

SimpleRule

```
syntax SimpleRule = RuleAttributes "syntax" Name "=" RuleExpression SimpleRuleBody;
```

`SimpleRule` is a named rule that sorts out context-free grammar. It supports direct and indirect right recursion. Left recursion is not supported. The (*and+) loops are supported as well.

Each `SimpleRule` declaration also describes an AST node (class). An AST class may contain one or more fields which are calculated according to the grammar (defined by `RuleExpression`). A field is created for each sub-rule in `RuleExpression`. Field names are created automatically or by an author of the grammar. See the "AST fields" section for details.

Example:

```
syntax Property = Key ":" Value;
```

This rule describes JSON property. `Key` and `Value` refer to the other rules that should be either declared in the same syntax module or imported from other modules.

`SimpleRuleBody` may contain:

- declarations of other nested rules
- rule aliases
- methods

See "`SimpleRuleBodyMember`" for details.

See also

[RuleAttributes](#)
[Name](#)
[RuleExpression](#)
[SimpleRuleBody](#)