How to add formulas and chemical names to the game

Files

There are three files containing the formulas and chemical names: `entities1.lvl`, `entities2.lvl` and `entities3.lvl`, the number corresponding to the difficulty. They can be found in the folder `assets/data/levels`.

Formatting your line

To add new chemical elements to the files, you have to separate each element with the `::` operator. `french_name::english_name_or_symbols::good_answer::wrong_answer1::wrong_answer2::wrong_answer3` Example: `Amidure::Azanide::NH2-::NH::NH3::NH4+`

You don't have to worry about the order of the wrong answers, as it will be randomized with the good answer in the game.

Spaces are allowed.

`Acide chlorhydrique::Hydrochloric acid::HCl::KCl::H2Cl2::HCl2`

Comments start with `//`.

```
// This line will be ignored by the program.
```

Symbols and ids

Instead of writing the English names for each element you add, you can use symbols. The `*` symbol means that the french chemical name will be picked (accents are removed).

To simplify the following examples, I will write `(...)` instead of `good_answer::wrong_answer1::wrong_answer2::wrong_answer3`

Example: `Acétate::*::(...)` => `Acetate`

You can add ids after `*` to not only pick the french chemical name and remove accents, but also add other modifications to it.

- `*i` replaces last occurence of y by i. `Hydroxyde::*i::(...)` => `Hydroxide`
- `*l` removes the last character. `Hydrogène::*l::(...)` => `Hydrogen`

- `*ium` removes the last character and adds ium suffix. `Chrome::*ium::(...) => Chromium`
- `*ide` replaces `ure` suffix by `ide`. `Chlorure::*ide::(...) => Chloride`
- `*rev` interchanges the words and removes French preposition. `Cyanate de sodium::*rev::(...) => Sodium cyanate`
- `*rev` also translates `argent` into `silver`, as many elements of the files have it. `Iodate d'argent::*rev::(...) => Silver iodate`
- `*acid` interchanges the words, changes the suffix (from `ique` to `ic`) and capitalizes the result. `Acide maléique => Maleic acid`

You can call multiples symbols by separating them using the `:` operator. All modifications will be made in order.

```
Peroxyde d'hydrogène::*l:*rev:*i::(...)
```

1. `*l => Peroxyde d'hydrogen`
2. `*rev => Hydrogen peroxyde`
3. `*i => Hydrogen peroxide`

You can also use several times the same symbol.

```
Hydroxyde de baryum::*i:*rev:*i::(...)
```

1. `*i => Hydroxyde de barium`
2. `*rev => Barium hydroxyde`
3. `*i => Barium hydroxide`