

CHAPTER 6

Procedures and Functions

starting out with >>> **APP INVENTOR**
FOR ANDROID



TONY GADDIS · REBECCA HALSEY

Topics

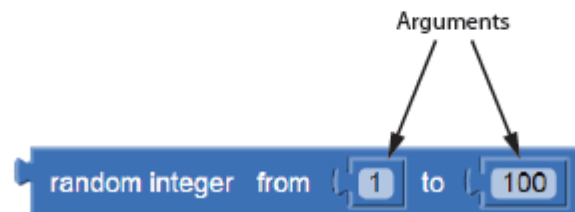
- Modularizing Your Code with Procedures
- Passing Arguments to Procedures
- Returning Values from Procedures

Passing Arguments to Procedures


- Sometimes it is useful to send one or more pieces of data to the procedure. These pieces are known as *arguments*.
- Figure 6-18 shows two arguments. The arguments specify the minimum and maximum values for a random integer.

Figure 6-18 Arguments Passed to the random integer Function

(Source: MIT App Inventor 2)



Passing Arguments to Procedures

- To equip a procedure block with a parameter value, open the procedures mutator bubble .
- Click and drag the input block from the left side of the bubble and insert it into the right side of the bubble.
- Parameters are variables.
- Variables should have meaningful names.

Passing Arguments to Procedures

Figure 6-19 Adding a Parameter to a Procedure (Source: MIT App Inventor 2)

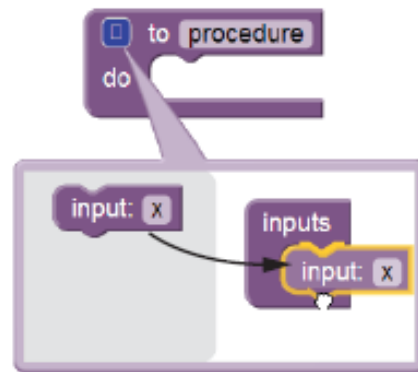
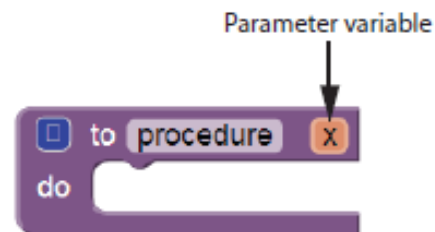


Figure 6-20 A Procedure with a Parameter Named x (Source: MIT App Inventor 2)



Passing Arguments to Procedures

- A parameter's scope is limited to the procedure that it belongs to. To get the value you use a `get` block.
- Use the *Variables* drawer of the Blocks column to create a `get` block as shown in figure 6-22.

Figure 6-22 Selecting the `get` Block from the *Variables* Drawer (Source: MIT App Inventor 2)



Passing Arguments to Procedures

- When a procedure has a parameter, you must pass an argument to that parameter.
- For example, if an app has the `DisplayValue` procedure as shown in Figure 6-24, the procedure's `call` block will have a socket named `ValueToDisplay` as shown in Figure 6-25.
- When you call the procedure, you must plug an argument into the socket.

Passing Arguments to Procedures

Figure 6-24 Getting the Value of a Parameter (Source: MIT App Inventor 2)

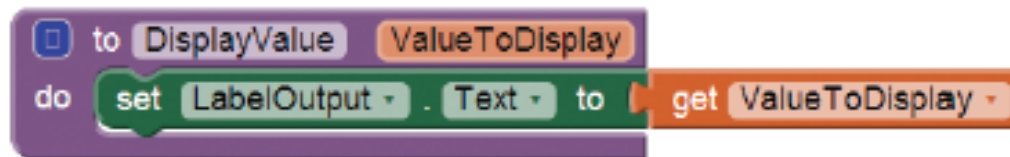
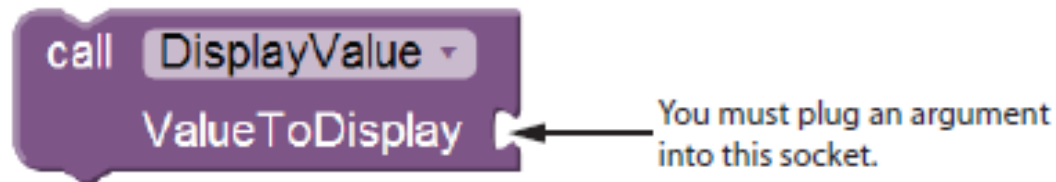


Figure 6-25 Socket for an Argument



Passing Arguments to Procedures

- The app's screen has three buttons, a `Click` event handler has been written for each.
- The workspace has the `DisplayValue` procedure.
- If the user clicks the *Display 5* button, the `ButtonDisplay5.Click` event handler executes.
- This causes the value 5 to appear in the text box as shown in Figure 6-29.

Passing Arguments to Procedures

Figure 6-27 The ArgumentDemo App in the Designer (Source: MIT App Inventor 2)

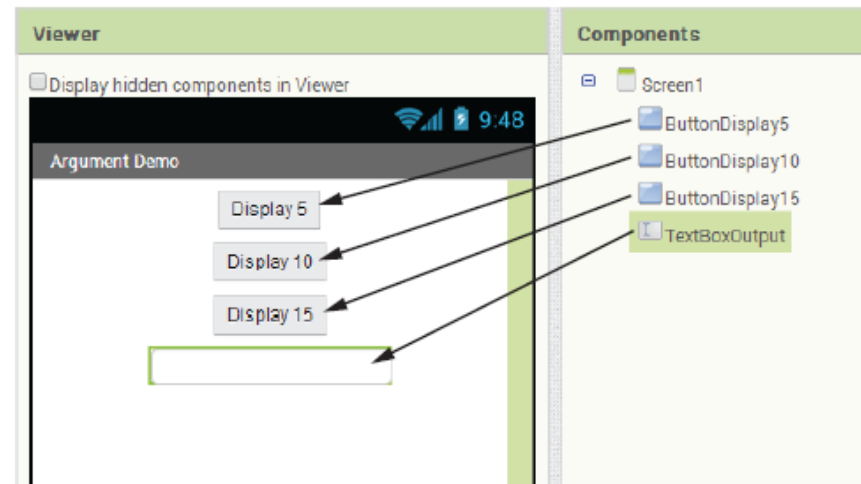
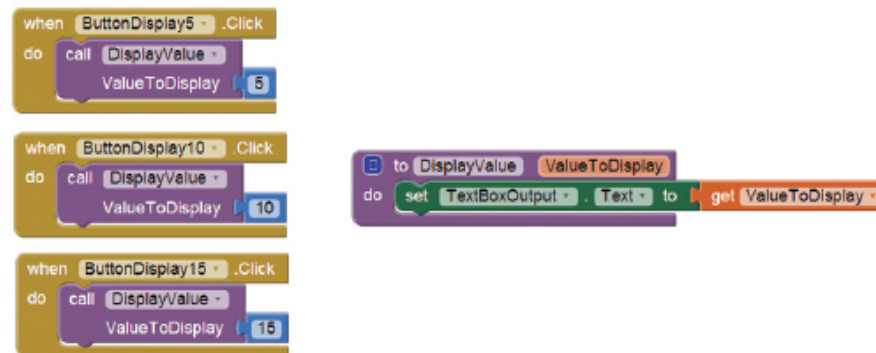


Figure 6-28 The ArgumentDemo App Workspace (Source: MIT App Inventor 2)



Passing Arguments to Procedures

Figure 6-29 Passing the Number 5 as an Argument (Source: MIT App Inventor 2)

