

# Používateľská príručka

Pre spustenie aplikácie sú potrebné tri JavaScriptové súbory:

- `parser.js`
- `earley-oop.js`
- `tokenizer.js`

Aplikácia zároveň pre svoje fungovanie potrebuje gramatiku, slovník a konfiguráciu pre jazyk, ktorý parsuje, tieto súbory by sa mali nachádzať vo svojich osobitných podpriechinkoch v priečinku `res`:



Na vytvorenie sparsovaného XML z nejakého textu, je treba v súbore `parser.js` vytvoriť inštanciu triedy `Parser`, ktorá dostáva dva parametre. Prvým z nich je text, ktorý sa bude parsovať a druhým je jazyk vo forme reťazca napríklad "SK" alebo "EN".

Ďalej sa na túto inštanciu `Parseru` zavolajú funkcie `buildXML()` a `stringifyTree()` prvá z nich doplní do stromu všetky tagy a druhá vráti hotové XML.

```
1 var p = new Parser("- Tak čo je nové? - spytuje sa zvedavo mama. -  
    Aj sama to vieš, - odpovedá dcéra.", "SK");  
2 p.buildXML();  
3 p.stringifyTree()
```

---

Trieda `Parser` umožňuje aj vytvorenie HTML a JSON reprezentácie stromu pomocou funkcií `getHTMLTree()` a `getJSONTree()`, ktoré ich vrátia ako string, prípadne pomocou funkcií `saveHTMLTree(<path>)` a `saveJSONTree(<path>)`, ktoré ich priamo uložia do súboru.

V prípade, že by bolo potrebné upraviť gramatiku - napríklad pridať pravidlo, tak stačí v priečinku `grammars` v niektorom súbore s gramatikou vytvoriť nový atribút JSON-u vo forme:

```
1 {  
2     ...,  
3     "LavaStranaPravidla": "PrvaMoznost | DruhaMoznost | ...",  
4     ...,  
5 }
```

---

Alternatívy, ktoré môžu vzniknúť sa oddeľujú pomocou `|` (zvislá čiara).

V gramatike je možné(a aj nutné) použiť tokeny z prislúchajúceho slovníku v priečinku `terminalRules`, ktoré sú vo forme:

```
1 {  
2     ...,  
3     "RegexKtoryMatchujeToken": "TypTokenu",  
4     "[0-9]+$": "Num",  
5     ...,  
6 }
```

---

Úpravy gramatiky sa väčšinou nepodaria na prvý pokus a preto je vhodné využívať parserom vygenerovaný HTML strom na debugovanie, prípadne si vypísať stromy alebo hotové XML do konzoly:

```
1 var p = new Parser("Toto je vstupný text", "SK");  
2 p.buildXML();  
3 p.saveHTMLTree("../app/res/trees/tree.html"); // subor otvoriť vo  
    webovom prehliadaci  
4 p.printTrees(); // vypise stromy do konzoly  
5 console.log(p.stringifyTree()); // vypise XML do konzoly
```

---