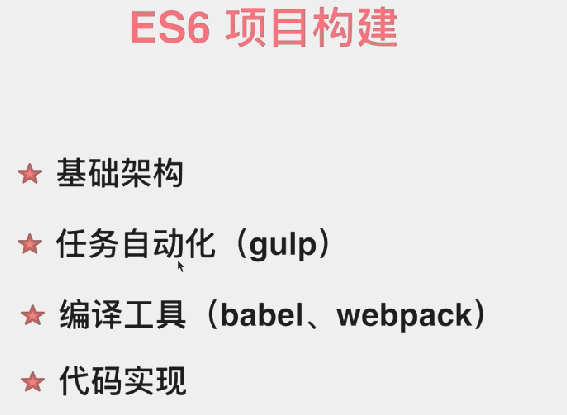
# ES6零基础教学解析彩票项目

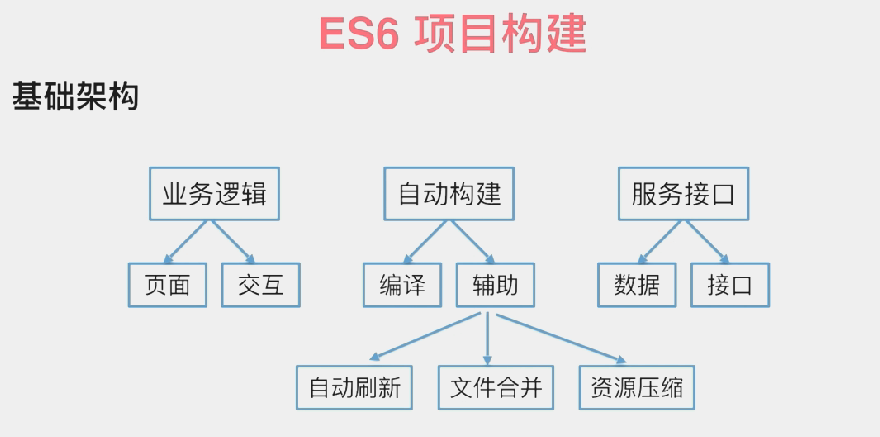
项目地址：<http://es6lottery.t.imooc.io/>

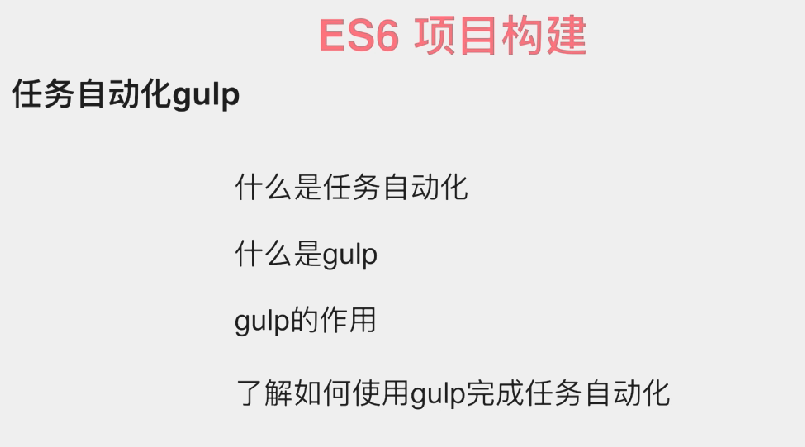
## 2-1项目构建介绍



Babel:是将ES6编译为ES5，让浏览器识别的。

Webpack：是处理模块化、项目依赖关系的。





1. 任务自动化：减少人工操作，让机器（你的电脑） 自动监听你所有的操作，自动去响应。
2. Gulp：完成自动化的工具，增强工作流程的一个工具。
3. Gulp的作用：利用各种插件，完成任务自动化。（node.js）
4. 如何使用Gulp：

看gulp的中文文档，API。

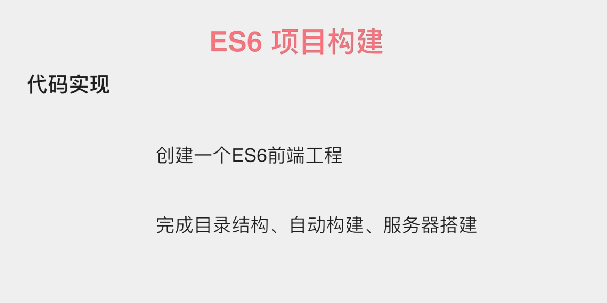
Gulp中文网看API，了解插件。



Babel：javascript的一个编译器。ES6本身是不能在浏览器上运行的，需要编译。

Webpack：解决模块化的东西。

Webpack-stream：是在gulp的基础上用的。因为gulp是基于stream的。



## 2-2项目目录创建

Es6/

App/（前端界面）

Js/

Class/（类）

test.js

Index.js（js入口文件）

Css/

Views/（视图模板ejs）

Error.ejs

Index.ejs

Server/

>Npm –g install express,npm –g install express-generator（安装命令工具）

>Express –e . //使用express脚手架，引擎模板用ejs。

Tasks/（工具）

Util

Args.js //处理命令行参数

>Npm init //生成package.json文件，然后一路回车（输入yes之后无响应？）

package.json

.babelrc //设置babel编译工具的配置文件（文件名必须是.babelrc）

gulpfile.babel.js //（加babel是因为针对es6文件，所以此文件名也是必须）

## 2-3 命令行处理、创建JS编译任务

编写tasks/util/ args.js文件 //处理命令行参数的

Import yargs from “yargs”

编写tasks/ scripts.js文件 //构建脚本，对js做处理的

引入各种包，然后安装。（npm镜像淘宝可以解决安装中的问题）

npm install gulp-if gulp-concat webpack webpack-stream vinyl-named gulp-livereload gulp-plumber gulp-rename gulp-uglify gulp-util yargs --save-dev

--save-dev ：可以在pakage.json中写入包依赖的信息 字段

创建一个gulp任务。

## 2-4 创建模板、服务任务脚本

编写tasks/pages.js文件。 //创建一个处理模板的构建脚本

编写tasks/css.js文件。 //创建一个处理CSS的构建脚本

编写tasks/server.js文件。 //创建一个服务器的构建脚本

## 2-5 文件自动监听，项目构建测试

Scriprs.js文件 是把app下的前端原始文件 写入server中的public中。

当源文件发生改变的时候，我们还需要tasks/browser.js文件，实现动态编译更新到server中。

tasks/browser.js：将app下的原始文件有变化时，自动的更新至server下。

tasks/clean.js：怕生成文件覆盖原有文件的问题，所以每次清空server/public、server/views。

安装此阶段用到的包

npm install gulp-util gulp-live-server del --save-dev

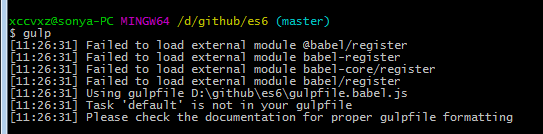
tasks/bulid.js：把所有任务都关联起来。

tasks/default.js：默认文件默认任务。Default

执行一下gulp试试，

调试改错：

发现错误。



1-4：没有安装babel，在scripts.js中的loader中，忘记安装了。。

5：gulpfile.babel.js 为空文件。

6.：gulpfile文件没写。

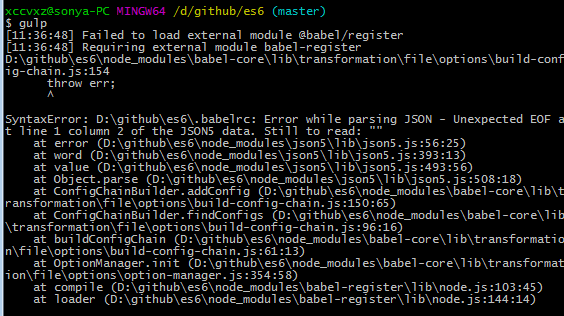
解决：

1．安装bable。

npm install babel-loader babel-core babel-preset-env --save-dev

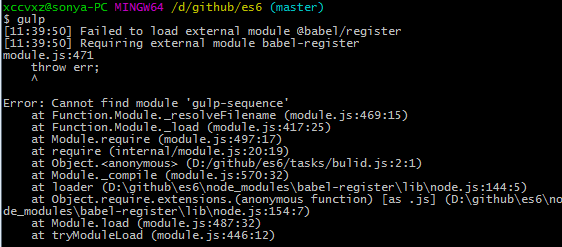
2.写gulfile.babel.js文件：加载tasks下的所有文件。

再运行一次。



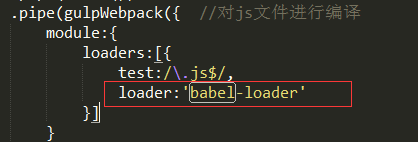
又报错，原因为.babelrc没写，写完之后需要安装一个包。

npm install babel-preset-es2015 --save-dev。



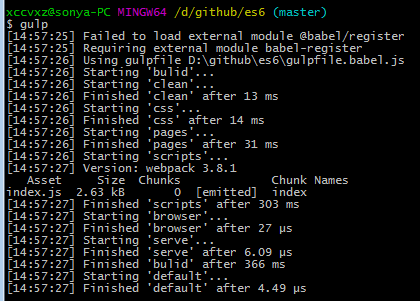
安装gulp-sequence。

错误：把babel改为babel-loader。

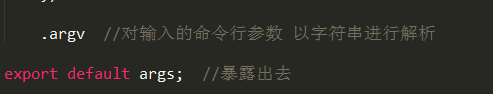


Build中的任务顺序，写的是每个gulp.task(“任务名”)中的任务名，不是文件名！

Ok！终于执行成功！

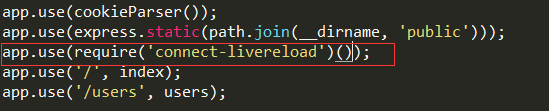


但gulp –watch 不成功，没有监听。改正错误in Args.js：(把args暴露出去)



在浏览器中输入localhost:3000；(此时不能自动更新页面)

在app.js中加入下面一句话：（一定要在express.static的下面）



然后安装connect-livereload包。

## 3-1 let.const命令

Let是块作用域。

不能重复声明变量。

let a=1;

let a=2;

报错！

ES6默认开启严格模式。

Const是常量，不能修改，且声明时必须赋值，也有块作用域。

但对象中的属性可以增加、修改。因为指向对象的指针是不变的，而对象本身被容是可以改变的。

## 3-2解构赋值

## 5-1课程总结



