

ActionScript3

Úvod do skriptovacieho jazyka

Tvorba tlačidiel a odkazov

Aplikácia Adobe Flash (hlavne novšie verzie CS3 – CS5) používa pre rozšírenie funkcionality programu jazyk ActionScript 3.0, čo je robustný skriptovací jazyk. ActionScript dodáva flashovým animáciám interaktivitu a profesionalitu.

Terminológia skriptovacích jazykov

Pri používaní skriptovacieho jazyka ActionScript 3 sa stretávame s týmito pojmi:

Premenná:

predstavuje určitú časť dát, ktorá môže či nemusí byť konštantná. Pri vytváraní, resp. definovaní premennej jej musíme prideliť aj typ, ktorý určuje, aký druh dát má premenná reprezentovať. Napr. premenná typu **String** uchováva akýkoľvek reťazec alfanumerických znakov, zatiaľ čo premenná typu **Number** musí obsahovať len číslo.

Názvy premenných musia byť jedinečné (názov sa nesmie opakovať), pričom záleží na veľkosti písmen. Napr. premenná *mypassword* nie je taká istá ako *Mypassword*. Názvy premenných môžu obsahovať len čísla, písmená a podčiarkovníky. Nesmú však začínať číslom.

Kľúčové slovo:

kľúčové slovo je v prostredí jazyka ActionScript vyhradené, kľúčové slovo, ktoré sa používa na vykonanie určitej úlohy. Napr. pomocou kľúčového slova *var* sa vytvára premenná. Úplný zoznam kľúčových slov sa nachádza v nápovede programu Flash. Vzhľadom k tomu, že sa jedná o vyhradené slová, nie je možné ich použiť ako názvy premenných alebo na iné účely.

Argumenty:

argumenty ďalej špecifikujú príkaz, ku ktorému sa vzťahujú. Píšu sa medzi okrúhle zátvorky (). Napr. v kóde *gotoAndPlay(3)*; argument hovorí, aby sa skript posunul na snímok 3.

Funkcia:

funkcia je skupina príkazov, ktoré môžeme označiť nejakým spoločným názvom.

Objekty:

v jazyku ActionScript 3.0 sa pracuje s objektmi, čo sú abstraktné typy dát, ktoré pomáhajú vykonávať určité úlohy. Napr. objekt typu *Sound* nám pomáha ovládať zvuk a objekt typu *Date* slúži pre prácu s dátami súvisiacimi s časom. Symboly tlačidiel, ktoré vytvárame sú zase objektmi typu *Button*.

Každý objekt by mal mať svoj vlastný názov. Na objekt, ktorý má meno sa dá potom v kóde jazyka odkazovať a pomocou kódu takýto objekt aj ovládame.

Tlačidlám vo vymedzenej ploche sa hovorí *inštancia*. Inštancie a objekty sú vo skutočnosti synonymá.

Metódy:

metódy sú kľúčové slová, ktorých výsledkom je nejaká akcia. Metódy sú „robotníkmi“ ActionScriptu, pričom každý objekt má svoju vlastnú sadu metód. Napr. objekt typu *MovieClip* okrem iných používa metódy *stop()* a *gotoAndPlay()*.

Vlastnosti:

vlastnosti popisujú objekt. Napr. medzi vlastnosti filmového klipu patria výška a šírka, súradnice x,y,z a mierka. Mnoho vlastností je možné meniť, zatiaľ čo iné je možné len čítať, čo znamená, že popisujú daný objekt.

Používanie správnej syntaxe vo skriptovacom jazyku

Porozumieť základnej syntaxi skriptovacieho jazyka, čo je v podstate gramatika a interpunkcia jazyka zabezpečí, že kód ActionScript sa stane pre nás porozumiteľný. Medzi používanú syntax patrí:

bodkočiarka – na konci riadku oznamuje ActionScriptu, že došiel nakoniec riadku, a že sa má presunúť na ďalší riadok kódu.

okružla zátvorka – každá okrúhla zátvorka musí mať odpovedajúcu uzatvárajúcu zátvorku, to platí aj pre hranaté a zložené zátvorky. Pokiaľ niečo otvoríme, musíme to tiež zatvoriť.


bodka – pomocou operátora bodky (`.`) je možné pristupovať k vlastnostiam a metódam objektu. Najprv sa napíše názov inštancie, potom nasleduje bodka a nakoniec názov vlastnosti

či metódy. Na bodku sa možno pozerat' aj ako na spôsob oddeľovania objektov, metód a vlastností.

úvodzovky - kedykoľvek napíšeme nejaký reťazec, musíme ho vložiť medzi úvodzovky.

komentáre - do skriptu je možné pridávať komentáre. Jednoriadkový komentár zadávame po dvoch lomkách (//), viacriadkový medzi lomky a hviezdičky (/ * */). Komentáre slúžia na popis jednotlivých segmentov kódu.

Aplikácia Flash má k dispozícii panel Actions, do ktorého sa píše kód ActionScript. Pri písaní kódu ponúka Flash nasledovnú pomoc:

- Slová, ktoré majú v ActionScript špecifický význam (napr. príkazy alebo kľúčové slová), sa pri písaní zobrazujú modrou farbou. Slová, ktoré sú v ActionScript vyhradené (napr. názvy premenných), budú čierne, reťazce zelené a komentáre, ktoré ActionScript ignoruje, zase šedé.
- Pri práci s panelom Actions vie Flash rozoznať akciu, ktorú práve píšeme a zobrazit' nám odpovedajúcu pomoc pri písaní kódu. Existujú dve druhy pomoci pri písaní kódu: rozvaľovania nápoveda, ktorá obsahuje úplnú syntax pre danú akciu a rozbaľovacia ponuka obsahujúca prehľad možných prvkov jazyka ActionScript.
- Medzi ďalšiu pomoc patrí možnosť kontroly správnosti písania kódu. K overovaniu syntaxi skriptu, ktorý práve píšeme, stačí kliknúť na ikonku „Skontrolovať syntax“ (Check Syntax) . Chyby syntaxi sa vypisujú na panely, ktorý sa nazýva „Chyby kompiátora“ (Compiler Errors).

VYTVORENIE OBSLUHY UDALOSTI PRE TLAČIDLO

Udalosť je situácia, ku ktorej dochádza v prostredí Flash, ktorý ju dokáže identifikovať a reagovať na ňu.

Medzi základné udalosti Flashu patrí:

- kliknutie myši
- pohyb myši
- stlačenie klávesy na klávesnici

Tieto udalosti produkuje používateľ, existujú však udalosti, ku ktorým dochádza nezávisle na používateľovi, ako je úspešné načítanie dát alebo dokončenie prehrávania zvuku.

Pomocou jazyka ActionScript môžeme písať kód, ktorý udalosti identifikuje a prostredníctvom obsluhy udalosti na ňu reaguje.

Prvým krokom pri obsluhu udalosti je vytvorenie **poslucháča**, ktorý bude identifikovať udalosť. Poslucháč môže vyzerat' napr. takto:

```
kde_počúvať.addEventListener(akú_udalosť, reakcia_na_udalosť);
```

jediným konkrétnym príkazom je tu `addEventListener()`. Ostatné slová sú zástupcami symbolov pre objekty a parametre.

Výraz ***kde_počúvať***, v ktorom dochádza k udalosti (obvykle je to tlačidlo), ***akú_udalosť*** je určitý druh udalosti (napr. kliknutie myšou) a ***reakcia_na_udalosť*** je názov funkcie, ktorá sa spustí, ak dôjde k danej udalosti.

Čiže, môžeme tento príkaz vysvetliť trochu názornejšie. Pokiaľ by sme chceli počúvať udalosť pomocou kliknutia myšou na tlačidlo s názvom „zastav“ a výsledná reakcia by malo byť spustenie funkcie s názvom „stopka“, výsledný kód by vyzeral takto:

```
zastav.addEventListener(MouseEvent.CLICK, stopka);
```

ďalším krokom je vytvorenie funkcie, ktorá bude reagovať na udalosť – v tomto prípade funkcia názov „stopka“. Funkcia jednoducho zoskupuje príkaz, alebo viac príkazov – akcie, ktoré sa tak spustia. Funkcia s názvom „stopka“ bude obsahovať jeden príkaz, ktorého úlohou bude niečo zastaviť. Naša funkcia by mohla vyzerat' takto:

```
function stopka (myEvent:MouseEvent) { };
```

názvy funkcie sú podobne ako názvy tlačidiel ľubovoľné. Môžeme ich pomenovať akýmkoľvek názvom. V našom prípade je to „stopka“. Táto funkcia vo vnútri okrúhlych zátvoriek obsahuje jeden parameter. Jeho názov je *myEvent*. Ide o udalosť, ktorej je priradená udalosť myši, čiže *myEvent:MouseEvent*. Dvojbodka popisuje, o aký typ objektu sa jedná. Ak dôjde k spustení funkcie, vykonajú sa všetky akcie, ktoré sú v zložených zátvorkách.

UDÁLOSTI MYŠI

Nasledujúci zoznam obsahuje kódy jazyka ActionScript 3 pre bežne používané udalosti myši. Je ich možné použiť pri tvorbe vlastných poslucháčov. Pri ich používaní je dôležité, aby sa napísali správne, t.j. používaj aj veľké aj malé písmená, tam, kde je to potrebné.

- MouseEvent.CLICK
- MouseEvent.Mouse_MOVE
- MouseEvent.Mouse_DOWN
- MouseEvent.Mouse_UP
- MouseEvent.Mouse_OVER
- MouseEvent.Mouse_OUT

Príkazy jazyka ActionScript3 pre navigáciu

Nasledujúci zoznam obsahuje kódy jazyka ActionScript 3 pre bežne používané príkazy navigácie. Môžeme ich použiť pri tvorbe tlačidiel pre zastavenie, spustenie alebo presunutie prehrávacej hlavy na iný snímok. Pri príkazoch *gotoAndStop* a *gotoAndPlay* je nutné v zátvorkách doplniť dodatočné údaje, resp. parametre.

- stop ();
- play ();
- gotoAndStop (cislo_snimky);
- gotoAndPlay (cislo_snimky);
- nextFrame ();
- prevFrame ();

príklad:

Nadefinujte trom obrázkom funkciu tlačidiel s odkazom na inú snímku.



1) po importovaní troch obrázkov na pracovnú plochu ich upravíme tak, aby boli rovnako veľké a medzery medzi nimi boli tiež rovnaké.

2) následne konvertujeme každý z obrázkov pomocou klávesy F8 na symbol Button. Zároveň ako názov symbolu uvedieme pre každé tlačidlo btn01 až btn03.



3) ďalším krokom je napísanie nasledovného kódu, pre každé tlačidlo zvlášť:

```
1 btn01.addEventListener(MouseEvent.CLICK, prepni_dalej2);
2 btn02.addEventListener(MouseEvent.CLICK, prepni_dalej3);
3 btn03.addEventListener(MouseEvent.CLICK, prepni_dalej4);
4
5 function prepni_dalej2 (myEvent:MouseEvent):void
6     {
7         gotoAndStop(2);
8     }
9 function prepni_dalej3 (myEvent:MouseEvent):void
10    {
11        gotoAndStop(3);
12    }
13 function prepni_dalej4 (myEvent:MouseEvent):void
14    {
15        gotoAndStop(4);
16    }
17
18
19
```

as:1
Line 19 of 19, Col 13

Popis kódu:

- v riadku 1 až 3 sme vytvorili pre každé tlačidlo poslucháča udalosti. Počúvať sa bude objekt – tlačidlo s názvom btn01 až btn03. Ako udalosť budeme počúvať, či sa na tlačidlo neklikne. Reakciou na udalosť bude následné spustenie funkcie „prepni_dalej2“ až „prepni_dalej3“.

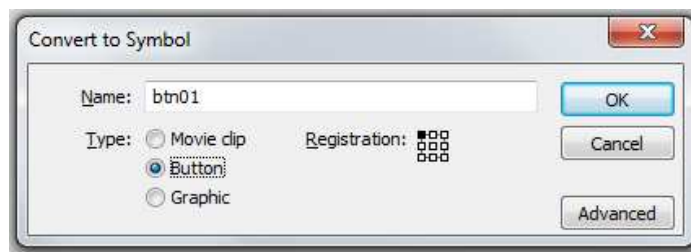
- v riadkoch 5 až 16 sú definované funkcie, pre každé tlačidlo zvlášť. Každá funkcia obsahuje jeden príkaz, ktorý sa vykoná. Ide o presunutie prehrávacej hlavy na príslušnú snímku.

VYTVORENIE ODKAZU NA EXTERNÚ WEBOVÚ STRÁNKU

PRÍKLAD

Vytvorte odkaz, ktorý po kliknutí otvorí nadefinovanú webovú stránku www.markiza.sk

1) vytvoríme tlačidlo pomocou obrázka alebo nakreslíme štvorec/obdĺžnik. Pomocou klávesy F8 ho prekonvertujeme na symbol – Button. Do políčka „Name“ napíšeme btn01, čo bude názov nášho tlačidla.



2) ďalej je potrebné zadať pomenovanie pre inštanciu tlačidla. Ako názov inštancie použijeme opäť názov btn01.



3) pre oživenie funkcie tlačidla napíšeme tento skript:

```
1 btn01.addEventListener(MouseEvent.CLICK, otvor_stranku);
2
3 function otvor_stranku (myEvent: MouseEvent): void
4     {
5
6         var nasa_stranka:URLRequest = new URLRequest ('http://www.markiza.sk');
7         navigateToURL (nasa_stranka, '_blank');
8
9     }
10
```