Syntax

Syntax is the form (structure or grammar) of a series (sequence or string) of items. It gives a way to know if something has the proper (valid or legal) form.

Full Name of a person can be defined by the following syntax:

> first, an optional title (or salutation), then a first name, optional middle name or initial followed by a period, then a last name and finally any number of endings.

Examples of proper FullNames:
- **Jane Doe**
- **Mr. John Q. Doe**
- **Ms. Jane May Doe**
- **Prof. Bill Q. Gates III**
- **Mary J. MacHinery II MD. Ph.D.**

Examples of improper FullNames:
- **Joe**
- **Mr. Doe**
- **Mrs. J. Doe**
- **Doe, Joe**
- **J. Doe III**

Meta-Language is the language used to describe a language. It includes words like Verb, Noun, CapitalLetter, FirstName, Initial, Salutation, which will often have the first letter capitalized here to avoid confusion,

Definitions of syntax can be done in many ways: by syntax boxes, syntax diagrams (rail diagrams), BNF (Backus-Naur Form), or verbally as done above.

Syntax Boxes are a way of defining the syntax by various kinds of boxes.

Solid boxes indicate the parts that are required (such as Name, first and last).

Dashed boxes indicating optional parts (0 to 1 parts, such as a Title or MidPart).

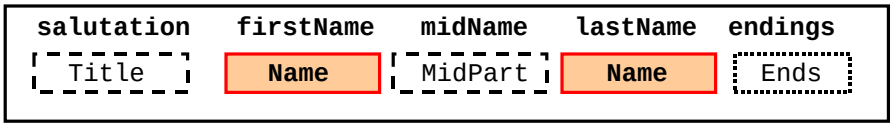
Dotted boxes indicating many occurrences (0 to n parts, such as List of endings).

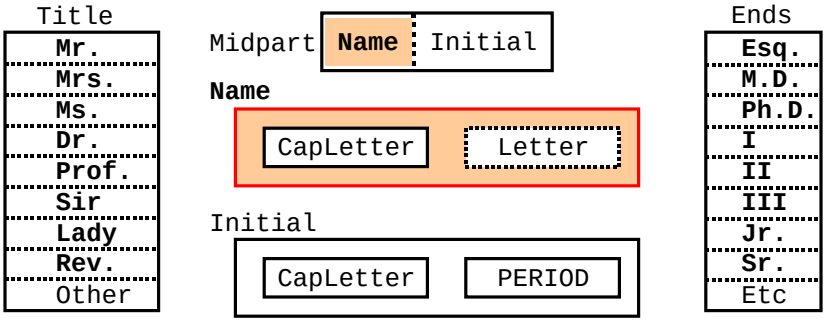
Bold text is to be written exactly as-is.

NonBold text is defined elsewhere in some box.

Words such as firstName and lastName describe the content of the syntactic box Name; they are part of the semantics (or meaning) rather than the syntax.

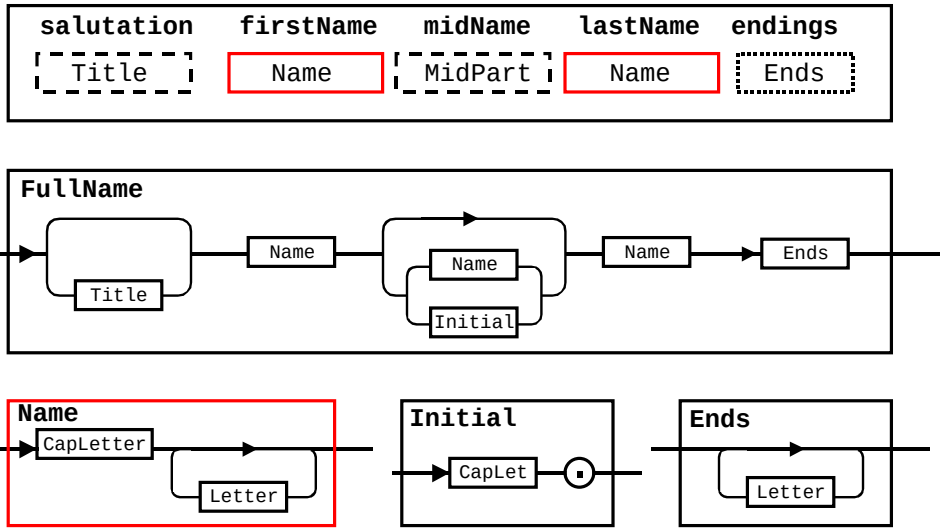
Full Name Syntax





optional

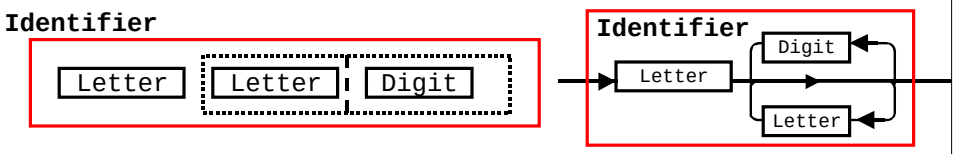
Syntax diagrams (or rail charts) can also describe syntax; they show all possible flows that are proper. The previous syntax boxes for the FullName are repeated and then followed by syntax diagrams. All flows of the parts (words) are from the left to the right unless drawn otherwise.



Names in programming are often called Identifiers. They describe objects, actions, methods, classes, etc). Identifiers differ from names of people; they have fewer parts (no first and last name), no spaces, and may include digits as in the examples:

sum, next, maximum, over21, majorityOf3, midOf3, etc

Identifiers begin with a letter, and may be followed by any number of letters or digits. this is described by the following Syntax Boxes or Syntax Diagram.



Nouns describe boxes, objects, classes, for example:
age, length, maxSize

Verbs describe actions, methods, routines, for example
maximize, sort, increment

Phrases describe logical or boolean expressions, conditions such as
isTall, increasing, over21.

optional

BNF notation (Baccus Naur Form or Baccus Normal Form) is yet another way to show syntax in a rather algebraic form, such as:

```
FullName is:
  Title? <firstName> [Name | (CapLetter '.')]? <LastName> Ends*
```

Words within angle braces, such as <FirstName> are names or categories of names.

Items followed by a question mark (?), such as Title? mean zero-or-one.

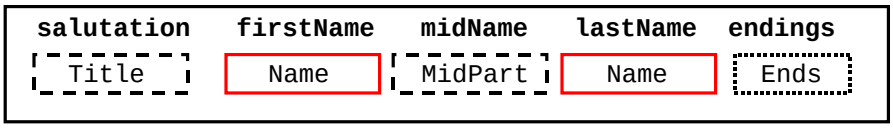
Items separated by bars (|) mean one or the other.

Items followed by an asterisk (*) mean zero-or-more of the item.

Items followed by a plus (+) mean one-or-more of the item.

Strings within a single quote (') are literals written exactly as shown.

The syntax box form is shown again for comparison.

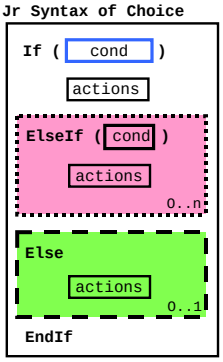


Choice form Syntax is slightly more complex, but does look simpler when shown as Syntax Boxes. It consists of an If-part, followed by any number of Else-parts (including none), followed by either one or no Else-parts. This Choice syntax is shown for pseudoCode Jr and Java; other languages are similar.

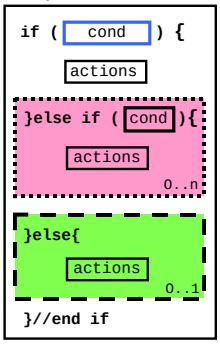
Boxes having many dots indicate that many (0 to n) actions may be included, depending on the given condition.
Boxes having few dashes indicate that some (0 or 1) actions may be included if none of the previous conditions have been satisfied.

PayFactor, at the far right, is an example of code having proper Java syntax.

Jr Syntax of Choice



Java Syntax of Choice



Example: pay factor

```
if (hours < 40.0) {
    rateFactor = 1.0;
}else if (hours < 60.0)  {
    rateFactor = 1.5;
}else if (hours < 90.0)  {
    rateFactor = 2.0;
}else if (hours < 168.0) {
    rateFactor = 3.0;
}else{ // hours > 7*24
    rateFactor = 0.0;
    // error message;
}//end if
```