

# BioBIKE Language Syntax

## A. Introduction to BioBIKE Language Syntax

### A.2. Basic conventions of BioBIKE

#### A.2.a. BioBIKE Syntax - Overview

Syntax in BioBIKE is much simpler and more regular than in English, but we can still recognize the similarity. Consider the following statement in BioBIKE:



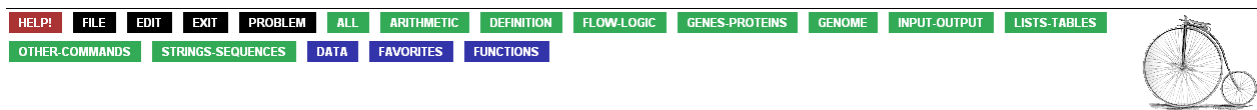
- BioBIKE, like English, has structure.
- The basic units of the language are *functions* (similar to English sentences), represented as *yellow boxes with thick solid boundaries*.
- The structure of the sentence is largely determined by its verb (also the name of the function), which in BioBIKE always resides at the beginning (**SEQUENCE-OF** in the above example).\*
- The structure consists of *holes* to be filled, *gray boxes* (like objects in English). When the information is required, the hole resides directly in the yellow box. These holes are called *arguments to the function*.
- Modifications to how the function works are represented as blue boxes if the modification is either done or not done (e.g. **INVERT** in the above example). These are called *flags*.
- Modifications to how the function works requiring extra information are represented as blue boxes with gray holes. These are called *keywords*. Collectively, flags and keywords are known as *options*.
- Holes can be filled with functions that provide the required content, leading to functions nested within other functions.

That's about all there is to know about the syntax of most BioBIKE functions.

#### A.2.b. How to Find and Invoke BioBIKE Functions

Your English vocabulary is no doubt several thousand words. BioBIKE has a much smaller vocabulary, and therefore most statements in English that sound logical to you will have no meaning in BioBIKE. To form meaningful functions, it is necessary to use precisely the language that BioBIKE knows about and nothing else. To this end, most words in BioBIKE are chosen from a menu. Menus are organized by subject and by alphabet. The first type of menu facilitates finding words when you have an idea of the meaning you're seeking. The second type is convenient when you know the word itself.

For example, go into BioBIKE<sup>†</sup> and note the *palette* at the top of the screen:



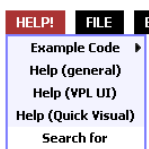
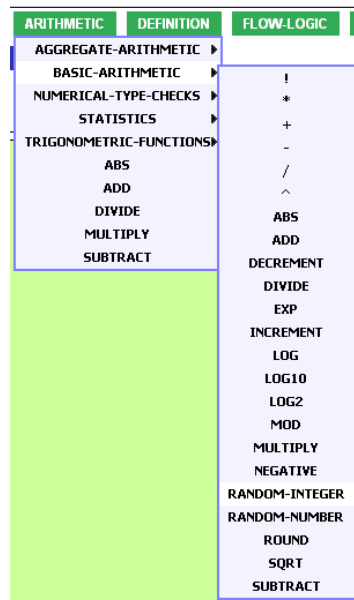
\* For the sake of readability, function names are rendered as nouns, since functions can replace objects.

(you may not see the **DATA**, **FAVORITES**, and **FUNCTIONS** buttons on your screen). Mousing over any of the buttons brings down a menu of choices. In some cases mousing over a choice brings up a submenu. Otherwise, you must click on the choice to select it.

Try mousing over the green **ARITHMETIC** menu, and then mouse over **BASIC-ARITHMETIC** (see figure to right). Finally, click on the function called **RANDOM-INTEGER**. A yellow function box should appear in your green workspace.

Now find the same function from the alphabetical menu on the palette called **ALL**. You'll need to mouse over that button and then mouse over the **R** button. Clicking **RANDOM-INTEGER** brings down the same function as before.

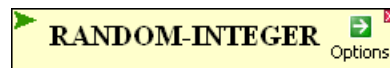
A third way to get the function is from your blue **FAVORITES** button on the palette. As you use BioBIKE, the **FAVORITES** button automatically becomes populated with the functions that you use the most. You can add or subtract functions from this button yourself (by **ADDING**, **DELETING**, or **CLEARING** favorites through the black **EDIT** button).




You can also *search* for a desired function through the search facility accessible through the red **HELP!** key (upper left in the palette). Entering a key word may bring you to the documentation of a function that uses that word or something similar to it.


### A.2.c. How to Work with a BioBIKE Function

The **RANDOM-INTEGER** box you brought down in the previous section has the three icons typical of function boxes in BioBIKE:




**Action icon** : Mousing over this icon brings up a menu of actions related to the function, including executing the function and function-specific help.

Click **EXECUTE** on the Action menu of **RANDOM-INTEGER**, executing the box and everything within it. The result will appear in the blue **Result Window**.

**Clear/Delete icon** : Clicking this icon will delete the function from the workspace.

Try it! (To get the function back, click the **UnDo** icon () at the top of the workspace).

**Option icon** : The way a function works can be modified by the addition of options chosen from a menu accessible through the Option icon.

Mouse over the Option icon of **RANDOM-INTEGER** and click the **FROM** option, then do the same thing but this time clicking the **TO** option. By using these options you can set the range of numbers from which **RANDOM-INTEGER** chooses.

We'll see in the next section how to fill in the holes governed by these options.

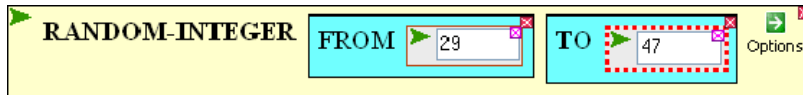
<sup>†</sup> You'll have a much better time if you actually use BioBIKE as you read this documentation.

#### A.2.d. Filling in Holes of Arguments and Keywords

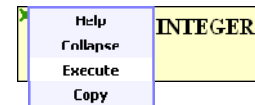
Confronted by the function produced as described in the previous section, how do you provide the two values demanded



by the FROM and TO options? To fill a hole, select it by clicking on the descriptor (*value* in the above example). The box will turn white, indicating that it is open for input. Do this for both holes, typing your favorite numbers into each. You will get something like this:



Now execute the function, by mousing over the action icon of the function and clicking **Execute**, as shown to the right, and... an error message!



You are trying to evaluate code that is not completed!...  
The most likely cause is that an entry box has not been closed.

Indeed! The entry boxes are still open. You know this because they're white. This is probably the most common error in BioBIKE. So, take heed:

***You cannot execute a function that contains an open, white box!***

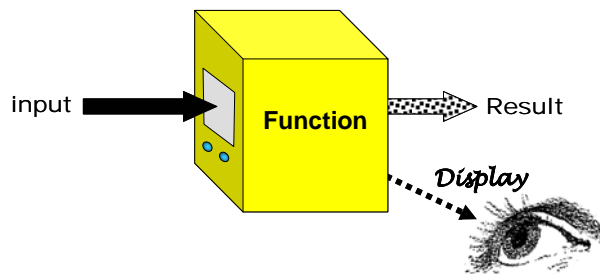
Close an entry box by tapping either **Tab** or **Enter** on your keyboard after typing in the box. Get into the habit of doing one or the other every time that you finish using the keyboard in BioBIKE. **Tab** differs from **Enter** in that it moves the cursor to the next hole and opens it for input.

Having made the mistake of going away from the open boxes, return to one of them by clicking in the white area and then closing it (**Tab** or **Enter**). Do it and execute... Now a random integer between 29 and 47 will very likely appear in the violet Result Window.

Another way to fill a hole is to select it and fill it with a function. You may also fill a hole by pasting into the box, using the **Paste** function in the Action menu. More on that in a later section.

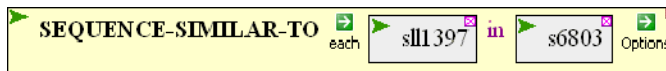
#### A.2.e. Execution of functions: Results and Displays

A function may be thought of as a black box (or, in BioBIKE, a more pleasing citrus-colored box). It accepts input from a human (or a previous function) through the gray argument boxes and emits a result. There's some room for dial twiddling, using the blue-green Options, but most of the magic takes place beyond reach inside the box.



All functions emit results, which can be used as input by subsequent functions. The results either are passed immediately to an enveloping function or, if there is no such function, appears in the **Result Window**. Results are therefore designed to be intelligible by functions and not always intelligible by humans.

Some functions also *display* information in a separate window. The function to the right provides an example. It displays a formatted table, part of which appears to the right. The same information is also contained in a table, the *result* of the function, represented in the **Results Window** as shown.



Display:

QUERY	Q-START	Q-END	TARGET	T-START	T-END	E-VALUE	%ID
1. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	53108	52260	0.0	100.0
2. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	3513163	3512315	0.0	100.0
3. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	2443999	2444847	0.0	99.88
4. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	2534106	2534954	0.0	99.88
5. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	3097436	3098284	0.0	99.88
6. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	124348	123501	0.0	99.88
7. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	1463455	1464302	0.0	99.88
8. <a href="#">S6803.sll1397</a>	1	849	<a href="#">S6803.chromosome</a>	1554776	1553929	0.0	99.88

Result:

```
1> <Table 2d (Numeric,Hash)>
```

The function looks within the genome of the cyanobacterium *Synechocystis* PCC 6803 for sequences similar to that of the gene *sll1397*. The results are displayed in formatted columns and also returned as a result, i.e. a table that can be used in subsequent computations.

It is essential to understand the difference between the *result* of a function and what the function *displays*. The result may be pasted into an argument box (or passed in by nesting) and used by another function. You can go through as long a sequence of Function --> Result --> Function --> Result ... as you like. In contrast, the display is solely for the amusement of the human. If you expect the display of a function to be used by a subsequent function, you will be disappointed.

A very common error is shown to the right. Executing **LENGTH-OF** produces a rather nice display of the genome sequence of the cyanobacterium *Prochlorococcus marinus* SS120. The result is 0! How can the length of the genome, which is displayed for anyone to see, be zero? This paradox disappears once you realize that **DISPLAY-SEQUENCE-OF** only displays the sequence. It does not return the sequence as a result. But since every function must have some result, it returns nothing, NIL. And the length of nothing is zero.

