再び HTML書き を Web の主役(?)にする

ための、テンプレートエンジン: YATT

X: @hkoba （えちこば）、フリーランス（一応）

仕事：Perl, Tcl/Tk, GCP 上の Linux サーバー管理etc... TypeScriptは勉強中

GitHub: hkoba

ブログ：hkoba.hatenablog.com

2023半ば、宮崎移住（木花地区）

今日の資料 ＞

hkoba.github.io → `Webナイト宮崎` `#21`

デモ環境 ＞

github.com/hkoba/yatt-starter-html

- Dockerfile あります

（...Perl と聞いて、お気づきとは思いますが...）

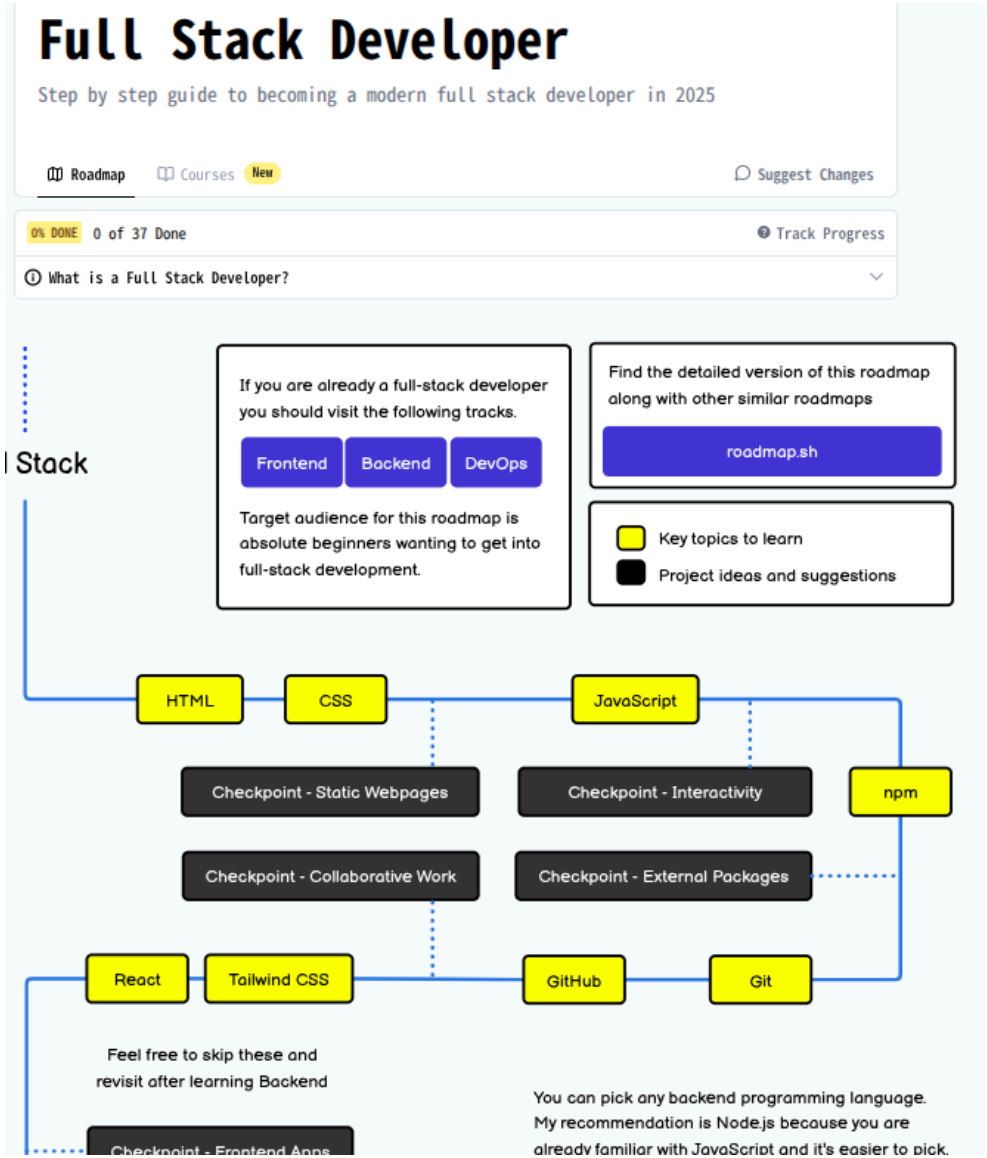
世間のトレンドから、程遠い話です

お許し下さい＞＜

さて本題...

昔は良かった...

- **HTML を書ける人** なら、誰でも
- **画面の原案** くらいは、自力で作れた



今の Web は

参入ハードルを

上げ過ぎでは…？

← https://roadmap.sh/full-stack

勉強すること、多すぎ！

とは言っても、（一定レベル以上の）Webサイトを作るなら

- （静的 or 動的）**テンプレートエンジン** は導入したい
- 理由：HTML だけですら…
    - 見た目の要求水準が上昇 ＝ 記述量が増えた
    - ページごとに書き換える箇所が沢山
        - 手作業だと負荷が大きい・漏れが出る

けど一般的に、

テンプレートエンジンは...

- 構文が...まぁまぁ難しい（HTML らしくない）ものが多い

また、せっかく導入しても

プログラムから、部品として呼ばれる

だけのテンプレートだと

- **HTML 書き**は、不自由な下請けのまま
- **自力で** 改善出来ることが少ない、主役感が無い

そこで、

YATT というテンプレートエンジン

を作っています（サーバーサイド、動的）

- 極力、HTML らしさを保つ
- ファイルをポンと足せば動く
- 入力を間違ったら、すぐに lint が知らせてくれる

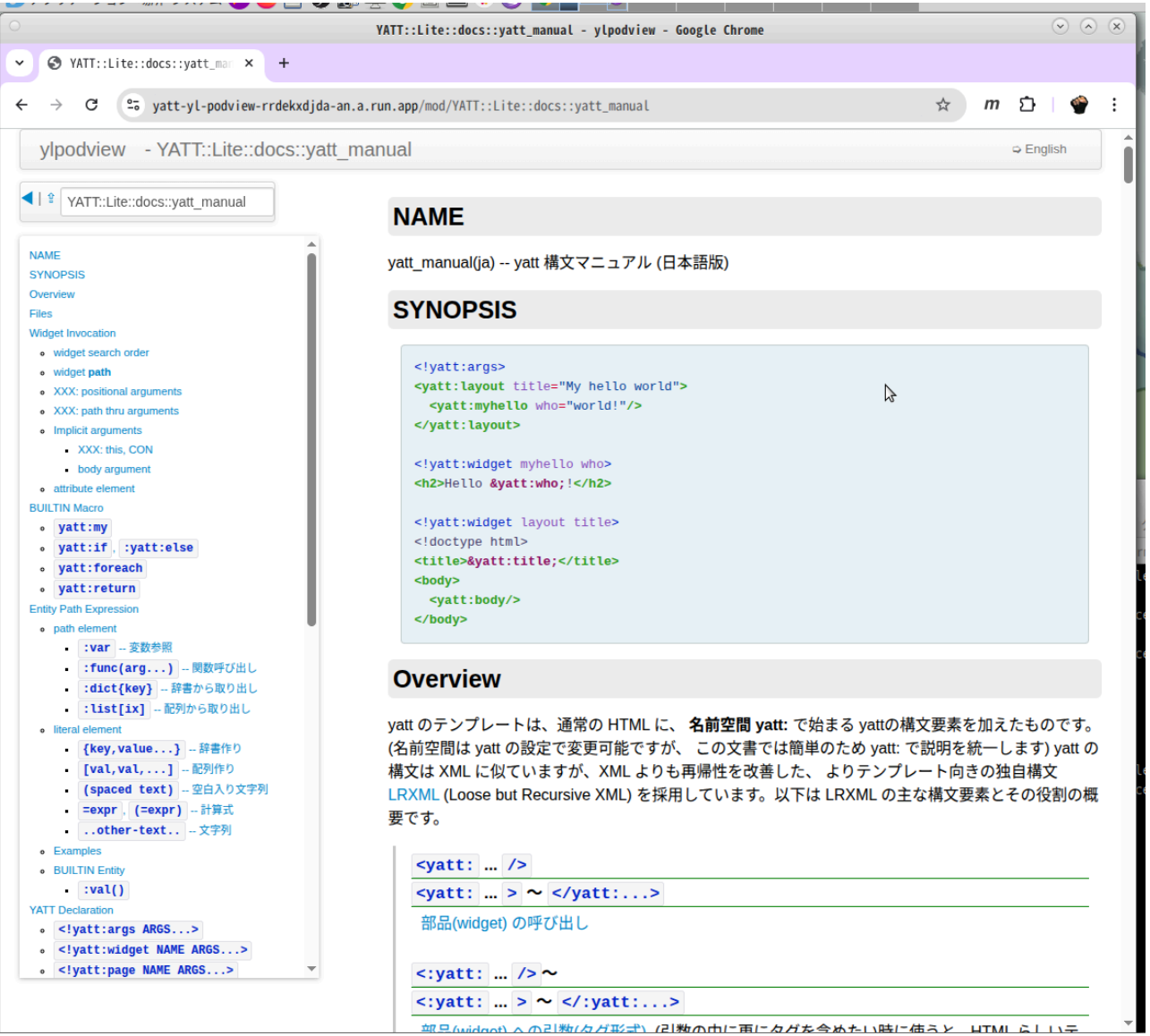
（〜ここで、デモに移動〜）

繰り返しですが...

- 任意のサンプル HTML を
- 適当なファイル名で、**ポンと置くだけ** で、動いて欲しい
    - （初期の PHP 的な世界観）
- **HTML 部品への括り出し** も、同レベルの操作にしたい

YATT の場合

- ファイルを置けば、それが route になる
- HTML 部品の括りだしも、
  - 別のファイルに括りだして、
  - `<yatt:ファイル名 />` タグを書くだけ
    - 名前空間は `yatt:` 以外にも設定できます



詳しくは...

マニュアルをご覧下さい

- `XXX:` あちこち未完ですが...🤦

10年以上も仕事に使って、改良を続けてます〜

Yet Another Template Transpiler - YATT

- YATT::Lite

  Perl 版

- yatt-js

  TypeScript 版, まだまだ開発中（Pre α レベル）

- 皆様のご意見を募集してます！
  - 一緒に Perl 書いてくれる人なども募集中！
  - あと、普通に友達ほしいっす！

追記：当日に頂いたご質問へのお返事です

当日はズレたお返事してしまった時もあったので、改めて書き直しました。

Q: Linux での LT かな？

A: はい、HP の Windows マシンの OS を Linux に載せ替えて使ってます。

スライドは marp で書いてます

Q: WebサーバもPerlですか？

A: はい、デモで使っていたのは plackup です。

その他の starman などの PSGI 互換サーバーも利用実績がありますし、古典的な CGI や FastCGI でも動きます。

Q: yattの名前の由来は？？？

A: **Y**et **A**nother **T**emplate **T**ranspiler です。

以前は Yet Another Template **Toolkit** と呼んでました。

Q: 部品のなかで、html タグ閉じずに動いてました？

A: はい、デモの時短のため、今日は閉じない書き方を用いました。
YATT は一般の HTML タグを解釈しない（`<yatt:..>` タグだけを解釈する）ので、
閉じない書き方をしても、それ自体はエラーにはなりません。

とはいえ HTML タグの閉じ忘れはエラーにしたいので、普段は header, footer の代わりに `layout.yatt` 一つの中で、タグの開きと閉じを対応させておきます。

```
<yatt:layout>
  ...
</yatt:layout>
```

layout.yatt の中で body引数を書いたところに `...` が埋め込まれます

Q: OSはFedoraですか？

A: はい、当日のデモで使ったのは Fedora42 です。

Q: Emacs信者ですか？

A: 信者を名乗れる自信はない(><)ですが、仕事の主役は依然として Emacs です〜

とはいえ VSCode にも追随したいので、一生懸命 Language Server を書くなどしてます。

Q: Perlの良さを一言で！

A: 雑にも堅くも書ける、懐の広さです！

個人的には、90年代から **変数名間違いを静的に検出する機能** が perl 本体に組み込まれていたこと（`use strict`, `use fields`）が、perl を使い続けてきた理由として大きいです。エディタの保存時に `perl -wc` するのがオススメです。

Q: 普段Perlでどんなサービス作られてますか？

A: お客様企業が医療従事者向けの Web アンケート屋さんなので、
そのインフラと、関連すること何でもです。(tcltk も使います)

最近で言うと、

- 社内業務効率化のための DB化と Web ダッシュボードづくり
- 特定の疾患領域で、先生が研究のため患者へアンケートをとって、その結果を把握しやすくする仕組み作り（...これもダッシュボードですね）

Q: Raku には移行しないんですか？

A: はい、今のところは...

元々 YATT は Raku への移行に備えてチームのスキルセットの Perl 依存度を下げるために作り始めた面があります。現在は TypeScript (deno)への移行を目指してます。

その他、頂いた反応へのお返事

Perl エンジニアさん！貴重！

Perl使い！渋い

Perl!!

→ ありがとうございます！

コンセプトが明確

シンプルなポン出しいいですね！

→ ありがとうございます！コンセプトが伝わって嬉しいです！

事前に想定したＱ＆Ａ

Q: タグに引数は渡せるの？

A: `<!yatt:args x y>` で宣言し、 `&yatt:x;` `&yatt:y;` で参照します

Q: （変数埋め込み等）出力のエスケープはどうなるの？

A: 変数宣言時の型で制御します。基本の text 型なら自動エスケープ。

Q: ループや条件分岐は書ける？

A: `<yatt:foreach>` がループで、 `<yatt:if>` が条件分岐です。

`&yatt:if(条件,then式,else式);` もあります

- 制御構造を拡張する、マクロ機能もあります

Q: 関数呼び出しを埋め込みたい時は？

A: Entity 関数呼び出し： `&yatt:func();` で関数を呼べます（予め登録）

Q: POST はどうするの？

A: `<!yatt:action 名前>` で、
POST のハンドラーを地の言語（Perl）で書くことが出来ます。

`*.ydo` として別ファイルに書くことも出来ます。

Q: データベースアクセスはどうするの？

A: YATT 自体では DB の機能は提供しません

通常の Perl のモジュールとして DBアクセスを書いて、
それを `&yatt:query();` みたいに呼べるようにします

Q: リクエストの状態はどこに持つ？

A: Request と Response をまとめた Connection オブジェクトが
各テンプレートに渡されます。ここに任意のキャッシュを保持できます。

Q: 中身はどうなっているの？

A: テンプレートを元に Perl のコードを生成する **トランスパイラー** です

生成したコードは（変更されるまで）キャッシュされます。

Q: チュートリアルはある？

A: （ほぼ）無いです、題材をもらえたら作ってみます

Q: マニュアルだけで作り始められる？

A: 肝心の Entity 定義周りが足りないので厳しい

Q: どのくらい使われてるの？

A: （多分）私のお客さん一社だけ...

Q: public にテンプレートを置くのは無用心では？

A: 別ディレクトリにも置けます。

- private テンプレート用の拡張子も設定できます（`*.ytmpl`）

Q: 拡張子は `.yatt` じゃないと駄目なの？

A: オプションで変更可能。 `.html` にも出来ます

- ただし、エディタの yatt モードの割り当てが問題に

Q: VS Code 拡張があるの？

A: あるけど github にしかないです

Q: その VS Code 拡張、どのくらい使える？

A: 普段は Emacs で書いてるので、まだ微妙...

Q: LanguageServer は何をサポートしてる？

```
textDocument/didOpen
textDocument/didSave
textDocument/didChange
textDocument/definition
textDocument/implementation
textDocument/documentSymbol
textDocument/hover
```

無いよりまし、くらい

Q: TypeScript 版の現状は？

A: (感覚で)3割も行かない

- でも、ご意見はぜひ

Q: （デモにある）nav_li タグの使い方は？

A:

```
<yatt:nav_li href="/">Top</yatt:nav_li>
<yatt:nav_li href="/new">新規作成</yatt:nav_li>
```