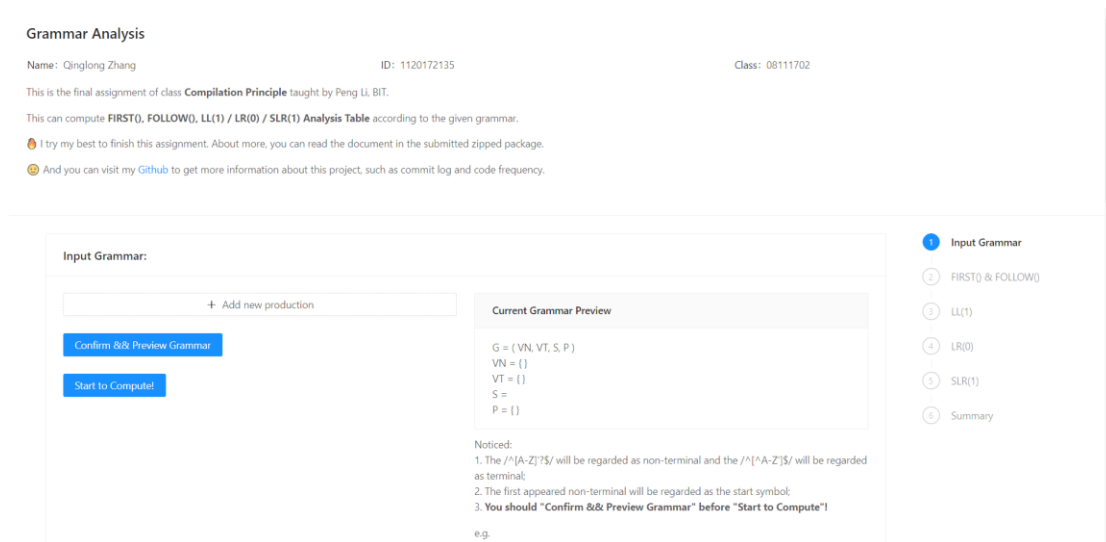


三、使用

终于到了最重要的环节：使用。显然对于一个 GUI 项目来说，功能性和易用性对于用户来说是最重要的。

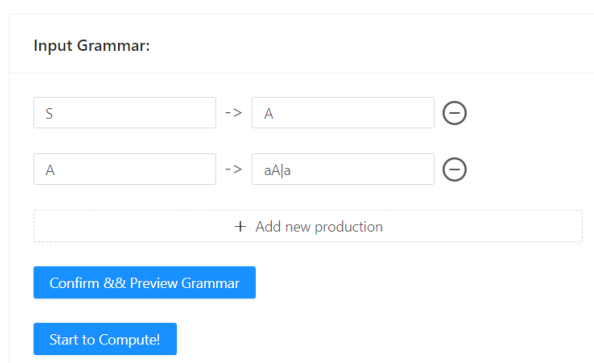
使用第一部分“运行”中的步骤将项目启动后，我们将会在浏览器中看到如下界面：



3.1 输入文法

“Input Grammar”部分为输入文法的区域，我们可以点击“Add new production”按钮来添加新的产生式，文法的非终结符、终结符、开始符号将会被自动推断出来，详细规则可查看界面中的右下文字说明。

我们将使用第 5 章 ppt 中的文法 $\{ S \rightarrow A, A \rightarrow aA|a \}$ 来演示，如下：



输入完成之后，点击下方的“Confirm & Preview Grammar”按钮，系统将在上方进行友好的提示：

✓ You've confirm the grammar, now you can start to compute!

同时用户输入的文法将会在右边以规范的形式显示出来，如下：

Current Grammar Preview

```
G = ( VN, VT, S, P )
VN = { S, A }
VT = { a }
S = S
P = { S -> A, A -> aA|a }
```

用户可以在这里检查系统自动推算出来的非终结符、终结符、开始符号、产生式是否与预期相同。

3.2 计算

完成 3.1 后，点击“Start to Compute!”按钮，计算完成后（实际上就是一瞬间的事）界面上方将出现提示：

i All computations have finished!

代表着以下内容已计算完成：

- (1) 所有非终结符的 FOLLOW 集；
- (2) 所有非终结符与产生式右部的 FIRST 集；
- (3) LL(1) / LR(0) / SLR(1) 分析表；

同时界面右侧将形象的显示计算的结果：



其中红色✖即代表不是此类文法，上图所表达的即此文法不是 LL(1)文法和 LR(0)文法，但是是 SLR(1)文法。

3.3 查看结果

我们向下滚动，可以依次看到 FIRST/FOLLOW 集、LL(1)分析表、LR(0)分析表及 DFA 可视化、SLR(1)分析表及 DFA 可视化：

FIRST() & FOLLOW()

X	FIRST(X)	FOLLOW(X)
S	{ a }	{ # }
A	{ a }	{ # }
aA	{ a }	
a	{ a }	

LL(1) Analysis Table

This grammar is not a LL(1) grammar!

LR(0) Analysis Table

This grammar is not a LR(0) grammar!

DFA Graph

SLR(1) Analysis Table

	ACTION		GOTO
	a	#	A
0	S2		1
1		acc	
2	S2	r2	3
3		r1	

DFA Graph

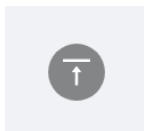
最后是对此文法的一个总结：

Summary

This grammar is a SLR(1) grammar.

You can click the button "↑" in the right bottom to go back to top.

我们可以点击右下角的按钮快速回到顶部：



选择刷新页面或者点击输入框旁的删除按钮删除已输入的产生式，开始另一个文法的计算：

Input Grammar:

<input type="text" value="S"/>	->	<input type="text" value="A"/>	<input type="button" value="⊖"/>
<input type="text" value="A"/>	->	<input type="text" value="aA a"/>	<input type="button" value="⊖"/>