

Тема 1. Рабочая среда и язык ActionScript

Практические задания

Введение

В практических заданиях содержатся постановки задач и коды на языке ActionScript 3.0 для программирования основных интерактивных элементов Flash. Дополнительно прилагаются файлы, созданные в Adobe Flash Professional CS5.5.

Для выполнения заданий предварительно необходимо изучить теоретические материалы по программированию на ActionScript 3.0 для тем 1 и 2. Кроме того, необходимо владеть практикой создания графических символов, клипов, кнопок и анимаций во Flash. Эти навыки осваиваются в дистанционном курсе «Использование системы Adobe Flash для разработки и программирования Web-сайтов»

В этой теме вы познакомитесь с включением кода в документ Flash, выполните анализ кода.

Переменные

Переменные, это контейнеры, которые содержат в себе какие-либо значения или данные. Для создания переменной всегда нужно использовать слова `var`. Тип переменной назначается добавлением к имени переменной двоеточия с последующим указанием типа этой переменной.

Примеры

```
var x:Number = 5; // вводится переменная x = 5. Тип Number - число
var text:String = «Привет»; // вводится переменная text, означающая слово «Привет». Тип
String – строка.
var y:Number; //так тоже можно, без присваивания значений
```

Внимание: переменная `text` не равна переменной `Text`, другими словами – имена переменных зависят от регистра.

Событие клавиатуры:

Событие будет происходить при нажатии клавиши.
Существует два типа событий клавиатуры: `KeyboardEvent.KEY_DOWN` и `KeyboardEvent.KEY_UP`
`KeyboardEvent.KEY_DOWN` – пользователь нажал на клавишу
`KeyboardEvent.KEY_UP` – пользователь отпустил клавишу.

События мыши изучаются на практике в *Теме 2*.

Проблемы последнего кадра

Пример находится в файле *Тема 1_1*.

Создайте следующий сценарий:

В основной временной шкале в первом кадре создайте анимацию вращения прямоугольного объекта.

1. Для этого предварительно создайте графический символ типа **Graphic**, затем на основе этого символа с помощью команды **Insert >New Symbol** создайте клип с анимацией вращения между 1 и 30 ключевыми кадрами.
2. В меню **Insert** выполните команду **Classic Tween**. На панели **Properties** раскройте раздел **Tweening**, в списке **Rotate** установите вариант **CW** (по часовой стрелке), число оборотов 3.
3. Перетащите экземпляр созданного клипа на основную сцену и присвойте ему имя **m11**. Нажмите **Ctrl+Enter**, чтобы провести тестирование клипа. Вы обнаружите, что в конце вращения объект останавливается на некоторое время. Это объясняется тем, что положение объекта в первом и последнем ключевых кадрах (**КК**) совершенно одинаковое, и эти кадры видимы подряд.
4. На основной шкале добавьте новый слой, присвойте ему имя **script** и в первый ключевой кадр этого слоя с помощью панели **Actions** добавьте код:

```
function everyFrame(event:Event):void
{
    if (m11.currentFrame == 1)
    {
        m11.visible = true;

    }
    if (m11.currentFrame == m11.totalFrames)
    {
        m11.visible = false;
    }
}
m11.addEventListener(Event.ENTER_FRAME, everyFrame);
```

Пояснение кода.

Метод `addEventListener()` создает прослушивателя событий, будет проверять, не произошло ли событие в сцене. В скобках задается тип события. В данном случае это событие `Event.ENTER_FRAME` – при переходе в кадр. После запятой указывается имя функции, которая вызывается, если происходит событие. В примере функция названа `everyFrame`. В функции указано, что в первом кадре экземпляр клипа с именем `m11` виден (`m11.visible = true`), а в последнем кадре клип становится невидимым (`if (m11.currentFrame == m11.totalFrames)`

```
{
    m11.visible = false;
}
```

3. Снова запустите анимацию. Теперь при проигрывании изображение последнего кадра на экран не выводится.

Примечание.

Для решения проблемы (без программирования) можно добавить ключевой кадр в предпоследний кадр клипа и удалить последний ключевой кадр.

Переход по URL

Пример находится в файле *Тема 1_2*.

Командой **Insert > New Symbol** создайте символ кнопки с именем **b1**. Создайте новый слой с именем **buttons** и перетащите экземпляр кнопки в первый КК этого слоя. Присвойте экземпляру кнопки имя *ButtonURL*.

На основной шкале добавьте новый слой, присвойте ему имя **script** и в первый ключевой кадр этого слоя с помощью панели **Actions** добавьте код:

```
function clickHandler(event:MouseEvent):void
{
    navigateToURL(new URLRequest("http://www.limtu.ru"));
}
ButtonURL.addEventListener(MouseEvent.CLICK,clickHandler);
```

Пояснение кода.

Прослушиватель реагирует на щелчок по кнопке с именем *ButtonURL*. Когда происходит такое событие, выполняется переход на URL с именем *http://www.limtu.ru*".

Для перехода по ссылке нажмите <Ctrl>+<Enter> и щелкните по кнопке.

Анимация динамических клипов

Перед выполнением следующих практических заданий изучите основные свойства объектов:

Свойство alpha

Указывает значение альфа-прозрачности заданного объекта. Допустимые значения находятся в диапазоне 0 (полностью прозрачный) — 1 (полностью непрозрачный).

Значение по умолчанию 1. Экранные объекты, для которых значение alpha равно 0, являются активными, несмотря на невидимость.

Свойство height, свойство width

Указывает высоту (ширину) экранного объекта в пикселах.

Свойство rotation

Указывает поворот в градусах. Значения от 0 до 180 представляют поворот по часовой стрелке. Значения от 0 до -180 представляют поворот против часовой стрелки. Значения за пределами этого диапазона, добавляются или вычитаются из 360 для получения допустимого значения. Например, rotation = 450 равносильно rotation = 90.

Свойство `scaleX`, `scaleY`

Указывает горизонтальное (вертикальное) масштабирование объекта от точки регистрации (в процентах). Точка регистрации по умолчанию — (0, 0). 1,0 равняется масштабу 100%.

Свойство `x` ; свойство `y`

Указывает координату `x` (`y`) объекта. Координаты объекта определяются относительно положения точки регистрации.

Следующий пример находится в файле *Тема 1_3*.

1. Создайте документ 300 x 200 пикселей.
2. Создайте клип с именем `m1`, определите его класс `BlueCircle`. Для этого щелкните правой кнопкой в библиотеке на имени клипа, выберите команду `Properties`, и в диалоговом окне `Symbol Properties` (рис. 1.1) присвойте клипу класс `BlueCircle`.

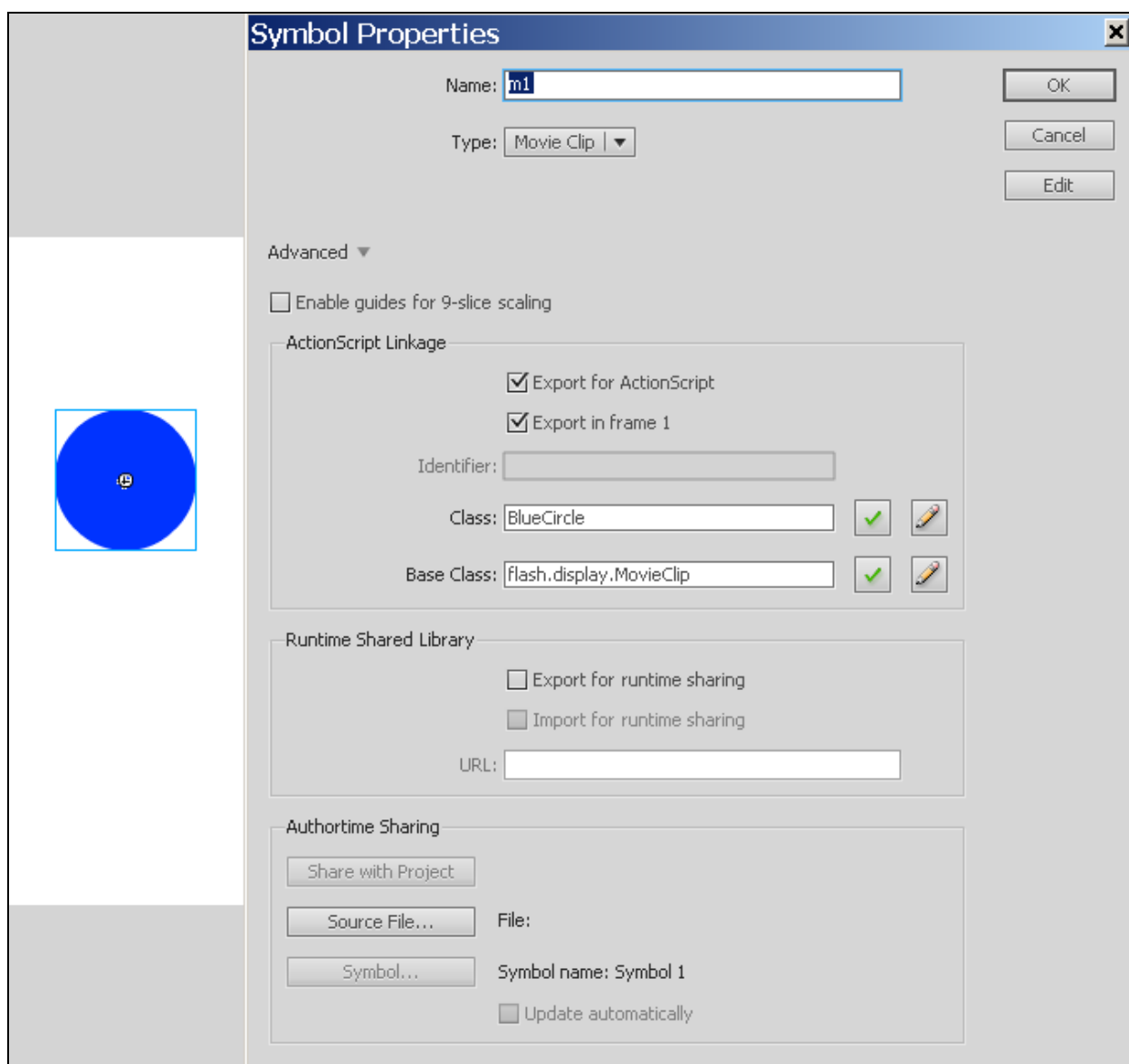


Рис. 1.1

3. Далее в первый кадр основной сцены добавьте следующий код:

```
function Main() {  
    // Добавление на сцену события  
    stage.addEventListener(MouseEvent.CLICK, AddCircle);  
}  
Main();  
  
function AddCircle(e:MouseEvent):void {  
    // Добавление на сцену круга  
    var newCircle:BlueCircle = new BlueCircle();  
    this.addChild(newCircle);  
  
    // Задание координат круга X и Y  
    newCircle.x = mouseX;  
    newCircle.y = mouseY;  
  
    // Задание значений масштаба и прозрачности  
    newCircle.scaleX = 0;  
    newCircle.scaleY = 0;  
  
    newCircle.alpha = 0;  
  
    // Добавление события ENTER_FRAME ( event listener)  
    newCircle.addEventListener(Event.ENTER_FRAME, ZoomCircle);  
}  
  
function ZoomCircle(e:Event):void {  
    // Создание круга щелчком  
    var circleMC:MovieClip = MovieClip(e.target);  
  
    // Приращение масштабов  
    circleMC.scaleX += .05;  
    circleMC.scaleY += .05;  
  
    // Задание условия изменения прозрачности  
    if (circleMC.scaleX < 2) {  
        circleMC.alpha += .03;  
    } else {  
        circleMC.alpha -= .03;  
  
        // Остановка события ENTER_FRAME при большой прозрачности  
        if (circleMC.alpha < .1) {  
            circleMC.removeEventListener(Event.ENTER_FRAME, ZoomCircle);  
        }  
    }  
}
```

```

    }
  }
}

```

Выполните тестирование клипа, щелкнув кнопки <Ctrl>+<Enter>. Для работы кода выполните несколько щелчков указателем мыши на экране. Изучите тщательно код.

Класс Sprite

Пример находится в файле *Тема 1_4*.

Класс Sprite является базовым классом, с помощью которого можно создавать клипы и их экземпляры программным путем. При этом создаваемые клипы не имеют Timeline. Описание кода показано далее.

//Создание прямоугольного объекта типа Sprite

```

var rectangle:Sprite = new Sprite();
rectangle.graphics.lineStyle(0,0x7B7B7B); //lineStyle(thickness, color)
rectangle.graphics.beginFill(0xA6A6A6); //beginFill(color)
rectangle.graphics.drawRect(0,0,60,60); //drawRect(x,y,width,height)
rectangle.graphics.endFill(); //endfill

```

var dot:Shape = new Shape(); //Небольшой объект типа Shape

```

dot.graphics.lineStyle(0,0xFF0000); //lineStyle(thickness, color)
dot.graphics.beginFill(0xFF0000); //beginFill(color)
dot.graphics.drawCircle(0,0,2); //drawCircle(x,y,radius)
dot.graphics.endFill(); //endfill

```

rectangle.addChild(dot); //Добавление объекта dot к прямоугольнику

dot.x = 1; //x position -> 1

dot.y = 1; //y position -> 1

this.addChild(rectangle); //Добавление прямоугольника на сцену

rectangle.x = 200; //x position -> 200

rectangle.y = 200; //y position -> 200

*/**

Добавление прослушивателя на сцену, и связывание его с функцией rotateRectObject

**/*

```

stage.addEventListener(Event.ENTER_FRAME, rotateRectObject);

```

```

function rotateRectObject(evt:Event):void {

```

//функция вращения прямоугольного объекта

```

rectangle.rotation+=10;

```

```

}

```

Выполните тестирование и анализ клипа.