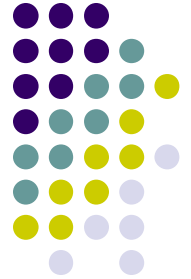


# Functions, Part 2 of 2

## Topics

- Functions That Return a Value
- Parameter Passing
- Local Variables

# Functions Can Return Values

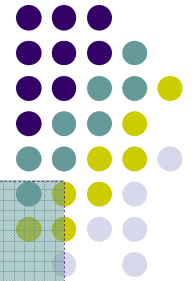


```

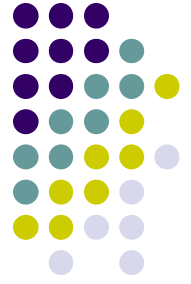
/*****
** AverageTwo - calculates and returns the average of two numbers
** Inputs:  num1 - a number
**          num2 - a number
** Outputs: the average of num1 and num2
*****/
function AverageTwo (num1, num2)
{
    var average;    /* average of the two numbers */
    average = (num1 + num2) / 2;
    return average;
}

```

# Using AverageTwo



```
<head>
<title>AverageTwo Example</title>
<script type="text/javascript">
  <!--
    function AverageTwo(num1, num2)
    {
      var average;
      average = (num1 + num2) / 2;
      return average;
    }
  //-->
</script>
</head>
<body>
  <script type="text/javascript">
    <!--
      var ave, value1 = 5, value2 = 8;
      ave = AverageTwo(value1, value2);
      alert("The average is " + ave);
    //-->
  </script>
</body>
```



# Parameter Passing

- **Actual parameters** (a.k.a. the arguments) are the parameters that appear in the function call.

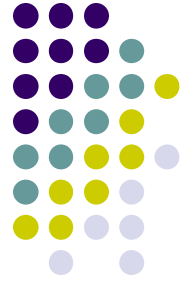
```
average = AverageTwo (value1, value2) ;
```

- **Formal parameters** are the parameters that appear in the function header.

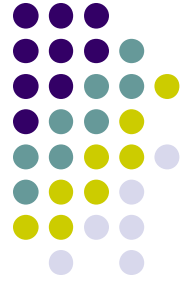
```
function AverageTwo (num1, num2)
```

- Actual and formal parameters are matched by position. Each formal parameter receives the value of its corresponding actual parameter.

# Parameter Passing



- Corresponding actual and formal parameters do not have to have the same name, but they may.



# Local Variables

- Functions only “see” (have access to) their own **local variables**.
- Formal parameters are declarations of local variables. The values passed are assigned to those variables.
- Other local variables can be declared within the function body.

# Parameter Passing and Local Variables



```
<head>
<title>AverageTwo Example</title>
<script type="text/javascript">
  <!--
    function AverageTwo(num1, num2)
    {
      var average;
      average = (num1 + num2) / 2;
      return average;
    }
  //-->
</script>
</head>
```

num1

num2

average

```
<body>
<script type="text/javascript">
  <!--
    var ave, value1 = 5, value2 = 8;
    ave = AverageTwo(value1, value2);
    alert("The average is " + ave);
  //-->
</script>
</body>
```

value1

value2

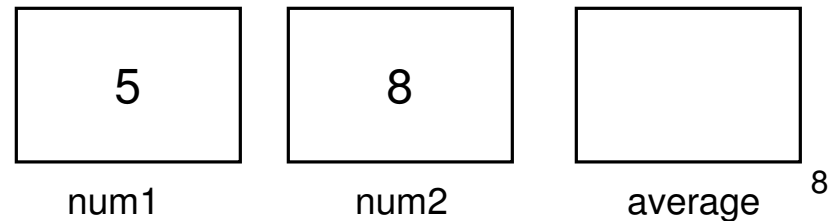
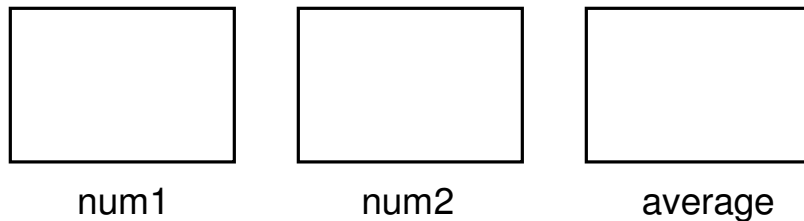
ave

# Same Name, Still Different Memory Locations



```
<head>
<title>AverageTwo Example</title>
<script type="text/javascript">
  <!--
    function AverageTwo(num1, num2)
    {
      var average;
      average = (num1 + num2) / 2;
      return average;
    }
  //-->
</script>
</head>
```

```
<body>
<script type="text/javascript">
  <!--
    var average, num1 = 5, num2 = 8;
    average = AverageTwo(num1, num2);
    alert("The average is " + average);
  //-->
</script>
</body>
```





# Changes to Local Variables Do NOT Change Other Variables with the Same Name



```
<head>
<title>AddOne Example</title>
<script type="text/javascript">
  <!--
  function AddOne(num1)
  {
    num1 = num1 + 1;
    alert("In AddOne: num1 = " +
          num1);
  }
  //-->
</script>
</head>
```



num1

```
<body>
  <script type="text/javascript">
    <!--
      var num1 = 5;
      AddOne(num1);
      alert("In the body: num1 = " +
            num1);
    //-->
  </script>
</body>
```



num1