

# Adobe Flash

## Actionscript 2

# **Actionscript 2**

## **BACKGROUND**

- ▶ **Object Oriented Programming Language**
- ▶ **Based On Javascript**
- ▶ **Compiled Language**

# **Actionscript 2**

## **BACKGROUND**

- ▶ **Object Oriented Programming Language**
  - Objects (movieclips, buttons, variables) are called by name
  - Script is separate from the content
- ▶ **Based On Javascript**
- ▶ **Compiled Language**

# Actionscript 2

## BACKGROUND

- ▶ **Object Oriented Programming Language**
- ▶ **Based On Javascript**
  - Format is similar
  - Blocks of code must be closed properly `}` `()` ;
- ▶ **Compiled Language**

# Actionscript 2

## BACKGROUND

- ▶ **Object Oriented Programming Language**
- ▶ **Based On Javascript**
- ▶ **Compiled Language**
  - Requires web browser (client) to have a Flash Player installed
  - Faster/More Stable
  - More Secure (hard to get to other peoples code)

# **Actionscript 2**

## **OBJECTS & NAMES**

- ▶ **Content Types**
- ▶ **What's In a Name**
- ▶ **Paths to content**

# Actionscript 2

## OBJECTS & NAMES

### ▶ **Content Types**

- Movieclip (mc, button, \_root) - contains a timeline
- Function - command that flash understands
- Variable - the way flash stores information (text, numbers, paths, etc)

### ▶ **What's In a Name**

### ▶ **Paths to content**

# Actionscript 2

## OBJECTS & NAMES

- ▶ **Content Types**
- ▶ **What's In a Name**
  - Instance Names allow us to point script at specific items
  - CANNOT begin with a number
  - Suffix provides code hints (`_mc`, `_btn`)
- ▶ **Paths to content**

# **Actionscript 2**

## **OBJECTS & NAMES**

- ▶ **Content Types**
- ▶ **What's In a Name**
- ▶ **Paths to content**
  - Tells Flash where something lives
  - Similar to an address
  - Uses a “dot” notation
  - Can be relative or specific

# Actionscript 2

## OBJECTS & NAMES

- ▶ **Content Types**
- ▶ **What's In a Name**
- ▶ **Paths to content**
  - Tells Flash where something lives
    - `containerClip1, containerClip2, target Item`
  - Similar to an address
  - Uses a “dot” notation
  - Can be relative or specific

# Actionscript 2

## OBJECTS & NAMES

- ▶ **Content Types**

- ▶ **What's In a Name**

- ▶ **Paths to content**

- Tells Flash where something lives
- Similar to an address

website addresses uses /

mydomain.com/somefolder/file.html

mac addresses use :

Macintosh HD : User : Applications : flash.app

- Uses a “dot” notation
- Can be relative or specific

# Actionscript 2

## OBJECTS & NAMES

- ▶ **Content Types**
- ▶ **What's In a Name**
- ▶ **Paths to content**
  - Tells Flash where something lives
  - Similar to an address
  - Uses a “dot” notation

`movieClipInstanceName.movieClipInstanceName.targetItem`

- Can be relative or specific

# Actionscript 2

## OBJECTS & NAMES

- ▶ **Content Types**

- ▶ **What's In a Name**

- ▶ **Paths to content**

- Tells Flash where something lives
- Similar to an address
- Uses a “dot” notation
- Can be relative or specific

relative (from item) `this._parent._parent`

specific (from root) `_root.instanceName.instanceName`

# **Actionscript 2**

## **WRITING & SYNTAX**

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
  - myName:var = “Bryce”;
  - play();
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
  - `()`, `{}`, `“”`, `“`
  - `myName:var = “Bryce”;`
  - `gotoAndPlay(argument);`
- ▶ **Giving Orders**

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**
  - functions are orders or procedures for flash to follow
    - `stop();` tells flash to “stop playing” the timeline
  - Unless told otherwise flash assumes you want to change “this”
  - Flash comes with simple functions
  - Some Functions take arguments
  - Or you can write your own functions

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**
  - functions are orders or procedures for flash to follow
  - Unless told otherwise flash assumes you want to change “this”
    - `stop();` would apply to the current timeline
    - `circleClip_mc.stop();` would stop the timeline of a clip with the instance name of “circleClip\_mc”
  - Flash comes with simple functions
  - Some Functions take arguments
  - Or you can write your own functions

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**
  - functions are orders or procedures for flash to follow
  - Unless told otherwise flash assumes you want to change “this”
  - Flash comes with simple functions

`stop();`

`play();`

`gotoAndStop();`

- Some Functions take arguments
- Or you can write your own functions

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**
  - functions are orders or procedures for flash to follow
  - Unless told otherwise flash assumes you want to change “this”
  - Flash comes with simple functions
  - Some Functions take arguments
    - arguments give the function more information
  - `gotoAndStop(frame name or number)` tells Flash where to “goto”
  - Or you can write your own functions

# Actionscript 2

## WRITING & SYNTAX

- ▶ **Commands must end with a semi colon**
- ▶ **Blocks are opened and closed with like characters**
- ▶ **Giving Orders**
  - functions are orders or procedures for flash to follow
  - Unless told otherwise flash assumes you want to change “this”
  - Flash comes with simple functions
  - Some Functions take arguments
  - Or you can write your own functions

# Actionscript 2

## USER INTERACTION

### ▶ **How do we make a button work?**

- Find the Button
- Choose an action (button press, roll over, roll off, etc)
- Decide what happens

# Actionscript 2

## USER INTERACTION

### ▶ **How do we make a button work?**

- Find the Button

`(this)portfolio_btn.`

- Choose an action (button press, roll over, roll off, etc)
- Decide what happens

# Actionscript 2

## USER INTERACTION

### ▶ How do we make a button work?

- Find the Button

```
(this)portfolio_btn.
```

- Choose an action (button press, roll over, roll off, etc)

```
(this)portfolio_btn.onRelease = function(){  
}
```

- Decide what happens

# Actionscript 2

## USER INTERACTION

### ▶ How do we make a button work?

- Find the Button

```
(this)portfolio_btn.
```

- Choose an action (button press, roll over, roll off, etc)

```
(this)portfolio_btn.onRelease = function(){  
}
```

- Decide what happens

```
(this)portfolio_btn.onRelease = function(){  
    _root.gotoAndStop(10);  
}
```

# **Actionscript 2**

## **TROUBLE SHOOTING & MISC**

- ▶ **Comments (help yourself and others follow your code)**
- ▶ **Error Console**
- ▶ **Trace Command**
- ▶ **Check your code (common problems)**

# Actionscript 2

## TROUBLE SHOOTING & MISC

- ▶ **Comments (help yourself and others follow your code)**
  - Text that is hidden from flash
  - single line comments simply start with //
  - multi-line comments start and end `/* Comment Text */`
- ▶ **Error Console**
- ▶ **Trace Command**
- ▶ **Check your code (common problems)**

# **Actionscript 2**

## **TROUBLE SHOOTING & MISC**

- ▶ **Comments (help yourself and others follow your code)**
- ▶ **Error Console**
- ▶ **Trace Command**
- ▶ **Check your code (common problems)**

# Actionscript 2

## TROUBLE SHOOTING & MISC

- ▶ **Comments (help yourself and others follow your code)**
- ▶ **Error Console**
- ▶ **Trace Command**
  - Trace allows you to ask Flash questions
  - Answers might not always be what you expect!
  - Trace(“this button was clicked”);
- ▶ **Check your code (common problems)**

# Actionscript 2

## TROUBLE SHOOTING & MISC

- ▶ **Comments (help yourself and others follow your code)**
- ▶ **Error Console**
- ▶ **Trace Command**
- ▶ **Check your code (common problems)**
  - Capitalization matters

myclip, MyClip, and myClip are all different to Flash

# Actionscript 2

## TROUBLE SHOOTING & MISC

- ▶ **Comments (help yourself and others follow your code)**
- ▶ **Error Console**
- ▶ **Trace Command**
- ▶ **Check your code (common problems)**
  - Capitalization matters
    - myclip, MyClip, and myClip are all different to Flash
  - Specific commands must end with a semi colon
    - `gotoAndStop(10); //works`
    - `gotoAndStop(10) // will cause an error`

# Actionscript 2

## TROUBLE SHOOTING & MISC

- ▶ **Comments (help yourself and others follow your code)**
- ▶ **Error Console**
- ▶ **Trace Command**
- ▶ **Check your code (common problems)**
  - Capitalization matters
    - myclip, MyClip, and myClip are all different to Flash
  - Specific commands must end with a semi colon
    - `gotoAndStop(10); //works`
    - `gotoAndStop(10) // will cause an error`
  - Be sure to close objects using like characters `()`, `{}`, `''`

# **Actionscript 2**

## **ACTIONSCRIPT DEMO**