

# Visual programming blocks

## – Variables

We use the term *variable* in the same way as it is used in mathematics and other programming languages: a named value that can be changed (varies). Variables can be created in several different ways.

- Every [count with](#) and [for each](#) block uses a variable and defines its values. These values can only be used in these blocks. A traditional computer science term for these are [loop variables](#).
- User-defined functions (also known as "procedures") can define inputs, which creates variables that can be used only within the function. These are traditionally called "[parameters](#)" or "arguments".
- Users may create variables at any time through the "set" block. These are traditionally called "[global variables](#)". Blockly does not support [local variables](#).

## Default names

While users can choose any name for a variable, core Blockly provides a default name, "item", as shown in the below picture. Some applications provide other default values, such as "value", also shown below.

## Drop-down menu

Clicking on a variable's drop-down symbol (triangle) gives the following menu:



The menu provides the following options.

- the names of all variables defined in the program.
- "Rename variable...", which changes the name of this variable wherever it appears in the program. Selecting this opens a small window that prompts the user for the new name with the text: "Rename all %1 variables to:", where %1 is replaced by the old name (here "item").
- "New variable...", which allows the user to enter a new name for the variable without replacing or changing variables with the old name (here "item"). Selecting this opens a small window that prompts the user for the new name with the text "New variable name:".

# Blocks

## Set

The **set** block assigns a value to a variable, creating the variable if it doesn't already exist. For example, this sets the value of the variable named "age" to 12.



## Get

The **get** block provides the value stored in a variable, without changing it.



It is possible, but a bad idea, to write a program in which a **get** appears without a corresponding **set**.

# Example

Consider the following example of code:



The first row of blocks creates a variable named "age" and sets its initial value to the number 12. The second row of blocks gets the value 12, adds 1 to it, then stores the sum (13) into the variable. The final row displays the message: "Happy birthday! You are now 13"

---

Revision #2

vytvořené 3 roky nazpět uživatelem [Admin](#)

aktualizováno 3 roky nazpět uživatelem [Jiri Krulis](#)