Vývoj geografických aplikácií v GIS

Hana Stanková

Lekcia 7

Tvorba vlastného nástroja

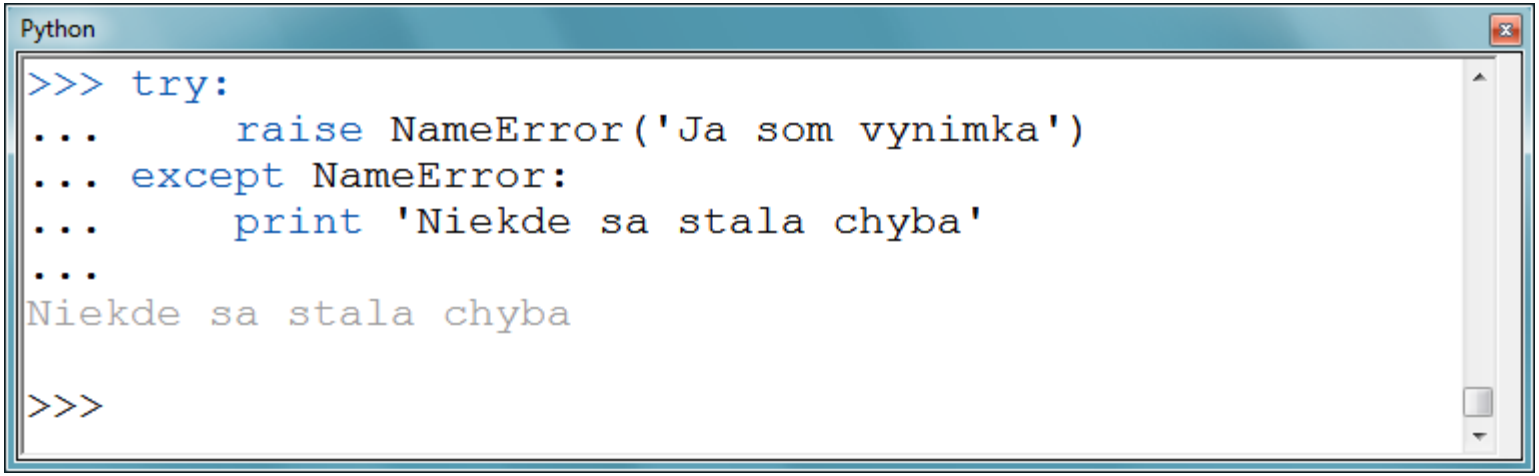
Čistenie kódu

- príkazy **try**, **except**, **finally**
- používajú sa, ak chceme zachytiť výnimky v programe (napr. vstupná trieda objektov nemá žiadne objekty)
- najprv sa vykonajú príkazy za kľúčovým slovom **try:**
- ak sa neobjaví výnimka, dokončí sa vykonanie príkazov
- ak sa objaví výnimka, zvyšok príkazov sa preskočí a výnimka sa porovnáva s výnimkami za kľúčovým slovom **except:** (môže ich byť aj viac)

Čistenie kódu

- ak sa výnimka zhoduje, vykonajú sa príkazy za except a pokračuje sa ďalej za príkazom try
- ak sa výnimka nezhoduje (nezachytí sa), pokračuje sa vo vykonávaní vonkajšieho príkazu try alebo sa program zastaví
- za kľúčové slovo **finally:** môžeme umiestniť príkazy, ktoré chceme aby sa vykonali bez ohľadu na to, či sa výnimka objaví alebo nie (napr. vymazať kurzor)

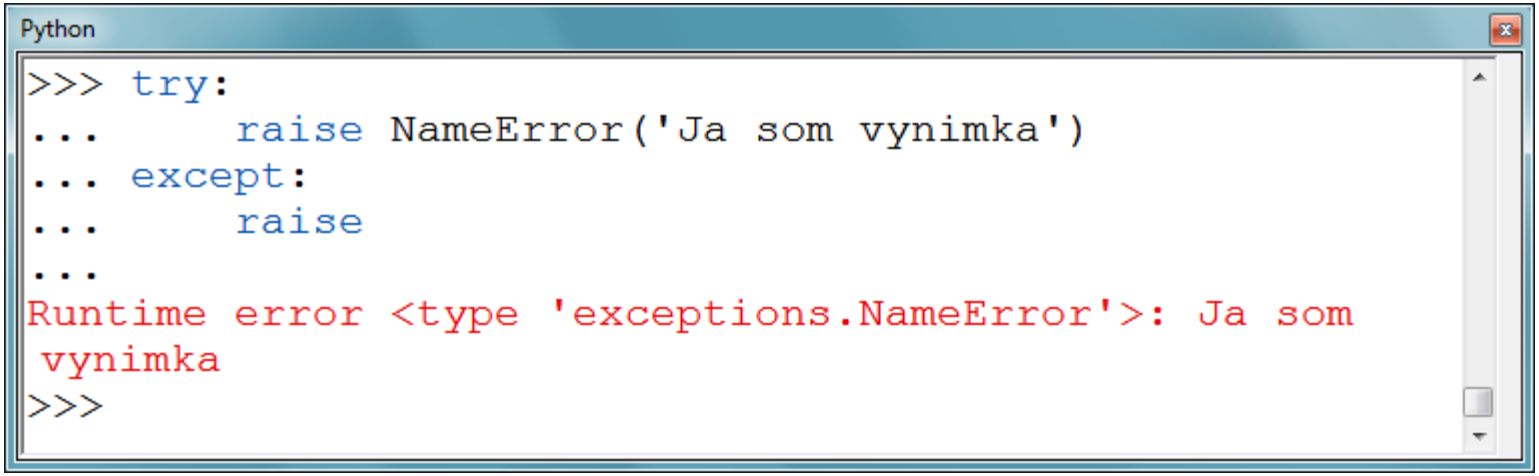
Čistenie kódu



Čistenie kódu

- kľúčové slovo **except:** zachytáva výnimky až po najvyššiu úroveň (systémové)
- príkaz **raise** umožňuje tvorbu vlastných výnimiek (musia byť inštanciami alebo podtriedami triedy Exception)
- ak sa zadá príkaz **raise** bez argumentov, znovu sa získa výnimka, ktorá je práve aktívna, a vypíše sa chyba (bez tohto by sa len vykonali príkazy za except)

Čistenie kódu



Chyby a správy

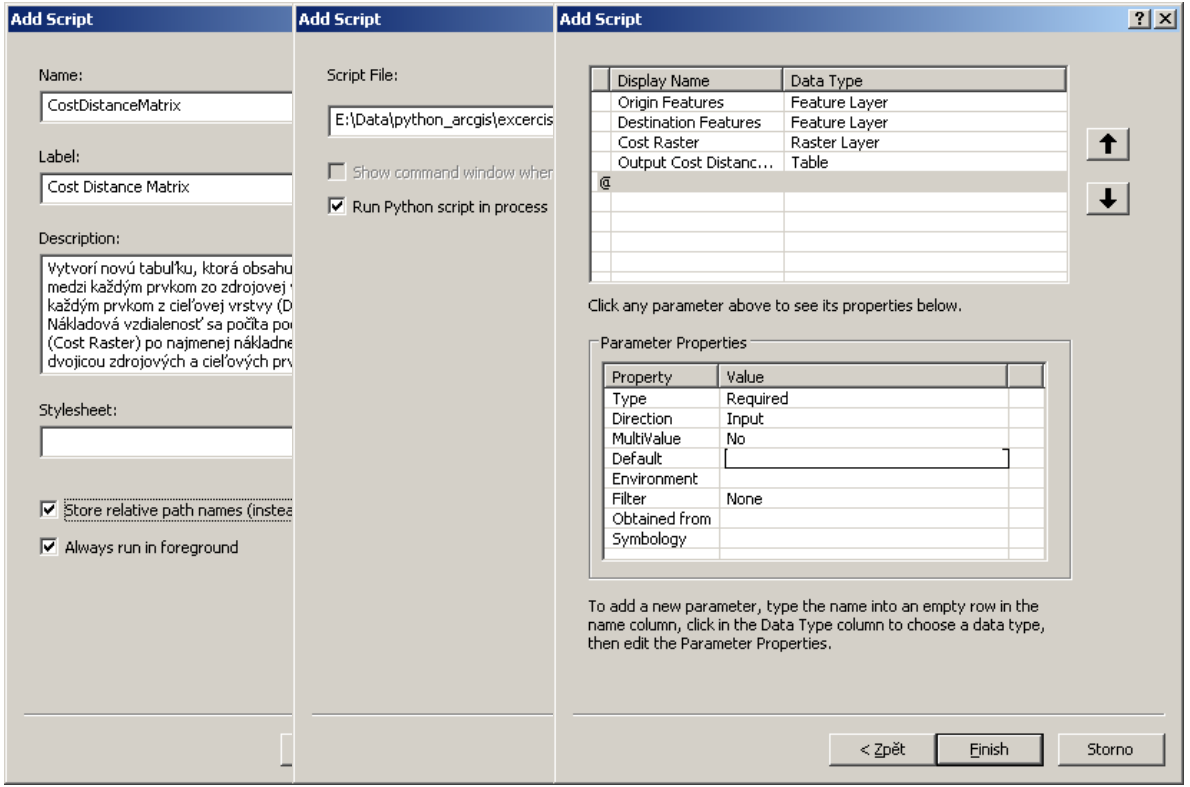
- na zachytávanie správ pri spúšťaní nástroja slúži funkcia **arcpy.GetMessages()**
- na zobrazenie chýb a správ používateľovi slúžia funkcie:
- **arcpy.AddMessage()** - všeobecné a informatívne správy (závažnosť 0)
- **arcpy.AddWarning()** - varovné správy (závažnosť 1)
- **arcpy.AddError()** - chybové hlásenia (závažnosť 2)

Chyby a správy

- výpis správ v interaktívnom okne:
  print arcpy.GetMessages()
- funguje to len v Python okne mimo ArcGIS

Tvorba nového nástroja

- ArcCatalog
- New - Toolbox
- Add - Script



Nastavovanie parametrov

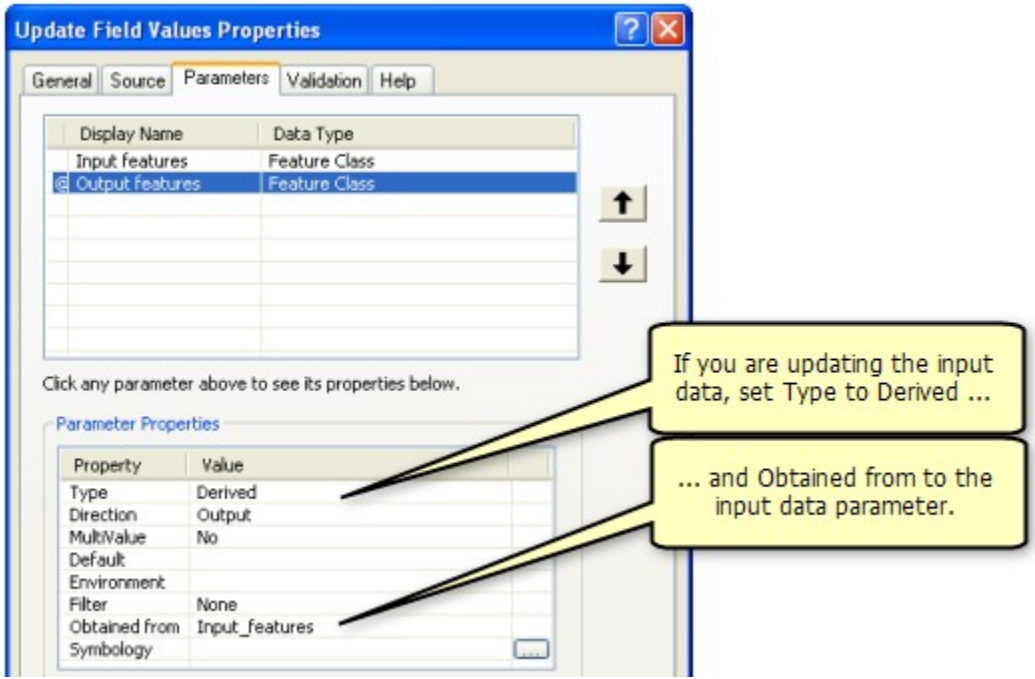
- základné vlastnosti:
- **Display Name**
- **Data Type** - Feature Class, Shapefile, File, Text File, String, Table, Raster ...
- ďalšie vlastnosti:
- **Type** - Required, Optional, Derived (len výstupné)
- **Direction** - Input, Output
- **MultiValue** - Y (zoznam hodnôt), N

Nastavovanie parametrov

- ďalšie vlastnosti:
- **Default** - prednastavená hodnota
- **Filter** - 6 typov filtrov
- **Obtained from** - pre vstupné a odvodené výstupné parametre
- **Symbology** - symbolika (.lyr) pre zobrazovanie výstupov

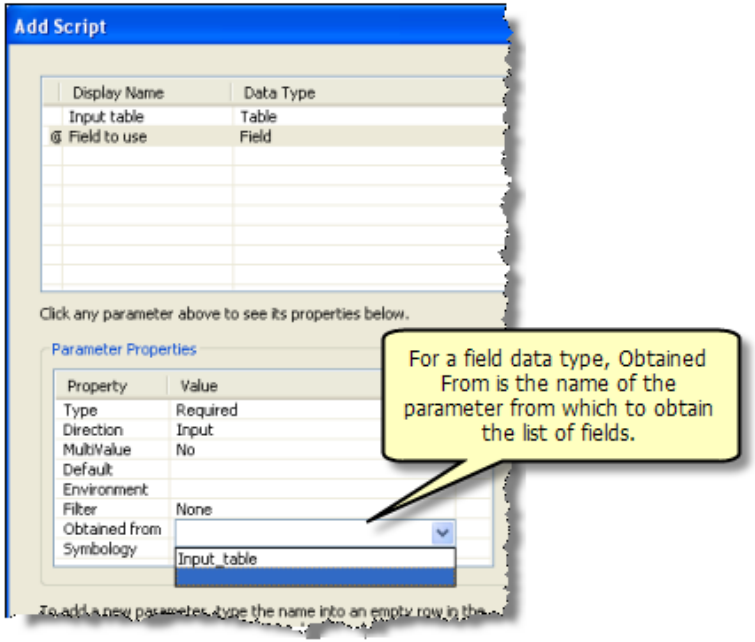
Odvodené parametre

- keď chceme aktualizovať vstupné údaje:



Získané parametre

- keď chceme vybrať atribútové pole zo vstupnej vrstvy:



Filtrovanie parametrov

- typy filtrov:
- **Value List** - zoznam predvolených hodnôt (číselných alebo textových)
- **Range** - povolený rozsah hodnôt
- **Feature Class** - zoznam povolených typov (Point, ...)
- **File** - zoznam povolených koncoviek súborov (.txt, ...)
- **Field** - zoznam povolených typov atribútov
- **Workspace** - zoznam povolených typov prostredí

Validácia parametrov

- kontroluje, či sú všetky parametre zadané korektne, a ak nie, podá o tom nejakým spôsobom správu
- vykonáva sa pomocou bloku Python kódu
- dva druhy validácie:

  1. interná (základná) validácia - ArcGIS ju vykonáva automaticky

  2. vlastná validácia - môžeme vytvoriť vlastný kód

Interná validácia

- kontroluje:
- či sú zadané povinné parametre
- či je zadaná hodnota správneho typu
- či zadané hodnoty zodpovedajú filtru
- či existujú vstupné vrstvy
- či existujú výstupné vrstvy (podľa nastavenia overwriteOutput)
- generuje predvolenú cestu pre výstupné vrstvy

Vlastná validácia

- pomocou vlastnej validácie môžeme:
- aktualizovať hodnoty filtrov na základe iných parametrov (napr. vstupnej vrstvy)
- aktivovať/deaktivovať parametre
- počítať predvolené hodnoty
- ... a ďalšie

Vlastná validácia

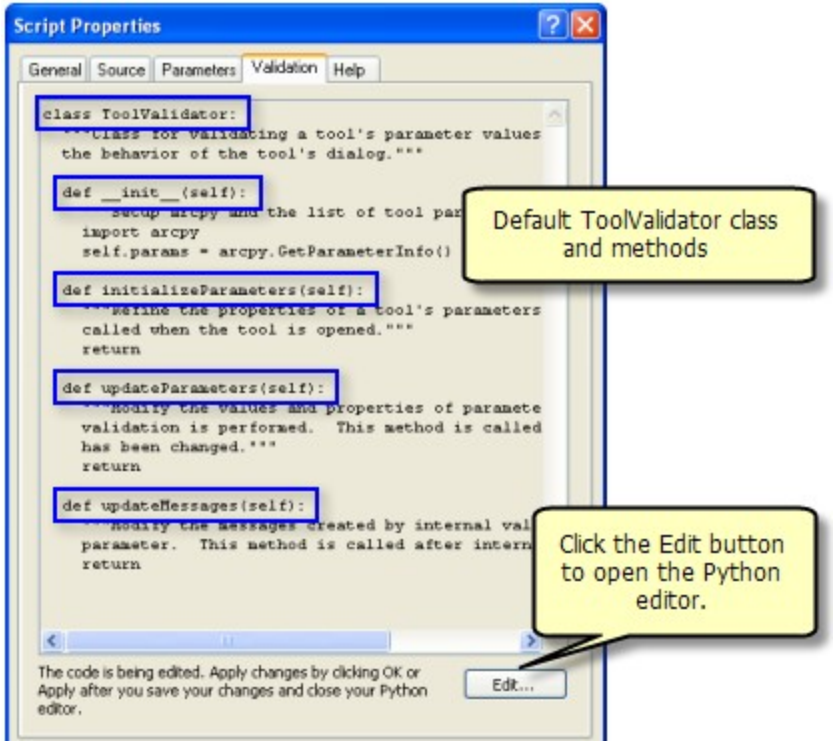
- **Script Properties - Validation**
- trieda **ToolValidator** obsahuje tieto metódy:
- **__init__** - inicializuje triedu, importuje knižnice, inicializuje objekt (self)
- **initializeParameters** - volá sa, keď sa prvý krát otvorí dialógové okno (alebo sa nástroj spustí z prík. riadku)
- **update Parameters** - volá sa pri každej zmene parametra, následne sa vykoná interná validácia

Vlastná validácia

- trieda **ToolValidator** obsahuje tieto metódy:
- **updateMessages** - volá sa po skončení internej validácie

- metódy môžeme editovať cez Python editor
- na začiatku obsahujú iba príkaz return
- musia byť všetky tri prítomné, aby bol trieda ToolValidator validná

Vlastná validácia



Vlastná validácia

- príklad - aktivovanie/deaktivovanie parametrov:

```
def updateParameters(self):

  # If the option to use a weights file is selected (the user chose
  #  "Get Spatial Weights From File"), enable the parameter for specifying
  #  the file, otherwise disable it
  #
  if self.params[3].value == "Get Spatial Weights From File":
    self.params[8].enabled = 1
  else:
    self.params[8].enabled = 0
```

Preberanie parametrov

- dva spôsoby preberania parametrov:
  1. funkcia **arcpy.GetParametersAsText()**
  2. metóda **sys.argv** - limitovaný počet znakov

  sys.argv[0] - názov súboru so skriptom

```
# Read the parameter values:
#  1: input workspace
#  2: input clip features
#  3: output workspace
#
inWorkspace   = arcpy.GetParameterAsText(0)
clipFeatures  = arcpy.GetParameterAsText(1)
outWorkspace  = arcpy.GetParameterAsText(2)
env.workspace = inWorkspace
```

```
# Read the parameter values:
#  1: input workspace
#  2: input clip features
#  3: output workspace
#
inWorkspace   = sys.argv[1]
clipFeatures  = sys.argv[2]
outWorkspace  = sys.argv[3]
env.workspace = inWorkspace
```

Dokumentácia nástroja

- Tool - Item Description