## VBugs

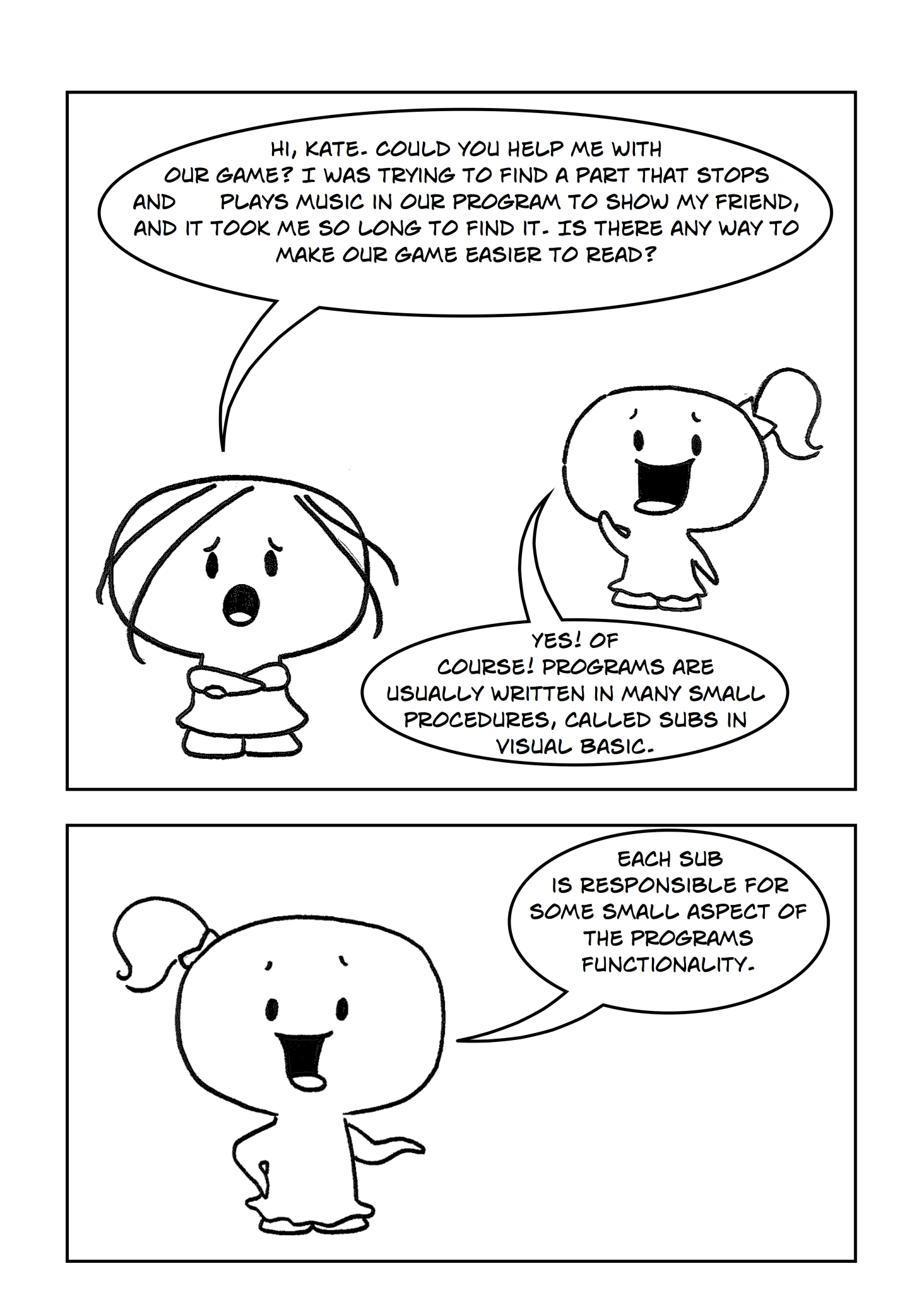
## Chapter 6

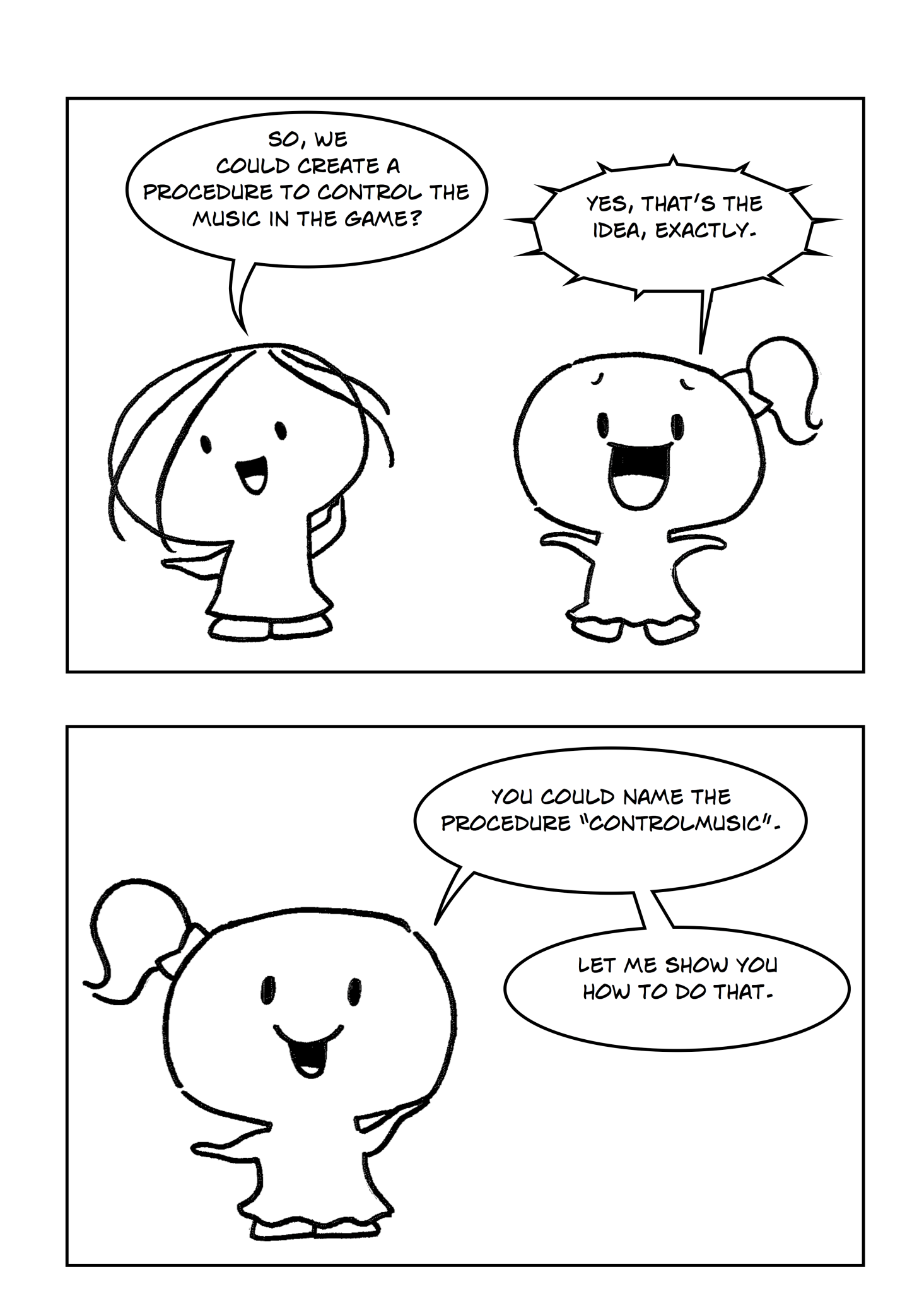
###### Methods in VB.NET



# Summary:

I this chapter you will continue work on your Bugs program from last chapter. You will learn about methods in VB, their purpose and how to create and use them in your program. You will not need any additional material for this chapter.

****

****

# Part 1

### Methods, Subs, Functions and Parameters

A Method is a procedure built into the program. They are a series of statements that are executed when called from somewhere else in your program. Methods allow us to handle code in a simple and organized fashion, this is important as our program becomes more complex. There are two types of methods in VB .NET: those that return a value like a number (Functions) and those that just perform an action (Sub Procedures).

Sub procedures (or just Subs) are methods which do not return a value. Each time when the Sub procedure is called the statements within it are executed until the matching End Sub is reached. Sub Main(), the starting point of the program itself is a sub procedure. When the [application](http://www.startvbdotnet.com/language/methods.aspx) starts execution, control is transferred to the “Sub Main” procedure automatically Figure 1 presents an example of Sub Main() Declaration:

|  |
| --- |
| Public Sub Main()  'this method is called by Default  'statements to execute  ...  End Sub |

Figure 1

A function is a method which returns a value. Functions are used to evaluate data, make calculations or to transform data. Declaring a Function is similar to declaring a Sub procedure. Functions are declared with the Function keyword. Figure 2 shows the example of function declaration.

|  |
| --- |
| Public Function CalculateAvarage() As Integer  'needs to be called somwhere in yur program  'returns an integer  'statements to execute  ...  End Function |

Figure 2

Each method can take some parameters to operate with. A parameter is an argument that is passed to the method by the method that calls it. Parameters are enclosed in parentheses (brackets) after the method name in the method declaration.

An example of this would be a program that calculates the price of an item. Let’s say store is having a sale and different types of items might attract different discounts. You might have a **method** that calculates the discount which is called from the **method** that calculates the price. The **Sub procedure** of calculate price would call the **Function** of discount and pass on to it the **parameters** of type of item it is and the price so it could return total discount on the item.

*cha 2 - worksheet.pngQuestion 1: Define the terms: Method; Sub Procedure; Function; Parameter.*

A method is not executed until it is called from somewhere else. A method is called by referencing its name along with any required parameters. Figure 3 shows a method call with required parameters (a = CalculateAvarage(a, b)) and a method call that does not require parameters (HelloWorld()):

|  |
| --- |
| Public Sub Main()  'calls the sub that prints “Hello World!”  HelloWorld()  'declaring variables to use  Dim a, b As Integer  a = 5  b = 2  'using a funcion and passing variables into it  a = CalculateAvarage(a, b)  text.Drawtext(c)  End Sub |

Figure 3

Variables declared within methods are called “method variables”. They have “method scope” which means that once the method has finished executing they are destroyed and their memory is reclaimed. For example, in Figure 4 a and b variables are declared inside the CalculateAvarage() Sub. Those two variables are accessible only within the method and not from outside the method.

|  |
| --- |
| Public Sub CalculateAvarage()  'local variables a and b  Dim a, b As Integer  a = 5  b = 2  c = a/b  'statements to execute  ...  End Sub |

Figure 4

*cha 2 - worksheet.pngQuestion 2*

1. *Which of the above Subs (Figure 3 or Figure 4) would execute first? Explain your answer.*
2. *Could you access the variable “c” from the Main() procedure to draw it to the screen? Explain your answer.*

### Control Music Method

In our case we want to create a public ControlMusic() sub right above the Public Sub Main(). We then need to cut the code that is responsible for stopping and starting music from the Game Loop, and paste it inside the ControlMusic() Sub.

*Exercise 1: Creating ControlMusic() Sub Procedure*

cha 2 - worksheet.pngMake the following changes in your program and write your solutions to the worksheet:

1. Create ControlMusic() Sub, which will stop and start the music during the game. Call this method inside the Game Loop.

When we start breaking up things into Sub Procedures it makes the code easier to read as well as easier to find things. Now we can use the following navigation feature to move between the methods(Figure 5):

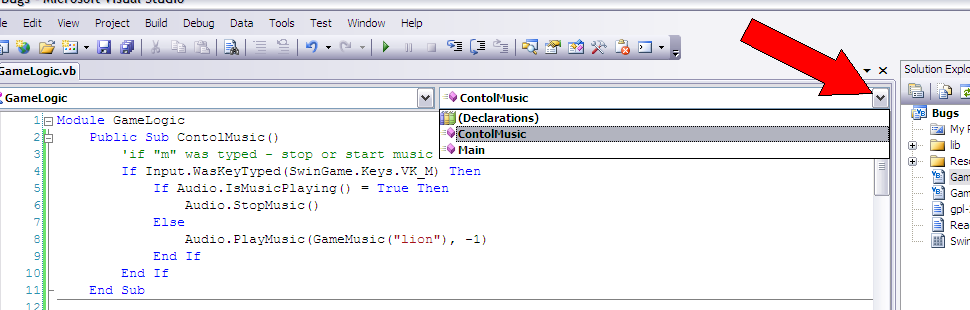
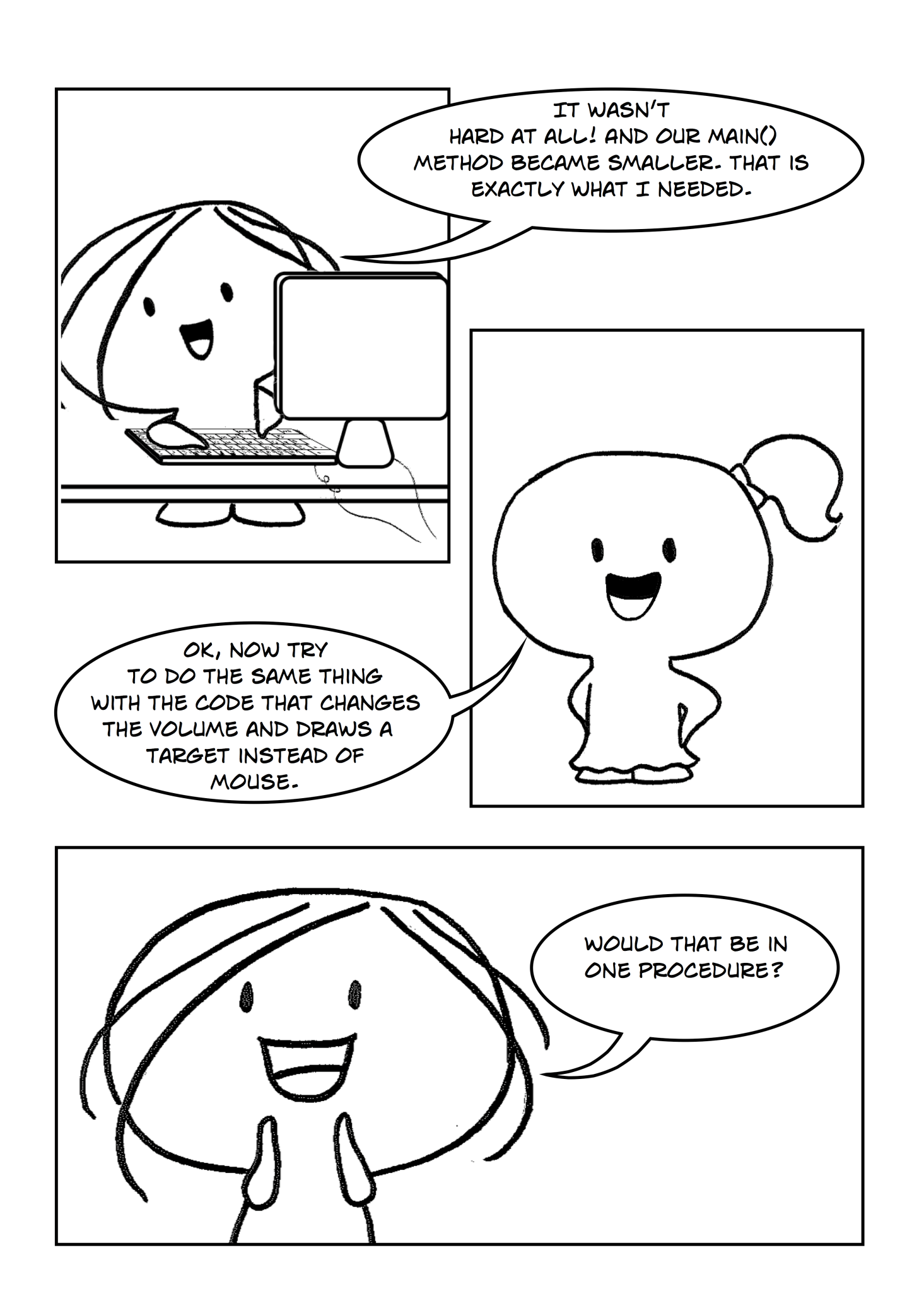
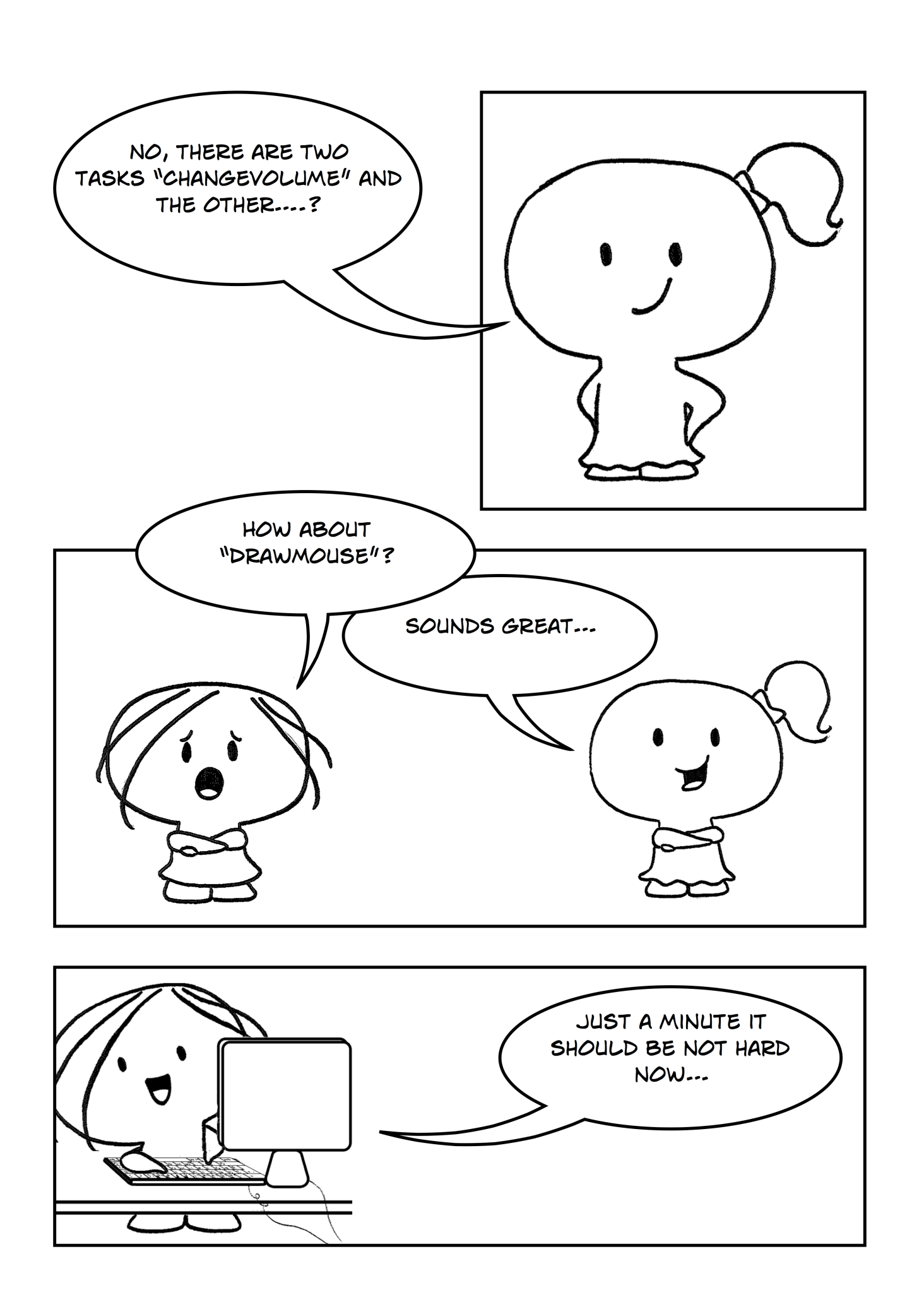


Figure 5

****

****

## Part 2

### Draw Mouse and Mouse Point Methods

Now you have learned how to create subs, you can continue to make our program much more readable and structured by creating two new subs – ChangeVolume() which will be responsible for changing the volume up and down and DrawMouse() which will draw the target image instead of default mouse pointer.

To move code that is responsible for changing the volume, proceed the same steps as in part 1 for ControlMusic() sub. As for DrawMouse() sub, you will need to add a mousePoint variable as Point2D type.

*Exercise 1: Making additional changes*

cha 2 - worksheet.pngMake the following changes in your program and write your solutions to the worksheet:

1. Create a new Sub Procedure called ChangeVolume() which will raise and lower the volume of the music (cut the code from Main sub). Put this sub before the Main() method and call it inside the Game Loop.
2. Create a new Sub Procedure called DrawMouse() which draws the target instead of the default mouse pointer(cut the code from Main sub). Call this sub inside the Game Loop .

**NOTE:** You will need to have mousePoint = Input.GetMousePosition() both in the Main procedure and in the DrawMouse procedure. mousePoint has to be declared in both places as well.

*cha 2 - worksheet.png*Question 1: *Why is it that mousePoint has be declared in both sub procedures Main() and DrawMouse()? Explain your answer.*

saveicon.png Remember to save your project (File – Save All). Once you have finished then you can close Visual Studio or move on to the next chapter.