:

```
// molecule to its atoms
/ function convertMolecToItsAtoms(molec, ratio) {
    return molec * ratio;
}
```

```
// get the number of each atom for ratios
const atoms = result.parts[0].ea;

// make a new dictionary from the info above for ease
const atoms_dict = new Object;
var atoms_dictLatex = "With these no. of atoms: ";
for (let i = 0; i < atoms.length; i++) {
    atoms_dict[atoms[i].element] = atoms[i].number;
    atoms_dictLatex += atoms[i].number + " " + atoms[i].element + " atom(s), ";
}</pre>
```

The process of converting here is the same as the one above with just an added step. Here, everything is first converted to its "amount", then multiplied by the ratio (atoms/molecule) to get the number of atoms, which is then converted to whatever unit the user specifies.

As such, we can see those in action in the following code snippets. Again, we convert from the starting unit of the molecule to the ending unit of the atom within that molecule.

```
if (element.includes(element_2) && element_2 in atoms_dict) {
    good_input = true;
    var ratio = atoms_dict[element_2];
    if (starting_unit == "grams") {
        if (ending_unit == "mol") {
            result_ = convertGramsToAmount(rawInput, result.mw);
            result_ = convertMolecToItsAtoms(result_, ratio);
            result_ = convertToLatexSciNotation(convertAmountToMol(result_));

if (starting_unit == "grams") {
        if (ending_unit == "mol") {|...|}
        else if (ending_unit == "atoms/molecules") {
            result_ = convertGramsToAmount(rawInput, result.mw);
            result_ = convertToLatexSciNotation(convertMolecToItsAtoms(result_, ratio));
        }
}
```

```
result_ = convertGramsToAmount(rawInput, result.mw);
       result_ = convertMolecToItsAtoms(result_, ratio);
       result_ = convertAmountToMol(result_);
       result_ = convertToLatexSciNotation(convertMolToGrams(result_, mass_second));
else if (starting_unit == "mol") {
     result_ = convertMolToAmount(rawInput);
     result_ = convertMolecToItsAtoms(result_, ratio);
     result_ = convertAmountToMol(result_);
     result_ = convertToLatexSciNotation(convertMolToGrams(result_, mass_second));
else if (starting unit ==
 if (ending_unit == "grams") {...
} else if (ending_unit == "atoms/molecules") {
      result_ = convertMolToAmount(rawInput);
      result_ = convertToLatexSciNotation(convertMolecToItsAtoms(result_, ratio));
else if <mark>(starting_unit == "m</mark>ol<mark>")</mark> {
 if (ending_unit == "grams") { ···
} else if (ending_unit == "atoms/molecules") { ··
      result_ = convertMolToAmount(rawInput);
      result_ = convertMolecToItsAtoms(result_, ratio);
      result_ = convertToLatexSciNotation(convertAmountToMol(result_));
else if <mark>(starting_unit == "atom</mark>s/molecules") {
 if (ending_unit == "grams") {
      result_ = convertMolecToItsAtoms(rawInput, ratio);
      result = convertAmountToMol(result );
      result_ = convertToLatexSciNotation(convertMolToGrams(result_, mass_second));
else if (starting_unit == "atoms/molecules") {
 if (ending_unit == "grams") { ···
} else if (ending_unit == "mol") {
      result_ = convertMolecToItsAtoms(rawInput, ratio);
      result_ = convertToLatexSciNotation(convertAmountToMol(result_));
le if (starting_unit == "atoms/molecules") {
  if (ending_unit == "grams") { ···
  } else if (ending_unit == "mol") { ···
     result_ = convertToLatexSciNotation(convertMolecToItsAtoms(rawInput, ratio));
```

Limitations

- It cannot convert from one molecule to another (yet) [an input that gets either the ratio of two molecules, or the balanced equation is needed]
- It uses DriveToWeb to make the website visible to everyone on the internet. If DriveToWeb slows down or shuts down, the website will also be the same.

Other stuff:

To make inputting numbers, multiplication and exponents, we convert symbols like "*" and "^" into its LaTeX equivalent to make the output pretty. Moreover, since Javascript shows exponents as "e", we convert that to the form 10^(n) for a pretty output.

In "updatePreview", this is what runs whenever the user inputs something or does something in the website. This function tries to get the molar mass of the molecule (or atom) that the user inputs, the value, the starting unit, and the ending unit. Afterwards, it calculates it and outputs it.

```
// main meat showing the given, known, and step-by-step solution
    async function updatePreview() {
        // get the input
        var rawInput = input text.value.trim();
        const latexInput = convertIoLaTeX(rawInput);
        rawInput = convertToLaTeX(rawInput);

        // get element or compound and the units
        const element = element_compound_molecule_type.value.trim();
        const element_2 = element_compound_molecule_type_2.value.trim();

        if (element == "" || element_2 == "") {
            previewText.innerHIML = "";
            return;
        }

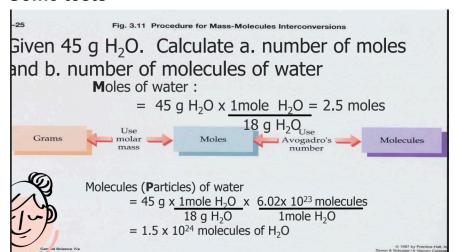
        const starting_unit = start_unit.value;
        const ending_unit = end_unit.value;
        const element_for_CI = element.replace("_", "");
        const element_for_CI_2 = element_2.replace("_", "");

        var result;

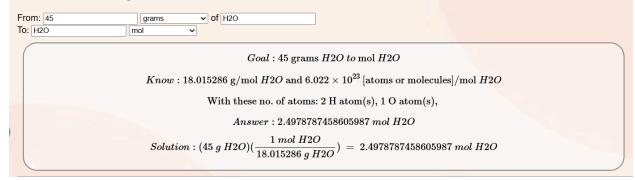
        try {
            result = CI.Chemcalc.analyseMF(element_for_CI);
            element_compound_molecule_type.classList.remove("error");
        } catch (error) {
            element_compound_molecule_type.classList.add("error");
            previewText.innerHIML = "\\[\text{\text{\text{\text{Bad input: Make sure no trailing symbols are p Math]ax.typesetPromise();
            return;
        }
}
```

index.html contains the basic html structure of the website. stoich.css contains basic css of the website wherein everything is organized and made pretty.

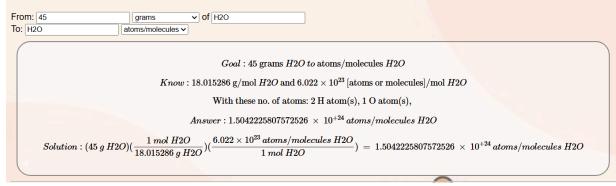
Some tests



Stoichiometry Conversion



Stoichiometry Conversion



[Matches!]

```
Atoms of oxygen and hydrogen # atoms H = 1.5 \times 10^{24} molecules H<sub>2</sub>O x 2H atoms H<sub>2</sub>O molecule = 3.0 \times 10^{24} atoms H # atoms O = 1.5 \times 10^{24} molecules H<sub>2</sub>O x 1 O atom H<sub>2</sub>O molecule = 1.5 \times 10^{24} atoms O
```

Stoichiometry Conversion

Fro To:	m: 45 grams v of H2O H atoms/molecules v
	$Goal: 45 ext{ grams } H2O ext{ to atoms/molecules } H$
	$Know$: 18.015286 g/mol $H2O$ and $6.022 imes 10^{23}$ [atoms or molecules]/mol $H2O$
	With these no. of atoms: 2 H atom(s), 1 O atom(s), $Answer: 3.008445161514505 \ \times \ 10^{+24} \ atoms/molecules \ H$
	$Solution: (45~g~H2O)(\frac{1~mol~H2O}{18.015286~g~H2O})(\frac{6.022\times10^{23}~atoms/molecules~H2O}{1~mol~H2O})(\frac{2~atom(s)~H}{1~molecule~H2O}) = 3.008445161514505\times10^{+24}~atoms/molecules~H2O)$

Stoichiometry Conversion

[Matches!]