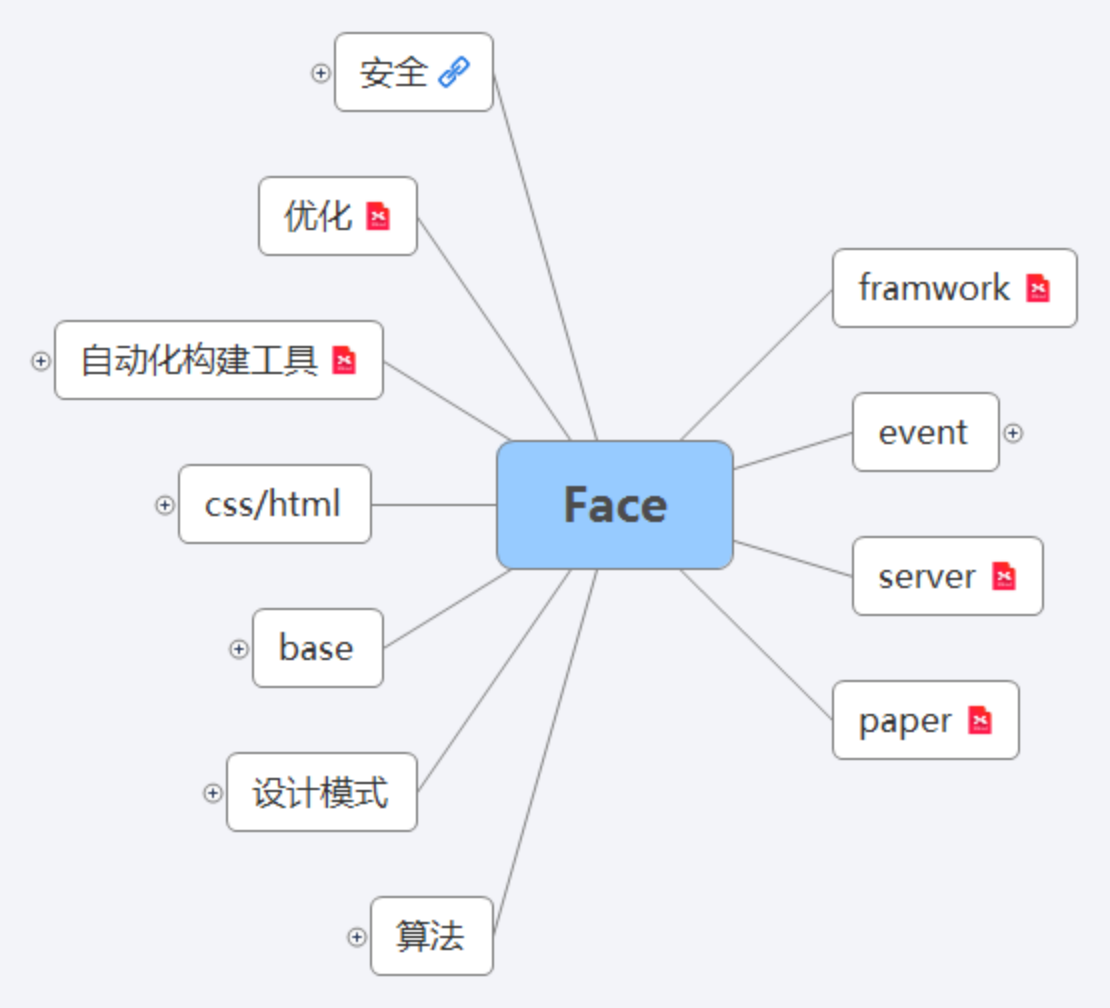
Face



1. framwork
2. event
   - 2.1. borwer
   - 2.2. node
   - 2.3. 解析类
     - 2.3.1. 不要混淆nodejs和浏览器中的event loop
   - 2.4. 阮一峰

2.4.1. borwer

只要主线程空了，就会去读取"任务队列"，这就是JavaScript的运行机制。这个过程会不断重复。

栈

执行栈

任务队列

事件的队列

I/O操作

定时器

setTimeout(fn,0)的含义是，指定某个任务在主线程最早可得的空闲时间执行，也就是说，尽可能早得执行。它在"任务队列"的尾部添加一个事件，因此要等到同步任务和"任务队列"现有的事件都处理完，才会得到执行。

2.4.2. node

process.nextTick

setImmediate

setImmediate指定的回调函数，总是排在setTimeout前面。实际上，这种情况只发生在递归调用的时候

```
setImmediate(function (){
  setImmediate(function A() {
    console.log(1);
    setImmediate(function B(){console.log(2);});
  });

  setTimeout(function timeout() {
    console.log('TIMEOUT FIRED');
```

```
    }, 0);
});
```

3. server

4. paper

5. 算法

   5.1. 排序

      5.1.1. 冒泡排序

      5.1.2. 选择排序

      5.1.3. 插入排序

      5.1.4. 快速排序

      5.1.5. 桶排序

      5.1.6. js标准参考教程

      5.1.7. js重用排序

   5.2. 递归

   5.3. 数据结构

   5.4. 时间复杂度和空间复杂度

   5.5. js常用算法

      5.5.1. 子主题 1

      5.5.2. 子主题 2

7.1.1. ajax

7.2. js

7.2.1. 常规题

子主题 1

子主题 2

子主题 3

子主题 4

7.2.2. 常规操作

内存泄漏教程

let

变量的提升

let

创建（变量提升）

初始化

赋值

var

创建（变量提升）

初始化(变量提升)

赋值

for(let)

for(let i = 0; i< 5; i++) { 循环体 } 在每次执行循环体之前，JS 引擎会把 i 在循环体的上下文中重新声明及初始化一次

js深入理解

this指向

原型，原型链、作用域、作用域链

闭包

8. css/html

8.1. css

8.2. html

8.2.1. canvas

9. 自动化构建工具

9.1. webapck

9.2. glup

10. 优化

11. 安全

11.1. XSS(Cross-Site Script)

11.1.1. 类型

反射型(非持久型)

存储型(持久型)

DOM-Based型

11.1.2. 防御

给cookie设置httpOnly属性

脚本无法读取该Cookie,自己也不能用，非根本解决XSS

Web相关编码和转义

URL 编码

HTML 编码

Javascript 转义

输入检查

URL解析环境

11.2. CSRF（Cross Site Request Forgery 跨站请求伪造）

11.2.1. 防御

验证码

refer 验证

token验证

11.3. DDOS攻击（分布式拒绝服务(Distribute Denial Of Service)）

11.3.1. 验证码

11.3.2. 限制请求频率

11.3.3. 增加机器增加服务带宽

11.3.4. 设置自己的业务为分布式服务

11.3.5. 使用主流云系统和 CDN

11.3.6. 提高 web server 的负载

11.4. HTTP劫持