



{CUSTOM}

Andrew Seaman, Booker Martin, Ian Green, Veronica Backer-Peral



After the tragic loss of the Custom Hotel, we wanted to dedicate our **customizable** language to the fallen Westchester landmark.

Like the Custom Hotel, we want our language to appeal to everyone in the community and foster an environment where people can share their **unique** version of the language.

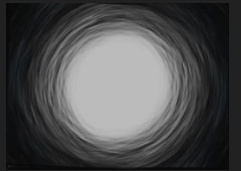
```
{  
  "if": "maybe",  
  "else": "yolo",  
  "for": "fosho",  
  "while": "whileWhileWhile",  
  "in": "within",  
  "break": "gimmeABreak",  
  "print": "gimme",  
  "return": "sendIt",  
  "length": "whatsDaSize",  
  "true": "facts",  
  "false": "nawMan",  
  "string": "letterz",  
  "int": "decimalBegone",  
  "float": "floatYoBoat",  
  "bool": "boolin",  
  "void": "darkness"  
}
```

To make this possible, Custom features **keywords** that you define in a customizable JSON.

Custom is a **statically typed** language with the following types:

`string, int, float, bool, void`

`"Hello", 2, 3.14, true,`



Variable Declaration: `int x = 1`

Function Declaration Comparison

JavaScript	Custom
<pre>function getFirstValue(keyValues) { return keyValues["first"] }</pre>	<pre>int getFirstValue(<string, int> keyValues) { return (keyValues["first"]) }</pre>

Custom requires the **return type** as well as **parameter types**

Operators

		Operator	Example
Precedence	High	> >= == != < <=	return x < y
		&&	return(true && false)
		+ -	int x = 4 + 20
		* / %	float x = 6 % 9
		!	return(!true)
		-	int x = -1
		^	float y = 1^100
Low			

Data Structures

Array:

```
float[] probabilities = [0.1, 0.4, 0.5]
```

Set:

```
string{} cheese = {"brie", "cheddar",  
    "mozzarella", "gouda"}
```

Dictionary:

```
<string, int> playersAndScores = {"Anthony" : 1,  
    "Steve" : -1, "Gerry" : 3}
```

Sets and dictionaries do *not* allow duplicates, but arrays do

Loops

For Loop:

JavaScript

```
for (let i = 0; i < 10; i++) {  
    console.log(i)  
}
```

Custom

```
for (int i = 0; i < 10; i++) {  
    print(i);  
}
```

While Loop:

JavaScript

```
let i = 0;  
while (i < 10) {  
    console.log(i)  
    i++  
}
```

Custom

```
int i = 0;  
while (i < 10) {  
    print(i);  
    i++;  
}
```


The “default” config for our examples so far...

```
{  
  "if": "if",  
  "else": "else",  
  "for": "for",  
  "while": "while",  
  "in": "in",  
  "break": "break",  
  "print": "print",  
  "return": "return",  
  "length": "length",  
  "true": "true",  
  "false": "false",  
  "string": "string",  
  "int": "int",  
  "float": "float",  
  "bool": "bool",  
  "void": "void"  
}  
  
int main (int x, int y) {  
  if(x < y) {  
    print('please work')  
  }  
  else if(x > y) {  
    print('cry')  
  }  
  else {  
    print('cry more')  
  }  
  return 1;  
}
```

Custom-ize!

```
{  
  "if": "maybe",  
  "else": "yolo",  
  "for": "fosh",  
  "while": "whileWhileWhile",  
  "in": "within",  
  "break": "gimmeABreak",  
  "print": "gimme",  
  "return": "sendIt",  
  "length": "whatsDaSize",  
  "true": "facts",  
  "false": "nawMan",  
  "string": "letterz",  
  "int": "decimalBegone",  
  "float": "floatYoBoat",  
  "bool": "boolin",  
  "void": "darkness"  
}
```

```
decimalBegone main (decimalBegone  
    x, decimalBegone y) {  
    maybe(x < y) {  
        gimme('please work')  
    }  
    yolo maybe(x > y) {  
        gimme('cry')  
    }  
    yolo {  
        gimme('cry more')  
    }  
    sendIt 1;  
}
```

RealHotGirlScript

```
{  
  "if": "iHaveSomethingToSay",  
  "else": "BECAUSEIMONFUCKINGVACATION",  
  "for": "openHerUp",  
  "while": "wylin",  
  "in": "thatsSOinRn",  
  "break": "GTFO",  
  "print": "supLilBitch",  
  "return": "andThemsTheFacts",  
  "length": "howMuchLonger",  
  "true": "trueShit",  
  "false": "fraudulentAssBitch",  
  "string": "wordz",  
  "int": "digitz",  
  "float": "longz",  
  "bool": "boolz",  
  "void": "leftOnRead"  
}
```

```
digitz main (digitz x, digitz y) {  
  iHaveSomethingToSay(x < y) {  
    supLilBitch('please work')  
  }  
  BECAUSEIMONFUCKINGVACATION  
    iHaveSomethingToSay(x > y) {  
      supLilBitch('cry')  
    }  
  BECAUSEIMONFUCKINGVACATION {  
    supLilBitch('cry more')  
  }  
  andThemsTheFacts 1;  
}
```

Passing 112 tests with 100% coverage

112 passing (204ms)

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered Line #s
All files	100	100	100	100	
grammar	100	100	100	100	
customGrammar.js	100	100	100	100	
src	100	100	100	100	
analyzer.js	100	100	100	100	
ast.js	100	100	100	100	
generator.js	100	100	100	100	
optimizer.js	100	100	100	100	
parser.js	100	100	100	100	
stdlib.js	100	100	100	100	

One more thing...

Error: "June" is NOT an acceptable configuration keyword

```
at must (file:///C:/Users/booke/Documents/Projects/lmu-cmsi-488/Custom/  
at analyze (file:///C:/Users/booke/Documents/Projects/lmu-cmsi-488/Cust  
at Context.<anonymous> (file:///C:/Users/booke/Documents/Projects/lmu-c  
at processImmediate (node:internal/timers:463:21)
```





Each person's version
of Custom is truly
their own. When you
see "CUSTOM," you
know you are home.

-Us



{CUSTOM}

Andrew Seaman, Booker Martin, Ian Green, Veronica Backer-Peral

Thank you!