

## шаблонизаторы

### общая идея

фиксируется каркас из захардкоженной разметки

те места, которые "сменные", заменяются на "гнезда" или "placeholders"

которые традиционно обозначаются фигурными скобками, обычно двумя

и далее схема: вызываем метод **render**, которому передаём имя **представления (view, вьюха)**

и JSON Options, где перечислены пары ключ:значение как имяГнезда:значениеПодстановки

*представление* - это динамическое генерируемое нечто, скорее всего по шаблону

handlebars	pug (ранее jade)				
должен быть <i>выбран</i> в качестве механизм шаблонов	шаблонизатор по умолчанию <a href="http://html2jade.org/">http://html2jade.org/</a>				
<b>npm i -S express-handlebars</b> let handlebars = require('express-handlebars') .create({ defaultLayout:'main' });  app.engine('handlebars', handlebars.engine); app.set('view engine', 'handlebars');	<b>npm i -S pug</b> устанавливаем, далее он сам подцепляет				
<div><div><p><i>макет - меташаблон, шаблон вьюшек</i> файл типа views/<b>layouts</b>/main.handlebars - в нём есть {{{body}}} и т.п. т.е. placeholder для подстановки <b>тегов</b> для представлений If you don't want Handlebars to escape a value, use the "triple-stash", {{{.</p></div><div><p><i>представление, вьюшка - шаблон страницы</i> файл типа views/about.handlebars - т.е. конкретика для маршрута; там HTML и {{..}} куда подставляются <b>не-теги</b></p></div></div> <p>дефолтный макет используется всеми вьюшками если не указано иное</p> <table><tr><td>views/layouts/main.handlebars</td><td>views/root.handlebars</td></tr><tr><td>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Express Site&lt;/title&gt;&lt;meta charset="utf-8"&gt; &lt;/head&gt; &lt;body&gt;   {{{body}}} &lt;/body&gt; &lt;/html&gt;</td><td>&lt;h1&gt;Welcome, friends!&lt;/h1&gt; &lt;p&gt;You are at {{text}} page!&lt;/p&gt;</td></tr></table>	views/layouts/main.handlebars	views/root.handlebars	<!DOCTYPE html> <html> <head> <title>Express Site</title><meta charset="utf-8"> </head> <body> {{{body}}} </body> </html>	<h1>Welcome, friends!</h1> <p>You are at {{text}} page!</p>	<p><i>вот простая вьюха</i> views/main.pug</p> <pre>html   head     title= title   body     h1= title     p Hello everyone...     a(href='/') Homepage!</pre>
views/layouts/main.handlebars	views/root.handlebars				
<!DOCTYPE html> <html> <head> <title>Express Site</title><meta charset="utf-8"> </head> <body> {{{body}}} </body> </html>	<h1>Welcome, friends!</h1> <p>You are at {{text}} page!</p>				
app.get('/:key', (req, res) => { res.render('root', {text: req.params.key} ); });	подключаем app.get('/:key', (req, res) => { res.render('main.pug', {title: 'page ' + req.params.key + ' is in development!'} ); });				

без шаблона	с шаблоном
<code>res.status(404).send('&lt;h2&gt;Not yet here, pardon!&lt;/h2&gt;');</code>	<code>res.status(404).render('404.pug',{....});</code>

Выше мы видели, что макет забирает в свой `{{body}}` код из представлений. Очевидно, может быть нужно менять также и другие разделы, вроде `head`.

server.js	
<pre>handlebars = require('express-handlebars') .create({ defaultLayout:'main',   helpers: {     section: function(name, options){       if(!this._sections) this._sections = {};       this._sections[name] = options.fn(this);       return null;     }   } });  ...  .get('/:key', (req, res) =&gt; {   res.render('root',     {text: req.params.key}   ); });</pre>	
views/layouts/main.handlebars	views/root.handlebars
<pre>&lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;    {{{_sections.head}}}  &lt;/head&gt;  &lt;body&gt;    {{{body}}}  &lt;/body&gt; &lt;/html&gt;</pre>	<pre>{{#section 'head'}}   &lt;title&gt;Root view: {{text}}&lt;/title&gt;   &lt;meta charset="utf-8"&gt;&lt;meta name="author" content="Alan Turing"&gt; {{/section}}  &lt;h1&gt;Welcome, friends!&lt;/h1&gt; &lt;p&gt;You are at {{text}} page!&lt;/p&gt;</pre>