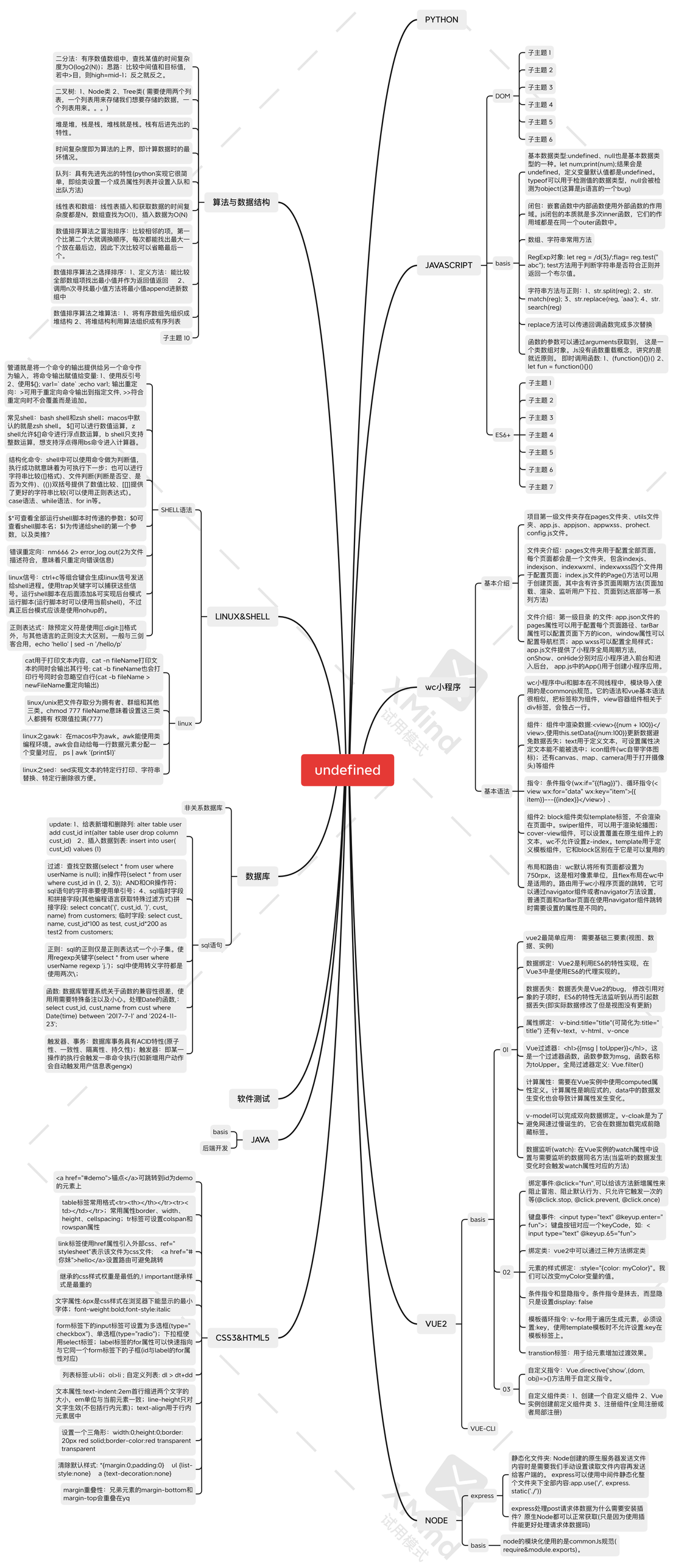
undefined
PYTHON
JAVASCRIPT
DOM
子主题 1
子主题 2
子主题 3
子主题 4
子主题 5
子主题 6
basis
基本数据类型:undefined、null也是基本数据类型的一种。let num;print(num);结果会是undefined，定义变量默认值都是undefined。typeof可以用于检测值的数据类型，null会被检测为object(这算是js语言的一个bug)
闭包：嵌套函数中内部函数使用外部函数的作用域。js闭包的本质就是多次inner函数，它们的作用域都是在同一个outer函数中。
数组、字符串常用方法
RegExp对象: let reg = /d(3)/;flag= reg.test(" abc"); test方法用于判断字符串是否符合正则并返回一个布尔值。
字符串方法与正则：1、str.split(reg); 2、str.match(reg); 3、str.replace(reg, 'aaa'); 4、str.search(reg)
replace方法可以传递回调函数完成多次替换
函数的参数可以通过arguments获取到， 这是一个类数组对象。Js没有函数重载概念，讲究的是就近原则， 即时调用函数: 1、(function(){})() 2、let fun = function(){}()
ES6+
子主题 1
子主题 2
子主题 3
子主题 4
子主题 5
子主题 6
子主题 7
wc小程序
基本介绍
项目第一级文件夹存在pages文件夹、utils文件夹、app.js、appjson、appwxss、prohect.config.js文件。
文件夹介绍：pages文件夹用于配置全部页面，每个页面都会是一个文件夹，包含indexjs、indexjson、indexwxml、indexwxss四个文件用于配置页面；index.js文件的Page()方法可以用于创建页面，其中含有许多页面周期方法(页面加载、渲染、监听用户下拉、页面到达底部等一系列方法)
文件介绍：第一级目录 的文件: app.json文件的pages属性可以用于配置每个页面路径、tarBar属性可以配置页面下方的icon，window属性可以配置导航栏页； app.wxss可以配置全局样式；app.js文件提供了小程序全局周期方法，onShow、onHide分别对应小程序进入前台和进入后台， app.js中的App()用于创建小程序应用。
基本语法
wc小程序中ui和脚本在不同线程中，模块导入使用的是commonjs规范。它的语法和vue基本语法很相似，把标签称为组件，view容器组件相关于div标签，会独占一行。
组件：组件中渲染数据:<view>{{num + 100}}</view>,使用this.setData({num:100})更新数据避免数据丢失；text用于定义文本，可设置属性决定文本能不能被选中；icon组件(wc自带字体图标)；还有canvas、map、camera(用于打开摄像头)等组件
指令：条件指令(wx:if="{{flag}}")、循环指令(<view wx:for="data" wx:key="item">{{item}}---{{index}}</view>) 、
组件2: block组件类似template标签，不会渲染在页面中。swiper组件，可以用于渲染轮播图；cover-view组件，可以设置覆盖在原生组件上的文本，wc不允许设置z-index。template用于定义模板组件，它和block区别在于它是可以复用的
布局和路由：wc默认将所有页面都设置为750rpx，这是相对像素单位，且flex布局在wc中是适用的。路由用于wc小程序页面的跳转，它可以通过navigator组件或者navigator方法设置，普通页面和tarBar页面在使用navigator组件跳转时需要设置的属性是不同的。
VUE2
basis
01
vue2最简单应用： 需要基础三要素(视图、数据、实例)
数据绑定：Vue2是利用ES6的特性实现，在Vue3中是使用ES6的代理实现的。
数据丢失：数据丢失是Vue2的bug， 修改引用对象的子项时，ES6的特性无法监听到从而引起数据丢失(即实际数据修改了但是视图没有更新)
属性绑定： v-bind:title="title"(可简化为:title="title") 还有v-text，v-html、v-once
Vue过滤器：<h1>{{msg | toUpper}}</h1>，这是一个过滤器函数，函数参数为msg，函数名称为toUpper。全局过滤器定义: Vue.filter()
计算属性：需要在Vue实例中使用computed属性定义。计算属性是响应式的，data中的数据发生变化也会导致计算属性发生变化。
v-model可以完成双向数据绑定。v-cloak是为了避免网速过慢诞生的，它会在数据加载完成前隐藏标签。
数据监听(watch): 在Vue实例的watch属性中设置与需要监听的数据同名方法(当监听的数据发生变化时会触发watch属性对应的方法)
02
绑定事件:@click="fun",可以给该方法新增属性来阻止冒泡、阻止默认行为、只允许它触发一次的等(@click.stop, @click.prevent, @click.once)
键盘事件：<input type="text" @keyup.enter="fun">；键盘按钮对应一个keyCode，如: < input type="text" @keyup.65="fun">
绑定类：vue2中可以通过三种方法绑定类
元素的样式绑定：:style="{color: myColor}"。我们可以改变myColor变量的值。
条件指令和显隐指令。条件指令是抹去，而显隐只是设置display: false
模板循环指令: v-for用于遍历生成元素，必须设置:key，使用template模板时不允许设置:key在模板标签上。
transtion标签：用于给元素增加过渡效果。
03
自定义指令：Vue.directive('show',(dom, obj)=>{}方法用于自定义指令。
自定义组件类：1、创建一个自定义组件 2、Vue实例创建前定义组件类 3、注册组件(全局注册或者局部注册)
VUE-CLI
NODE
express
静态化文件夹: Node创建的原生服务器发送文件内容时是需要我们手动设置读取文件内容再发送给客户端的。 express可以使用中间件静态化整个文件夹下全部内容:app.use('/', express.static('./'))
express处理post请求体数据为什么需要安装插件？原生Node都可以正常获取(只是因为使用插件能更好处理请求体数据吗)
basis
node的模块化使用的是commonJs规范(require&module.exports)。
算法与数据结构
二分法：有序数值数组中，查找某值的时间复杂度为O(log2(N))；思路：比较中间值和目标值，若中>目，则high=mid-1；反之就反之。
二叉树：1、Node类 2、Tree类(需要使用两个列表，一个列表用来存储我们想要存储的数据，一个列表用来。。。)
堆是堆，栈是栈，堆栈就是栈。栈有后进先出的特性。
时间复杂度即为算法的上界，即计算数据时的最坏情况。
队列：具有先进先出的特性(python实现它很简单，即给类设置一个成员属性列表并设置入队和出队方法)
线性表和数组：线性表插入和获取数据的时间复杂度都是N，数组查找为O(1)，插入数据为O(N)
数值排序算法之冒泡排序：比较相邻的项，第一个比第二个大就调换顺序，每次都能找出最大一个放在最后边，因此下次比较可以省略最后一个。
数值排序算法之选择排序：1、定义方法：能比较全部数组项找出最小值并作为返回值返回 2、调用n次寻找最小值方法将最小值append进新数组中
数值排序算法之堆算法：1、将有序数组先组织成堆结构 2、将堆结构利用算法组织成有序列表
子主题 10
LINUX&SHELL
SHELL语法
管道就是将一个命令的输出提供给另一个命令作为输入，将命令输出赋值给变量: 1、使用反引号 2、使用${}; var!=`date` ;echo var!; 输出重定向：>可用于重定向命令输出到指定文件, >>符合重定向时不会覆盖而是追加。
常见shell：bash shell和zsh shell；macos中默认的就是zsh shell。 3.[]可以进行数值运算，zshell允许$[]命令进行浮点数运算，b shell只支持整数运算，想支持浮点得用bs命令进入计算器。
结构化命令：shell中可以使用命令做为判断值，执行成功就意味着为可执行下一步；也可以进行字符串比较(]格式)、文件判断(判断是否空、是否为文件)、(())双括号提供了数值比较、[[]]提供了更好的字符串比较(可以使用正则表达式)。case语法、while语法、for in等。
$*可查看全部运行shell脚本时传递的参数；$0可查看shell脚本名；$1为传递给shell的第一个参数，以及类推?
错误重定向：nm666 2> error_log.out(2为文件描述符合，意味着只重定向错误信息)
linux信号：ctrl+c等组合键会生成linux信号发送给shell进程。使用trap关键字可以捕获这些信号。运行shell脚本在后面添加&可实现后台模式运行脚本(运行脚本时可以使用当前shell)，不过真正后台模式应该是使用nohup的。
正则表达式：除预定义符是使用[[:digit:]]格式外，与其他语言的正则没太大区别。一般与三剑客合用，echo 'hello' | sed -n '/hello/p'
linux
cat用于打印文本内容，cat -n fileName打印文本的同时会输出其行号; cat -b fineName也会打印行号同时会忽略空白行(cat -b fileName > newFileName重定向输出)
linux/unix把文件存取分为拥有者、群组和其他三类。chmod 777 fileName意味着设置这三类人都拥有 权限值拉满(777)
linux之gawk：在macos中为awk。awk能使用类编程环境。awk会自动给每一行数据元素分配一个变量对应， ps | awk '{print$1}'
linux之sed：sed实现文本的特定行打印、字符串替换、特定行删除很方便。
软件测试
JAVA
basis
后端开发
数据库
非关系数据库
sql语句
update: 1、给表新增和删除列: alter table user add cust_id int(alter table user drop column cust_id) 2、插入数据到表: insert into user(cust_id) values (1)
过滤：查找空数据(select * from user where userName is null); in操作符(select * from user where cust_id in (1, 2, 3)); AND和OR操作符; sql语句的字符串要使用单引号; 4、sql临时字段和拼接字段(其他编程语言获取特殊过滤方式)拼接字段: select concat('(', cust_id, ')', cust_name) from customers; 临时字段: select cust_name, cust_id*100 as test, cust_id*200 as test2 from customers;
正则：sql的正则仅是正则表达式一个小子集。使用regexp关键字(select * from user where userName regexp 'j.'); sql中使用转义字符都是使用两次\;
函数: 数据库管理系统关于函数的兼容性很差，使用用需要特殊备注以及小心。处理Date的函数,: select cust_id, cust_name from cust where Date(time) between '2017-7-1' and '2024-11-23';
触发器、事务：数据库事务具有ACID特性(原子性、一致性、隔离性、持久性)；触发器：即某一操作的执行会触发一串命令执行(如新增用户动作会自动触发用户信息表gengx)
CSS3&HTML5
锚点可跳转到id为demo的元素上
table标签常用格式:<tr><th></th></tr><tr><td></td></tr>；常用属性border、width、height、cellspacing；tr标签可设置colspan和rowspan属性
link标签使用href属性引入外部css、ref=" stylesheet"表示该文件为css文件； hello设置路由可避免跳转
继承的css样式权重是最低的,! important继承样式是最重的
文字属性:6px是css样式在浏览器下能显示的最小字体；font-weight:bold;font-style:italic
form标签下的input标签可设置为多选框(type=" checkbox")、单选框(type="radio")；下拉框使用select标签；label标签的for属性可以快速指向与它同一个form标签下的子框(id与label的for属性对应)
列表标签:ul>li；ol>li ; 自定义列表: dl > dt+dd
文本属性:text-indent:2em首行缩进两个文字的大小，em单位与当前元素一致；line-height只对文字生效(不包括行内元素)； text-align用于行内元素居中
设置一个三角形：width:0;height:0;border:20px red solid;border-color:red transparent transparent
清除默认样式: *{margin:0;padding:0} ul {list-style:none} a {text-decoration:none}
margin重叠性：兄弟元素的margin-bottom和margin-top会重叠在yq