

Creating a program (Tutorial E4)

http://www.atomiclearning.com/k12/en/movie/35638/play_window?type=Tutorial&sid=1670

[00:00:00.00]

You can write your own functions and programs using the Program Editor in the Calculator application. I'll insert the Calculator

[00:00:08.00]

application by pressing the Home key, and choosing Calculator.

The TI-Nspire™ handheld considers Functions and Programs very similar

[00:00:18.00]

things. One difference is that Functions can return a value that can be graphed or entered in a table, just like $\sin(x)$. In this

[00:00:27.00]

case, I want to write a simple set of instructions that will randomly generate a number, and then display the word "heads" or "tails"

[00:00:36.00]

depending on which number was generated. I'll write this as a program. I'll press the Menu key, choose Functions & Programs,

[00:00:47.00]

and then Program Editor, New. I'll call this program coins by typing "c-o-i-n-s", and I'll make sure that the Type is set to

[00:01:03.00]

Program. I'll leave the Library Access set to the default, and then press Enter to move into the Program Editor. I'll enter into

[00:01:14.00]

a split screen view, with my calculator entry on the left, and the Program Editor on the right. As you can see, the program framework

[00:01:22.00]

has already been entered. I'll press the down arrow key once to move down to the field below the Program statement. I'll be using

[00:01:33.00]

the variable x in my program, but I need to be careful whenever I do that, because other programs and functions might be sharing

TI-Nspire™ Handheld Script

[00:01:40.00]

a variable named “x”. This is called a global variable. I don’t want that type of variable; I want to tell the handheld to make

[00:01:48.00]

a version of the variable x that only applies within my program. This is called a local variable, and I can create that by pressing

[00:01:56.00]

the Menu key, then choosing Define Variables, Local. Now, I’ll type my variable name, “x”. To start a new line, I’ll the press

[00:02:12.00]

the carriage return key in the lower right corner of the keypad. Now, I need to tell the program to generate my random number and

[00:02:20.00]

store it in my local x variable. I’ll open the Catalog by pressing the Catalog key, and then I’ll press “R” on the keypad to move

[00:02:30.00]

to the r’s. Now I’ll scroll down using the NavPad until I highlight the random Integer function, randInt(). Notice that in the gray

[00:02:40.00]

bar on the bottom of the catalog there’s a hint that shows me what this function requires. According to the hint, I’ll need

[00:02:47.00]

to supply both the lower and upper bounds, separated by a comma. I’ll press Enter to copy randInt() to the program line, and then

[00:02:57.00]

I’ll enter my bounds. I only want it to generate a 1 or a 2. 1 will represent heads, and 2 will represent tails. I’ll type “1”

[00:03:07.00]

then a comma, and then “2” so randInt can only return a 1 or a 2. Those are the only two possibilities. I’ll press the right-

[00:03:16.00]

arrow key to move the cursor past the close parenthesis. Once a number is generated, I’ll need to store it in my local x variable

TI-Nspire™ Handheld Script

[00:03:26.00]

so I can test it. I'll press the Ctrl key and then the Var key to select the right-arrow symbol. This symbol allows me to store

[00:03:35.00]

an expression or value in a variable. Now I'll type "x". When I run the program later, the line "randInt(1,2) -> x" will generate

[00:03:47.00]

a random value of 1 or 2 and store it in my local variable, X. Now I'll press the carriage return to start a new line. To test

[00:03:58.00]

the stored x value, I'll need a programming statement called "If...Then...Else". This is called a conditional statement, because

[00:04:07.00]

it executes commands based on conditions. In this case, we want to tell the handheld "If x=1, then display the word 'heads.' Otherwise,

[00:04:16.00]

display the word 'tails.'" Another template helps you enter this statement - just press the Menu key, choose Control, and then

[00:04:26.00]

"If...Then...Else...EndIf". After the template is inserted, I'll leave the cursor right where it's sitting, just to the right of

[00:04:36.00]

the word "If," and I'll type "x=1". Then I'll press the down arrow key to start a new line, and I'll press the Menu key and choose

[00:04:46.00]

I/O>Disp. Now I'll type: "heads". This means that if x=1, the handheld will display the word "Heads". Now I'll press the down-

[00:05:05.00]

arrow key twice to move to the blank line under "Else," and I'll press the Menu key and choose I/O, Disp. Now I'll type: "tails"

[00:05:22.00]

Since x can only be 1 or 2, I know that if it's not 1, it must be 2, so if x doesn't equal 1, then I want my program to display

TI-Nspire™ Software Script

[00:05:33.00]

the word “tails.” Now, I’ll press Menu, Check Syntax & Store, and then choose the Check Syntax & Store option. If I’ve entered

[00:05:46.00]

the program correctly, I’ll see a dialog box letting me know the syntax is correct and that it’s been stored. If there’s a problem

[00:05:54.00]

with syntax, the handheld will display an Error dialog box instead. I’ll choose OK by pressing Enter. To run the program, I’ll press

[00:06:04.00]

Ctrl-Tab to move to the Calculator entry line, then press Var, highlight my “coins” program, and then press Enter. Now, I’ll

[00:06:18.00]

enter a set of empty parentheses, and then press Enter again. I can run the program repeatedly by just continuing to press Enter.

[00:06:27.00]

Notice that it randomly generates either the word “heads” or “tails” and displays one of those words every time I press Enter.

[00:06:38.00]