

Drawing Functions Recap

- create a drawing function for compound things (2-3 or more shapes involved), especially if repeated in the scene
- must *define* and *call* each function

```
void setup () {
  size(600,400);
}

void draw () {
  background(100);
  ellipseMode(CENTER);
  fill(255);
  stroke(0);
  ellipse(300,325,150,150);
  ellipse(300,200,100,100);
  ellipse(300,125,50,50);
}
```

without functions

```
void setup () {
  size(600,400);
}

void draw () {
  background(100);
  drawSnowman();
}

void drawSnowman () {
  ellipseMode(CENTER);
  fill(255);
  stroke(0);
  ellipse(300,325,150,150);
  ellipse(300,200,100,100);
  ellipse(300,125,50,50);
}
```

with functions



call

definition

More Than One Copy

this draws two snowmen, but exactly the same thing is drawn each time so one is on top of the other

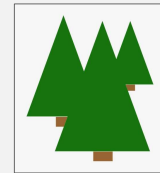
```
void draw () {
  background(100);
  drawSnowman();
  drawSnowman();
}
```

```
void drawSnowman () {
  ellipseMode(CENTER);
  fill(255);
  stroke(0);
  ellipse(300,325,150,150);
  ellipse(300,200,100,100);
  ellipse(300,125,50,50);
}
```

- if you want more than one copy, there has to be *some* difference between them

- position
- size
- color
- ...

- the solution is *parameterized* functions



Drawing Function Questions, Part 2

For the drawing function definition –

- put the definition outside other function definitions
- What is being drawn? → function *name* (and comment)
 - e.g. tree, car, ...
 - just one purpose!
- What differs from one copy to the next? → *parameters*
 - also consider future flexibility (within reason) – there might be only one copy now, but maybe you are thinking about some extra credit enhancements...
- How is it drawn? → function *body*
 - the drawing commands – include all necessary state (rectMode/ellipseMode, stroke, fill, etc) as well as the shapes
 - scope – can use parameters, system variables
 - it is legal to use animation variables but better to use parameters instead

Drawing Function Questions, Part 2b

For each parameter –

- What is it for? → parameter *name* (and comment)
- What kind of value is it? → parameter *type*
 - whole number – int
 - number with decimal point – float
 - true or false – boolean

Defining a Drawing Function With Parameters

What is being drawn? - a tree
What differs from one copy to the next? - position (x and y)
For each parameter -
What is it for? - position of the center of the bottom of the trunk
What kind of value is it? - number, specifically in int

```
// draw a tree
// (x,y) is the center of the bottom of the
// trunk
void drawTree (int x, int y) {
  ...
}
```

parameters
· declare using type and name (like variables)

name of the function
· must be different from other names being used
· should be descriptive of function's purpose

comments describe what the function does (draw a tree) and what its parameters are for (the position of the center of the bottom of the trunk)

Defining a Drawing Function With Parameters

```
// draw a tree
// (x,y) is the center of the bottom of the
// trunk
void drawTree ( int x, int y ) {
  ...
}
```

comments describe what the function does (draw a tree) and what its parameters are for (the position of the center of the bottom of the trunk)

Comments are essential for abstraction – they must describe everything someone needs to know to use the function without seeing the body.

- what the function does (draw a tree)
- what the parameters are for (tree position), including specifics (the position is the center of the bottom of the trunk)

Defining a Drawing Function With Parameters

```
// draw a tree
// (x,y) is the center of the bottom of the
// trunk
void drawTree ( int x, int y ) {
  rectMode(CENTER);
  stroke(0);
  // trunk
  fill(144,63,0);
  rect(x,y-20,15,40);
  // tree top
  fill(21,95,16);
  triangle(x-30,y-40,x,y-160,
          x+30,y-40);
}
```

How is it drawn?

function body – what the function does

use parameters like variables – but no initialization or update!

Drawing Function Questions, Part 3

For the function call(s) –

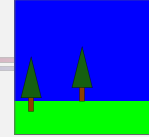
- When should the task be performed? → function *call*
 - put the call where you want it to happen
- What are the specific values for the things that can vary? → *arguments*

Calling a Drawing Function With Parameters

```
void setup () {
  size(400,400);
}
void draw () {
  background(0,0,255);
  // grass
  rectMode(CORNER);
  fill(0,255,0);
  stroke(0,255,0);
  rect(0,height-100,width,100);
  // trees
  drawTree(width/2,height-100);
  drawTree(50,height-70);
}
```

this is where the parameters get their values
list values in the same order as in the function definition

Full Example – Function With Parameters



```
void setup () {
  size(400,400);
}
void draw () {
  background(0,0,255);
  // grass
  rectMode(CORNER);
  fill(0,255,0);
  stroke(0,255,0);
  rect(0,height-100,
      width,100);
  // trees
  drawTree(width/2,
           height-100);
  drawTree(50,height-70);
}

// draw a tree
// (x,y) is the center of
// the bottom of the trunk
void drawTree ( int x,
               int y ) {
  rectMode(CENTER);
  stroke(0);
  // trunk
  fill(144,63,0);
  rect(x,y-20,15,40);
  // tree top
  fill(21,95,16);
  triangle(x-30,y-40,
          x,y-160,
          x+30,y-40);
}
```