

## AOZ Studio Beta - Bug #14

### Variables named the same as procedures - Compile error

01/17/2020 10:09 PM - Jason Wroe

<b>Status:</b>	Rejected	<b>Start date:</b>	01/17/2020
<b>Priority:</b>	Low	<b>Due date:</b>	
<b>Assignee:</b>	Francois Lionet	<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0:00 hour
<b>Target version:</b>	0.9.3		
<b>Affected version:</b>	1.0.0 (B3)		
<b>Description</b>			
Global HIT			
Procedure HIT			
End Proc			
Badly named but this compiled in AMOS, compile error in "Global HIT"			

#### History

##### #1 - 01/21/2020 04:18 PM - Francois Lionet

- Assignee set to Francois Lionet
- Priority changed from Normal to Low
- Target version set to 0.9.3

##### #2 - 09/21/2020 12:01 AM - Brian Flanagan

- Status changed from New to Rejected
- Estimated time set to 0:00 h
- Affected version changed from 0.9.2.6 to Beta RC2

The problem is, that since a procedure can now have a return value, there is no way for the transpiler to tell what the intention of the programmer is.

Just suppose hypothetically, that it was "possible", and I had the following program:

```
Global HIT
HIT=3

Procedure HIT
  Print "hit"
End Proc[5]

Print HIT
```

Here's where the conflict occurs. Is the result 3 or 5? There is NO way to tell. In order to know the difference, the computer would have to read the mind of the programmer.

The only way I can see around this would be to REQUIRE an empty set of square brackets in the Procedure definition for a procedure without parameters, in order to differentiate it from the variable with the same name. I don't think we want to do that. I'd rather avoid the [] on procedures with no parameters, that to have the dubious option of having a variable and a Procedure with the same name.

##### #3 - 02/18/2021 12:28 PM - Brian Flanagan

- Affected version changed from Beta RC2 to 1.0.0 (B3)