

pttypter

Uživatelská příručka verze 1.1.0

Marek Smolík (62880168)
11. prosince 2022

1 Úvod

pttypter je program pro příkazový řádek, který je schopen vzít vstup v podobě syntaktické struktury JSON a následně vrátit tento kód řádně naformátovaný a barevně zvýrazněný. To je schopen provést přímo v příkazovém řádku, nebo jakožto kombinaci HTML a CSS kódu.

2 Platformová přístupnost

Program je oficiálně distribuován pro platformy Windows a Linux (x64_64), avšak v případě Windowsu je minimální podporovaná verze 1909 (ve starších by bylo nutno aktivovat podporu ANSI kódů).

3 Spouštění

Jak již bylo řečeno, **pttypter** je designovaný primárně pro příkazový řádek. Buď **pttypter** v následujících příkladech cesta k binárnímu programu **pttypter**. Nápovědu k programu lze vypsát příkazem **pttypter -h** nebo **pttypter --help**.

```
1 $ pttypter -h
2
3 pttypter
4 Marek Smolik, 2022
5 Simple CLI app to pretty-print JSON
6
7 USAGE:
8   pttypter [OPTIONS]
9
10 FLAGS:
11   -h, --help          Prints help information
12
13 OPTIONS:
14   --input  STRING      The input JSON code
15   --theme  STRING      dark/light [DEFAULT: dark]
16   --output STRING      html/term [DEFAULT: term]
17
```

Obrázek 1: Výstup po invokaci programu s parametrem **-h** nebo **--h**

Hlavním parametrem je **--input**, který slouží k zadávání samotného JSONu. Parametr **--theme** slouží k nastavení barevného schéma. Jediné možné hodnoty jsou **dark** a **light**. Každá z těchto variant koresponduje k příslušné variaci barevného režimu inspirovaného stylem *Gruvbox*, původně vytvořeným Pavlem Percevem pro editor vim. Výchozí hodnota je **dark**, tedy tmavá.

```

^ ptttypter --input "{ \"ahoj\": \"světe\"}"
{
  "ahoj": "světe"
}

```

Obrázek 2: Ukázka invokace programu se zadáním kódu s výstupem v terminálu

Jak lze vidět, program ignoruje všechny whitespace znaky a kód řádně naformátuje a zabarví. Jediným problémem je, že speciální znaky jako " je nutno označit zpětným lomítkem \, avšak to je spíše záležitost shellu, kterým je program invokován.

Pokud používaný shell disponuje funkcionalitou rour, lze jich využít.

```

^ cat test.json
{
  "hello": "world"
}
target/x86_64-unknown-linux-gnu/debug on ↵ main [!?]
^ < test.json | ptttypter
{
  "hello": "world"
}

```

Obrázek 3: Použití rour k zadávání vstupního JSON kódu

Alternativní, možností výstupu je HTML + CSS kód. Ten lze získat například následovně:

```

1 $ ptttypter --input '{ "je_to_super": True, "problem": null}' --output html --theme light
2
3 <style>
4   .__ptttypter_pre {
5     font-family: monospace;
6     font-size: inherit;
7     display: block;
8     background: none;
9     white-space: pre;
10    -webkit-overflow-scrolling: touch;
11    overflow-x: scroll;
12    max-width: 100%;
13    min-width: 100px;
14    padding: 0;
15  }
16
17
18  .__ptttypter_pre {
19    background-color: rgb(251, 241, 199);
20    color: rgb(60, 56, 54);
21  }
22
23  .__ptttypter__fg {

```

```

24     color: rgb(60, 56, 54);
25 }
26
27 .__pttypter__bg {
28     color: rgb(251, 241, 199);
29 }
30
31 .__pttypter__number {
32     color: rgb(177, 98, 134);
33 }
34
35 .__pttypter__string {
36     color: rgb(215, 153, 33);
37 }
38
39 .__pttypter__keyword {
40     color: rgb(204, 35, 29);
41 }
42
43 </style>
44
45 <pre class="__pttypter_pre">
46 {
47 &nbsp;&nbsp;&nbsp;<span class="__pttypter__string">je_to_super</span>: <span class="__
48   __pttypter__keyword">True</span>,
49 &nbsp;&nbsp;&nbsp;<span class="__pttypter__string">problem</span>: <span class="__
50   __pttypter__keyword">null</span>
51 }
52 </pre>

```

Pokud používaný shell podporuje roury, lze následovně výstup také nasměrovat přímo do souboru přípony `html`. Při otevření tohoto souboru (respektive přepokopírování výstupu) ve webovém prohlížeči lze spatřit blok kódu vygenerovaný programem.

```

{
  "je_to_super": True,
  "problem": null
}

```

Obrázek 4: Ukázka vyrenderovaného HTML+CSS kódu vygenerovaného `pttypterem`

```

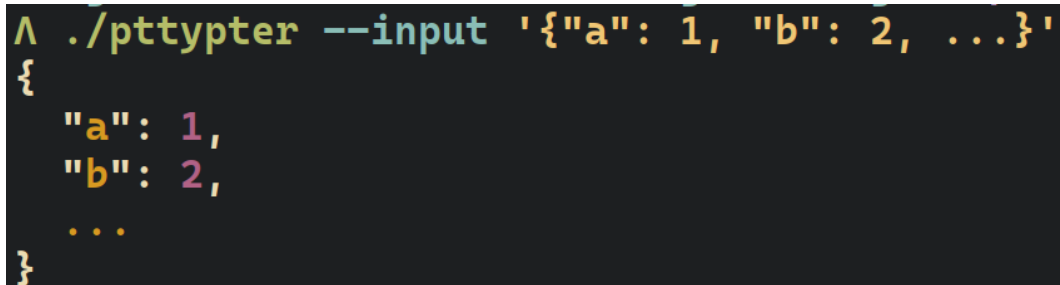
^ cat input.json
{
    "improve": {
        "away": true,
        "difficult": "tape",
        "his": -725025837,
        "central": true,
        "soap": -702298349,
        "supper": -542959661.8619902
    },
    "leader": true,
    "future": true,
    "exact": -566106614.1593692,
    "land": true,
    "mass": true
}%
pttypter/target/release on  main [!?]
^ pttypter < input.json
{
    "improve": {
        "away": true,
        "difficult": "tape",
        "his": -725025837,
        "central": true,
        "soap": -702298349,
        "supper": -542959661.8619902
    },
    "leader": true,
    "future": true,
    "exact": -566106614.1593692,
    "land": true,
    "mass": true
}

```

Obrázek 5: Ukázka vypsání většího JSON souboru

4 Validita vstupního kódu

Program nikdy nevyhodnocuje, zda-li je vstupní kód opravdu validní JSON. Samozřejmě to předpokládá, ale v určitých případech může být záměrně vstupní kód zinvalidněn. V takových případech se program pokusí se vstupem jaksí vypořádat, avšak kvalita výstupu takové procedury není zaručena.



```
Λ ./pttypter --input '{"a": 1, "b": 2, ...}'  
{  
  "a": 1,  
  "b": 2,  
  ...  
}
```

Obrázek 6: Ukázka výstupu programu s invalidním JSON vstupem