Sentence Generator (Rita.js)

Link

- Be sure to share the link to the proper program (with Rita.js already linked) with the teacher
- Have students click on the link to the program
- Make sure that they all have an account so that they can save their work
- Students can rename the program once they have it open

Basic Structure

- Declare two variable `rg` and `button`
- Make the `setup` function with `noCanvas()`
- Make `rg` a `new RiGrammar()`
- Add a rule for `<start>`

  - For now, make the rule `The <N>.`
  - Add a rule that defines `<N>` as three animals

- Create a button with the variable `button`
- Make a callback called `makeSentence'
  - The callback should make a new paragragh with `rg.expand()`

- Be sure to test the code

```
let rg, button;

function setup() {
    noCanvas();
    rg = new RiGrammar();
    rg.addRule('<start>', 'The <N>.');
    rg.addRule('<N>', 'cat | dog | butterfly');

    button = createButton('Generate');
    button.mouseClicked(makeSentence);
}


function makeSentence(){
    createP(rg.expand());
}
```

Adding Verbs

- Add a new rule for `<V>`

    - For now, make these verbs relate to each animal - meows, barks, flutters

- Test the code, notice that the verbs do not appear in the sentence
- Add `<V>` to the `<start>` rule

```
let rg, button;

function setup() {
    noCanvas();
    rg = new RiGrammar();
    rg.addRule('<start>', 'The <N> <V>.');
    rg.addRule('<N>', 'cat | dog | butterfly');
    rg.addRule('<V>', 'meows | barks | flutters');

    button = createButton('Generate');
    button.mouseClicked(makeSentence);
}


function makeSentence(){
    createP(rg.expand());
}
```

Verb Tense

- We can further define a rule, let's make the verbs either present or past tense
- Change the `<V>` rule such that it will select between either `<present>` or `<past>`
- Now add rules for both `<present>` and `<past>`
- We don't need to edit the `<start>` rule because `<present>` and `<past>` are a part of `<V>` which is already in the `<start>` rule

```
let rg, button;

function setup() {
    noCanvas();
    rg = new RiGrammar();
    rg.addRule('<start>', 'The <N> <V>.');
    rg.addRule('<N>', 'cat | dog | butterfly');
    rg.addRule('<V>`, '<present> | <past>');
    rg.addRule('<present>', 'meows | barks | flutters');
    rg.addRule('<past>', 'meowed | barked | fluttered');

    button = createButton('Generate');
    button.mouseClicked(makeSentence);
}


function makeSentence(){
    createP(rg.expand());
}
```

Prepositional Phrases

- Now we are going to add some prepositional phrases to our sentences
- Create a rule for `<Pphrase>`
- Add some phrases to the rule - under the chair, on the table, in the kitchen
- Be sure to add `<Pphrase>` to the `<start>` rule

```
let rg, button;

function setup() {
    noCanvas();
    rg = new RiGrammar();
    rg.addRule('<start>', 'The <N> <V> <Pphrase>.');
    rg.addRule('<N>', 'cat | dog | butterfly');
    rg.addRule('<V>`, '<present> | <past>');
    rg.addRule('<present>', 'meows | barks | flutters');
    rg.addRule('<past>', 'meowed | barked | fluttered');
    rg.addRule('<Pphrase>', 'in the kitchen | under the chair | on the table');

    button = createButton('Generate');
    button.mouseClicked(makeSentence);
}

function makeSentence(){
    createP(rg.expand());
}
```

Make the Program Your Own

- Customize your program so that it produces unique sentences
- Erase the examples and replace them with your own nouns, verbs, and prepositional phrases
- Work with the people in your group
- See if you can add different parts of speech like adjectives or adverbs