COMP 1010- Summer 2015 (A01)

Jim (James) Young

young@cs.umanitoba.ca

jimyoung.ca

functions and scope

note: scope is the range within which a variable exists. Outside of that scope, you cannot access or work with that variable.

blocks define scope!

Functions each have their own scope – data in one method is not visible to other methods – try it

variable scope is limited to its function


```
void sayHello()
{
  text(message,100,100);
}

void draw()
{
  String message = “Hey there!”;
  sayHello();
}
```

note: a variable declared in one function is not **accessible** from another function!

you cannot read it, or change it.

No nesting of scopes here

the same variable name?

```
void sayHello()
{
  String message = “say hi”;
  println(message);
}

void draw()
{
  String message = “Hey there!”;
  sayHello();
  println(message);
}
```

variables in different scopes can have the same name, but they do not share data – they are completely separate.

function parameters

Reusing code…

drawRandomSquares – great!

what if we want variation?

draw different sizes?

draw different colors?

We need to give parameters to the function.

Send data to a function-> parameters

instead of

drawRandomSquares();

You want to do:

drawRandomSquares(20); // max size

Or

drawRandomSquares(20, 255); // max size, color

the numbers are parameters

Send data to a function-> parameters

void functionName (**parameterType parameterName**)

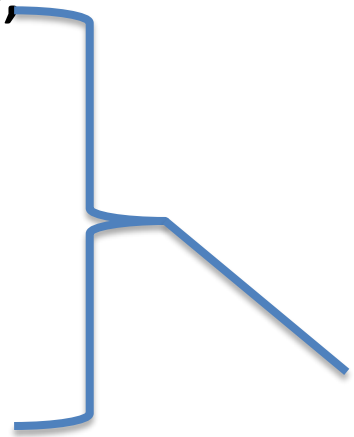
You can add parameters to the function

```
void drawRandomSquares() {
..
}

void drawRandomSquares(int size) {
..
}
```

user-defined functions: syntax

```
void methodName (parameterType parameterName)
{
   …//processing code;
}
```



function body

function header

How to use parameters inside a function?

```
void drawRandomSquares(int size) {
        println(size);
}
```

the parameters become local variables

data from outside is **copied** into the size variable and you can use it within the function just as any other variable.

Update drawRandomSquares

function parameters and… data is always copied

```
void changeData(int data) {
        data = data * 2;
}



void draw()
{
        int data = 4;
        changeData(data);
        println(data);
}
```

the information is copied here. inside the function you only have a copy. changes here do not reflect back!

Variant - keep tunnel vision!!!!!!

```
void changeData(int data) {
        data = data * 2;
}


void draw()
{
        int number = 4;
        changeData(number);
        println(number);
}
```

this variable name “data”
is important for inside the function
only. not related to how function is used

because information is copied, the
“caller” can give a literal, a variable,
do calculations, etc… NAME is irrelevant

another example…

make a function called statusPrint which prints a String message surrounded by -- and with its length appended in parenthesis. Print it in the bottom left corner

: e.g.,

if you call:

statusprint (“Hi!”);

it will print:

-Hi!- (3)

multiple parameters

a function can have as many parameters as you like, and they can be of different types!!

Let’s update drawRandomSquare to also take the color

multiple parameters – just use commas.

```
void drawRandomSquare(int maxSize, int color) {
…
}
```

you can mix types, e.g., have a double, a String, etc.

you can have as many parameters as you want, usually <= ~3