

Informatika pro moderní fyziky (10) konfigurační soubory YML, formát JSON, použití cizích API, mash-up aplikace

František HAVLŮJ

e-mail: haf@ujv.cz

ÚJV Řež

oddělení Reaktorové fyziky a podpory palivového cyklu

akademický rok 2019/2020

27. listopadu 2019

- 1 Tvorba obrázků
- 2 Persistence dat a formát JSON
- 3 Použití cizích API
- 4 Interaktivní mapa

Obsah

- 1 Tvorba obrázků
- 2 Persistence dat a formát JSON
- 3 Použití cizích API
- 4 Interaktivní mapa

Dokončení z minula

- v posledním souboru jsou navíc neznámé typy XX a YY – vhodně vyřešte:
- a) náhodná barva `rand, rand(123)`
- b) seznam barev pro neznámé typy
- vylepšení: načítáme barvy ze souboru! mohlo by to vypadat takto: (ale použijeme YAML - viz dále)

```
F4  ff0000
R1  00ffff
R2  00eeee
```

YAML - kamarád pro konfigurační soubory

- už je trochu nuda pořád ručně načítat soubory a parsovat je, normálně se to tak nedělá – použiju standardizovaný formát souboru
- na konfigurační soubory je skvělý formát YAML (YML) – jednoduchý hash formátovaný odsazením
- načtu standardní knihovnu `require "yaml"`
- parsování řetězce
`YAML.load(File.read("config.yml"))`
- uložení dat `File.write("config.yml", data.to_yaml)`
- s hvězdičkou: `RRGGBB`, `#RRGGBB` i `R,G,B`

```
colors:  
  F4: ff0000  
  R1: 00ffff
```

Obsah

- 1 Tvorba obrázků
- 2 Persistence dat a formát JSON**
- 3 Použití cizích API
- 4 Interaktivní mapa

Ukládání strukturovaných dat

- často mám data ve formě struktury (kombinace hash+pole, různý stupeň vnoření)
- z různých důvodů můžu chtít data uložit na disk a pak je znovu načítat
- (zejména efektivita zpracování, případně data z externích/webových zdrojů)
- bylo by dobré mít možnost uložit a načíst rovnou celý hash
- odpověď jsou strukturované metaformáty - YAML, XML, JSON

Práce s JSON

- v Ruby je k mání knihovna – `require "json"`
- generování JSON: `hash.to_json`
- čtení JSON: `JSON[data]`

Příklad – výsledky běhu kolem rybníka

- dva soubory – `ages.csv`, `times.csv`
- chci v jednom skriptu (tasku) načíst, spárovat a uložit
- a v jiném už rovnou načíst zpracovaná data
- a vypsát tabulku výsledků včetně ročníků narození
- pozor! JSON nezná symboly, uloží se jako řetězce

Obsah

- 1 Tvorba obrázků
- 2 Persistence dat a formát JSON
- 3 Použití cizích API**
- 4 Interaktivní mapa

K čemu to?

- spousta informací na webu je poskytována ve strojově čitelné formě
- API – rozhraní mezi aplikacemi
- s využitím webových služeb naše možnosti exponenciálně rostou (počasí, doprava, mapy, atd atd.)
- spousta věcí se dá udělat jako *mashup* – sice nic neumím, ale umím to dát dohromady

Typy / formáty

- URL – rovnou dostanu např. obrázek po zadání správného URL
- XML – velmi obecný, ale komplikovaný formát (“vypadá jako HTML”)
- JSON – velmi jednoduchý a kompaktní formát, vyvinutý pro JS (v podstatě jen číslo, řetězec, pole, hash)

URL API – google maps

- stačí správně vymyslet
- pozor na usage limits (v produkci je nutné lokální cache...)
- QR platba:

`http://qr-platba.cz/pro-vyvojare/restful-api/#generator-czech-image`

- Google Maps static API:

`https://developers.google.com/maps/documentation/staticmaps/`

Jednoduchý mashup: mapa o-závodů

- ORIS API – <http://oris.orientacnisporty.cz/API>
- úkol: vypíšme kalendář MTBO závodů v roce 2017
- <http://oris.orientacnisporty.cz/API/?format=json&method=getEventList&sport=3&datefrom=2017-01-01&dateto=2017-12-31>
- <https://developers.google.com/maps/documentation/static-maps/intro>
- API klíč – v praxi si musíte pořídit vlastní (ale nic to nestojí)
- AlzaSyC6_NpMTN-8oIOHWzaiwjeZ8eu_J3XI1IM

Obsah

- 1 Tvorba obrázků
- 2 Persistence dat a formát JSON
- 3 Použití cizích API
- 4 Interaktivní mapa**

Jednoduchý mashup: mapa o-závodů

- seznam závodů máme z předchozího kroku, teď bychom jenom chtěli, aby byla mapa pro změnu od seznamu a klikací
- `http://oris.orientacnisporty.cz/API/?format=json&method=getEventList&sport=3&datefrom=2017-01-01&dateto=2017-12-31`
- `http://api.mapy.cz`

Aspoň něco bychom si k tomu říct měli

- HTML: jazyk (založený na XML, takže nám trochu připomene SVG)
- JS: jazyk povětšinou běžící v prohlížeči, umožňující client-side interaktivitu (vypadá jako něco mezi C++ a Ruby)
- největší radost udělají spolu

Klasický postup: upravovat vzor

Většinou máme k dispozici nějaký příklad, ze kterého se dá vyjít – a to často i v situaci, kdy o tématu nevíme vůbec nic

- `http://api.mapy.cz/view?page=instruction`
- `http://api.mapy.cz/view?page=markerlayer`

A to je vše, přátelé!

