

Aurora Language Support

A Visual Studio Code extension providing language support for **Aurora**, an object-oriented scripting language where everything is treated as an object.

✨ Features

◆ Intelligent Autocomplete

- Suggests built-in objects (e.g. `String`, `Int`, `Terminal`)
- Context-aware method suggestions:

```
String.
```

→ `create`, `concat`, etc.

- Variable-aware autocomplete:

```
String.create(name="Liam")  
name.
```

→ `String` instance methods like `substring`, `contains`, etc.

◆ Parameter Suggestions

Autocomplete for method parameters:

```
Terminal.readInt(
```

→

- `message=`
- `min=`
- `max=`

Automatically filters out parameters already used.

◆ Syntax Highlighting

Highlights:

- Built-in types (`String`, `Int`, `Terminal`, etc.)
 - Methods
 - Strings
 - Numbers
 - Named parameters
-

◆ Auto-Closing Pairs

Automatically inserts:

- `()` for method calls
 - `""` for strings
-

◆ Run Button ►

Run Aurora files directly from the editor.

- Appears when editing `.aur` files
- Executes:

```
aurora yourfile.aur
```

- Uses an integrated terminal (no debugger required)
-



Getting Started

1. Install the Aurora CLI

Ensure the `aurora` command is available globally:

```
aurora file.aur
```

2. Create a File

Create a file with the `.aur` extension:

```
String.create(name=Terminal.readLine("What is your name? "; default="n/a"))

Int.create(age=Terminal.readInt("How old are you? "; min=0; max=100))

Terminal.writeLine("You are called"; name; "and you are"; age; "years old")
```

Language Overview

Aurora is designed around a simple principle:

Everything is an object.

There are:

- No traditional keywords (`if`, `for`, etc.)
- All operations are performed via object methods
- Variables are created using:

```
Type.create(name=value)
```

Supported Built-in Objects (partial)

- `String`
- `Int`
- `Float`
- `Boolean`
- `Terminal`
- `Optional`
- `Math`

Known Limitations

- No full type inference (yet)
- Limited scope awareness
- Nested expressions may not always resolve correctly
- Signature help and hover info are not implemented (planned)

Roadmap

- Signature help (parameter hints while typing)
- Hover documentation
- Error diagnostics from Aurora runtime
- Improved type inference
- Full Language Server (LSP) integration

Feedback

This extension is under active development. Suggestions, issues, and improvements are welcome.

