

Kurz LSL skriptování

Shiny Iceberg

2009



součást cyklu
Nejsme jelita

Kurz LSL skriptování

Shiny Iceberg

v Second Life od roku 2006
shiny.iceberg@virtualmagazine.cz



Aktuální projekty

virtualmagazine.cz
Urbanica, Shinyland
Bwindi Orphans
cyklus Nejsme jelita

Organizační body

Průběh lekce

- bude trvat zhruba 90-120 minut
- pokud nekladete dotazy, vypněte si mikrofon
- příklady si klidně zkoušejte přímo v hledišti

Vaše otázky

- můžete se ptát na konci každého snímku nebo na konci celé přednášky
- dotazy mohou být přes voice nebo IM
- při psaní IM z posledních řad použijte Shout

Plán přednášky

1. Fungování a struktura skriptu
2. Z čeho se skládá skript
3. Vlastnosti objektu
4. Pohyb objektu
5. Pose bally
6. Komunikace skriptu
7. Inventory objektu
8. Detektory
9. Particles
10. Příklady a dokumentace

Inventory objektu

Inventory může obsahovat položky typu

- objekt
- textura
- animace
- oblečení
- skript
- zvuk
- část těla
- gesto
- landmark
- snapshot
- notecard
- volací karta

Oproti inventory uživatele nemůže objekt obsahovat složky a položky s duplicitním názvem, při přetažení složky z inventory uživatele dojde k

- zkopírování jednotlivých položek
- přejmenování duplicitních položek
- deaktivování skriptů

Co nelze udělat skriptem

Některé činnosti spojené s inventory nelze vykonat pomocí skriptu, např:

- jakkoliv manipulovat s inventory uživatele kromě vložení nových položek
- vytvořit či přejmenovat položky v inventory primu
- zkopírovat do inventory rezznutý prim
- nakopírovat sám sebe do jiného primu
- číst inventory jiného primu
- attachnout na avatara položku z inventory uživatele

Práce s inventory

- `IIAllowInventoryDrop`
- `IIGetInventoryNumber`
- `IIGetInventoryName`
- `IIGiveInventory`
- `IIGiveInventoryList`
- `IIGetScriptName`
- `IIRemoveInventory`
- *event changed*

11AllowInventoryDrop (integer *AnoNe*) ;

VSTUP: TRUE/FALSE - zda povolit ostatním vhazování obsahu do objektu

VÝSTUP: nic

POZNÁMKA:

- položky se do objektu vhazují pomocí Ctrl + drag&drop


```
integer llGetInventoryNumber  
    (integer typ) ;
```

VSTUP: typ položek, které chceme spočítat

VÝSTUP: počet položek daného typu v inventory

POZNÁMKA: typ položek se zadává konstantou

- INVENTORY_ALL
- INVENTORY_TEXTURE
- INVENTORY_SOUND
- INVENTORY_LANDMARK
- INVENTORY_CLOTHING
- INVENTORY_OBJECT
- INVENTORY_NOTECARD
- INVENTORY_SCRIPT
- INVENTORY_BODYPART
- INVENTORY_ANIMATION
- INVENTORY_GESTURE

```
string l1GetInventoryName (integer typ,  
integer poradi) ;
```

VSTUP:

- *typ*: druh vybrané položky
- *poradi*: pořadové číslo vybrané položky

VÝSTUP: jméno vybrané položky

POZNÁMKA:

- položky jsou z hlediska skriptu vždy řazený abecedně

```
llGiveInventory(key komu, string co);
```

VSTUP:

- **komu:** UUID příjemce - avatara nebo objektu
- **co:** jméno položky, kterou chceme darovat

VÝSTUP: nic

POZNÁMKA:

- pokud je příjemce objekt, musí mít povoleno *AllowInventoryDrop*
- pokud je příjemce avatar, dojde k pozastavení skriptu na 3 sekundy

Příklad k `llGiveInventory`

```
default
{
    touch_start(integer cislo)
    {
        // priklad, který daruje notecard
        // detekce avatara, který kliknul na objekt
        key Avatar = llDetectedKey(0);
        string Info = llGetInventoryName(INVENTORY_NOTECARD, 0);
        llGiveInventory(Avatar, Info);

        // zkraceny priklad, který daruje landmark
        llGiveInventory(llDetectedKey(0), llGetInventoryName(INVENTORY_LANDMARK, 0));
    }
}
```

```
llGiveInventoryList(key komu,  
string slozka, list polozky);
```

VSTUP:

- **komu:** UUID příjemce - avatara nebo objektu
- **slozka:** název složky, do které se položky vloží
- **polozky:** seznam názvů položek, které se mají darovat najednou

VÝSTUP: nic

POZNÁMKA:

- jméno složky je u darování objektu ignorováno

Příklad k *llGiveInventoryList*

```
// Příklad vyžaduje objekt, který má v sobě nějaký landmark,  
// nějakou notecard a položku s názvem Moje textura  
  
default  
{  
    touch_start(integer cislo)  
    {  
        key Avatar = llDetectedKey(0);  
        string Polozka1 = llGetInventoryName(INVENTORY_NOTECARD, 0);  
        string Polozka2 = llGetInventoryName(INVENTORY_LANDMARK, 0);  
        list Seznam = [Polozka1, Polozka2, "Moje textura"];  
        llGiveInventoryList(Avatar, "Nejsme jelita TEST", Seznam);  
    }  
}
```

```
string l1GetScriptName ();
```

VSTUP: nic

VÝSTUP: jméno skriptu, ve kterém se příkaz volá

POZNÁMKA:

- tato funkce se často používá pro vydání celého inventory objektu kromě daného skriptu

`llRemoveInventory (string co) ;`

VSTUP: jméno položky inventory ke smazání

VÝSTUP: nic

POZNÁMKA:

- smazaná položka přestane existovat, tj. nevrátí se vlastníkovi objektu mezi smazané položky (Trash)

```
// zrusi text nad objektem
default
{
    state_entry()
    {
        llSetText("", <1.0, 1.0, 1.0>, 1.0);
        llRemoveInventory( llGetScriptName() );
    }
}
```


changed (integer *typZmeny*) ;

- event se aktivuje v případě změny některých parametrů primu - jeden z nich je změna inventory, tedy *CHANGED_INVENTORY*, často se používá v kombinaci s *CHANGED_ALLOWED_DROP*
- změna inventory může být způsobena navíc i
 - přejmenováním položky inventory
 - změna práv položky inventory
 - rekompilací skriptu (ale ne reset skriptu)
 - úprava a uložení stávající notecardy
 - změna inventory jiného primu v rámci primsetu

Více viz <http://www.lslwiki.net/lslwiki/wakka.php?wakka=changed>

Příklad : Skupinový výdejní automat

```
default
{
    touch_start(integer cislo)
    {
        key Avatar = llDetectedKey(0);
        if( llSameGroup(Avatar) )
        {
            integer pocetPolozek = llGetInventoryNumber(INVENTORY_ALL);
            string NazevObjektu = llGetObject();

            integer Index;
            string Jmeno;
            list Inventory;

            for (Index = 0; Index < pocetPolozek; Index = Index + 1)
            {
                Jmeno = llGetInventoryName(INVENTORY_ALL, Index);
                Inventory = Inventory + [Jmeno];
            }

            Index = llListFindList(Inventory, [llGetScriptName()]);
            Inventory = llDeleteSubList(Inventory, Index, Index);

            llGiveInventoryList(Avatar, NazevObjektu, Inventory);
            llInstantMessage(Avatar, "Doruceno do inventory do složky >> " + NazevObjektu + " <<");
        }
        else
        {
            llInstantMessage(Avatar, "Zboží je pouze pro členy skupiny.");
        }
    }
}
```

Příklad : Schránka na vzkazy 1/3

```
default
{
    state_entry()
    {
        llAllowInventoryDrop(TRUE);
    }

    changed(integer typZmeny)
    {
        if(typZmeny & (CHANGED_ALLOWED_DROP | CHANGED_INVENTORY))
        {
            llSleep(2.0);
            integer pocetNote = llGetInventoryNumber(INVENTORY_NOTECARD);

            if(pocetNote == 1)
            {
                string notecard = llGetInventoryName(INVENTORY_NOTECARD, 0);
                llGiveInventory(llGetOwner(), notecard);
                llInstantMessage(llGetOwner(), "Prišla ti zpráva: " + notecard);
                llSay(0, "Vaše zpráva " + notecard + " byla odeslána. Dekuji");
                llRemoveInventory(notecard);
            }
        }
    }
}
```

Příklad : Schránka na vzkazy 2/3

```
else
{
    list inventory;
    integer pocetInvi = llGetInventoryNumber(INVENTORY_ALL);
    integer pocitadlo = 0;

    while(pocitadlo < pocetInvi)
    {
        inventory = inventory + [llGetInventoryName(INVENTORY_ALL, pocitadlo)];
        pocitadlo = pocitadlo + 1;
    }

    // nasledujici tri radky kodu
    // odstani ze seznamu inventory jmeno tohoto skriptu
    list tentoSkript = [llGetScriptName()];
    integer index = llListFindList(inventory, tentoSkript);
    inventory = llDeleteSubList(inventory, index, index);
    pocetInvi = llGetListLength(inventory);
}
```

Příklad : Schránka na vzkazy 3/3

```
if (pocetInvi > 0)
{
    llSay(0, "Vložte pouze jednu zprávu zároveň ve forme notecard.");
    llSay(0, "Odstranuji vložený obsah, aniž by byl odeslán.");
}

while (pocetInvi > 0)
{
    llRemoveInventory(llList2String(inventory, 0));
    inventory = llDeleteSubList(inventory, 0, 0);
    pocetInvi = llGetListLength(inventory);
}
}
}
```

Otázky a diskuze

