

Syntax Highlighting:



To make coding in **SASM** more comfortable and developer-friendly, we've created a **custom VS Code theme** tailored specifically for the syntax of our language. This theme highlights key constructs, registers, operands, and labels, making it much easier to read and debug code at a glance.

🌟 Install the theme by using our [install script](#).

AST:



Understanding how your code is structured internally is crucial. That's why we built a visual AST (Abstract Syntax Tree) viewer, which displays the parsed structure of your program.

This tool helps visualize how the assembler understands your code, showing:

```
the hierarchical structure of scopes,  
statement ordering within blocks, and  
how constants and expressions are resolved.
```

!!! info "Local and Global Scopes"

```
Every code block in the visualized AST represents a scope.  
- Block 0 is the global scope.  
- Nested blocks represent functions, conditionals, or loops.
```

This makes it significantly easier to debug parsing errors, optimize macros, and understand the flow of control — even for larger programs.



Under Development

- 🔧 A new programming language called **ORIN** is currently under development. It is being designed to compile directly to SASM.

If you're interested in compilers, language design, or virtual machines — **contributions are very welcome!**