

Coding Convention

Where possible, conventions should strive to emulate the coding conventions already established by Adobe. Coding conventions are vital in improving code maintainability, particularly in a team environment and therefore are mandated unless explicitly stated.

Variable Naming

Variable names are **camel case, first letter lowercase**. **No abbreviations** unless brutally obvious or a temporary variable. The name should be **totally descriptive** of what the variable does. Do not be afraid of long variable names.

```
playerVelocity (not velocity or pVelocity or pv)
remainingHealth (not health)
timeRemainingUntilFailure (not timeRemaining or time)
```

Function Naming

Function names follow the **same rules as variable names**. They must **start with verbs** because all functions are **actions**. They usually **end with nouns**.

```
function movePlayer()
function killPlayer()
```

Boolean Naming

Boolean variable names begin with **is, has, or can**. To prevent confusion, public properties are recommended over functions that return the value of a member boolean variable.

```
isPlayerStanding
hasPlayerGottenKey
```

Class Members

Class member variables are prefixed with “**m_**” and must always be **private**.

```
m_remainingLives
m_moneyInAccount
```

Function Arguments

Function argument variables are prefixed with “**a_**” and must always be treated **read-only**.

```
function doSomething( a_importantNumber : int, a_fullName :
String )
```

Constants

Constants are upper case and use an underscore instead of camel casing to separate words.

```
MAXIMUM_COLUMNS  
FRAME_WIDTH
```

Bracketing

Brackets are always on their **own line**.

```
function liveLife() : void  
{  
    doVariousMenialTasks();  
    profit();  
}
```

Block Structures

All **looping, branching, and exception** blocks must **use brackets** even if the body is one line.

```
if( isPlayerAlive )  
{  
    doSomethingUseful();  
}  
else  
{  
    stinkLikeACorpse();  
}  
  
while( true )  
{  
    beStuckInAnInfiniteLoop();  
}
```

Long Lines

Exceptionally long lines should be continued on the next line after an **indent**.

```
submitTest( boyThisFunctionTakesALotOfParameters,  
            thereforeIShouldIndentHereOnSubsequentLines, true,  
            false, false, false, true, 'c', 'b', 'b', 'a', 'b',  
            "William Jennings Brian" );
```

Parenthesis

All parenthesis have **spaces inside**, unless they are **empty**.

```
if( isDoomsday )
{
    stealANiceTv();
    playXbox360( "Burnout Revenge" );
}
```

Member Variable Modification

Member variables (marked with **m_**) should only be modified **inside the class that owns them**. This typically means **avoiding getter and setter methods**.

```
function attackPlayer() : void
{
    //NO NO NO
    var p : Player = new Player();
    var maxHp : int = p.getMaxHP();
    var damage : int = p.getMaxHP() * .10;
    p.takeDamage( damage );

    //yes!!!
    var p : Player = new Player();
    p.takeDamagePercent( 10 );
}
```