

ЛАБОРАТОРНАЯ РАБОТА «ВНЕДРЕНИЕ ВИДЕО»

Программа *Flash* работает со следующими форматами файлов:

- **MOV** — формат **Quick Time** фирмы *Apple*;
- **AVI** (*Audio Video Interleave*) — стандартный видеоформат *Windows*;
- **ASF, WMV** — потоковые форматы *Windows Media*;
- **MPEG** (*Motion Picture Expert Group*);
- **DV, DVI** — форматы *Digital Video*;
- **FLV** (*Flash Video*) — «родной» формат видео для *Flash*;
- **3GP, 3GPP** — видеоформаты для мобильных устройств.

При импорте все видеофайлы переводятся в формат **FLV**, который является «родным» форматом для *Flash*.

Существует два способа использования видео:

- *внедрить* видео в фильм, при этом объем *Flash*-ролика увеличивается на размер видеофайла;
- *постепенно загружать* видео в процессе работы с жесткого диска или из Интернета (только для формата **FLV**).

Внедренное видео загружается в основную временную шкалу или внутрь отдельного клипа. В этом случае можно добавить новые слои с другими элементами, которые будут точно синхронизированы с видеоизображением (например, титры).

Задание 1: включить видео в флеш-ролик, разработать кнопки для управления видео.

1. Создайте новый документ (Action Script 2.0.). Создайте в нем слой *Фон*, а также слой *Кнопки*.
2. Создайте новый слой *Видео*. Выполните команды **File** → **Import** → **Import video**.
3. В окне **Select Video** щелкните по кнопке **Browse** (*Обзор*) и выберите видеофайл, перейдите к следующему окну.
4. В окне **Deployment** выберите вариант **Embed video in SWF and play in timeline** (внедрить видео в SWF-файл и проигрывать на временной шкале). В остальных окнах мастера оставьте все параметры без изменений.

В частности, оставьте отмеченными варианты

- **Place instance on stage** — разместить видео на сцене;
- **Expand timeline if needed** — увеличить временную шкалу, если необходимо;
- **Embed the entire video** — вставить видео целиком.

Предварительно с видеофайлом также можно выполнить следующие действия:

- разделить звук и изображение, отключить звук вообще;
- поместить видео на главную временную шкалу или в отдельный клип;
- вырезать отдельные куски из видеофильма;
- обрезать поля, то есть, взять только часть картинки;
- изменить размер клипа.

Когда импорт видео завершится, в библиотеке вы обнаружите новый символ типа **Embedded Video** (внедренное видео). Его копия будет на сцене.

5. Выделите на сцене видео фильм и установите на панели **Properties** координаты **X=0** и **Y=0**. Просмотрите фильм.

6. Временная шкала для слоя *Видео* была автоматически расширена, а для других слоев это надо сделать вручную. Перейдите к последнему кадру видео и вставьте там промежуточные кадры в слоях *Фон* и *Кнопки*.

7. Создайте новый слой *Программа*, перейдите к кадру 1 и добавьте к нему команду `stop() ;`.

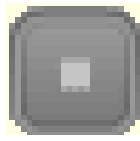


8. Выделите кнопку и присоедините к ней код

```

on (release) {
    if ( playing )
        _root.stop();
    else _root.play();
}

```



9. Выделите кнопку и присоедините к ней код:

```

on (release) {
    playBtn.gotoAndStop(1);
    gotoAndStop(1);
}

```

10. Для перехода на несколько кадров вперед или назад будем использовать два свойства главного монтажного стола:

- `_currentframe` – номер текущего кадра;
- `_totalframes` – общее число кадров в клипе.

Нужно ограничить перемещение, чтобы номер кадра, куда мы переходим, не стал меньше 1 или больше `_totalframes`. Для этого используются функции `max` и `min` объекта `Math`.

Если фильм сейчас проигрывается (`playBtn.playing=true`), после перехода надо продолжить проигрывание, то есть использовать переход `gotoAndPlay`. Если фильм остановлен, применяется метод `gotoAndStop`.



11. Выделите кнопку и добавьте к ней обработчик:

```

on (release) {
    frame = Math.max(1, _currentframe-10);
    if ( playBtn.playing )
        gotoAndPlay(frame);
    else gotoAndStop(frame);
}

```

Аналогичный код добавьте для кнопки быстрой перемотки вперед.

12. Ползунок-индикатор должен находиться на нулевой отметке, когда фильм находится на кадр 1, и на отметке 100% в конце фильма.

Клип *Индикатор* построен так, что при перетаскивании он вызывает функцию `_root.setFraction`, передавая ей в параметре `frac` значение смещения от 0 до 1. Легко проверить, что нужный номер кадра вычисляется как $1 + \text{frac} * (\text{_totalframes} - 1)$, а для округления к ближайшему целому значению нужно использовать метод `round`.

13. К первому кадру добавьте следующий код:

```

function setFraction ( frac ) {
    frame = Math.round(1+frac*( _totalframes-1 ));
    if ( playBtn.playing )
        gotoAndPlay(frame);
    else gotoAndStop(frame);
}

```

14. Обратная связь: при проигрывании изменяется номер текущего кадра `_currentframe` и нужно соответственно изменять положение движка. У клипа *Индикатор* есть метод `setFraction`, который принимает параметр от 0 до 1 — долю проигранной части фильма, которая вычисляется как $\text{part} = (\text{_root._currentframe} - 1) / (\text{_root._totalframes} - 1)$;

Вызов этой функции мы добавляем в обработчик enterFrame клипа. Выделите линию-индикатор и добавьте к ней обработчик события enterFrame:

```
onClipEvent (enterFrame) {  
    part = (_root._currentframe-1) / (_root._totalframes-1);  
    setFraction ( part );  
}
```

Проверьте окончательный вариант и сохраните файл.