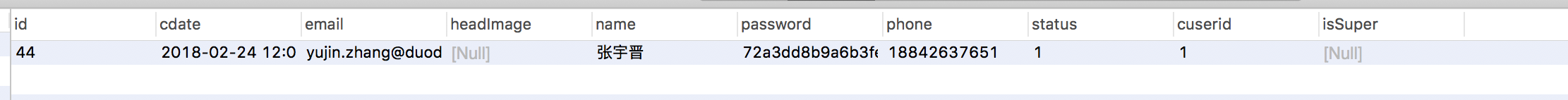
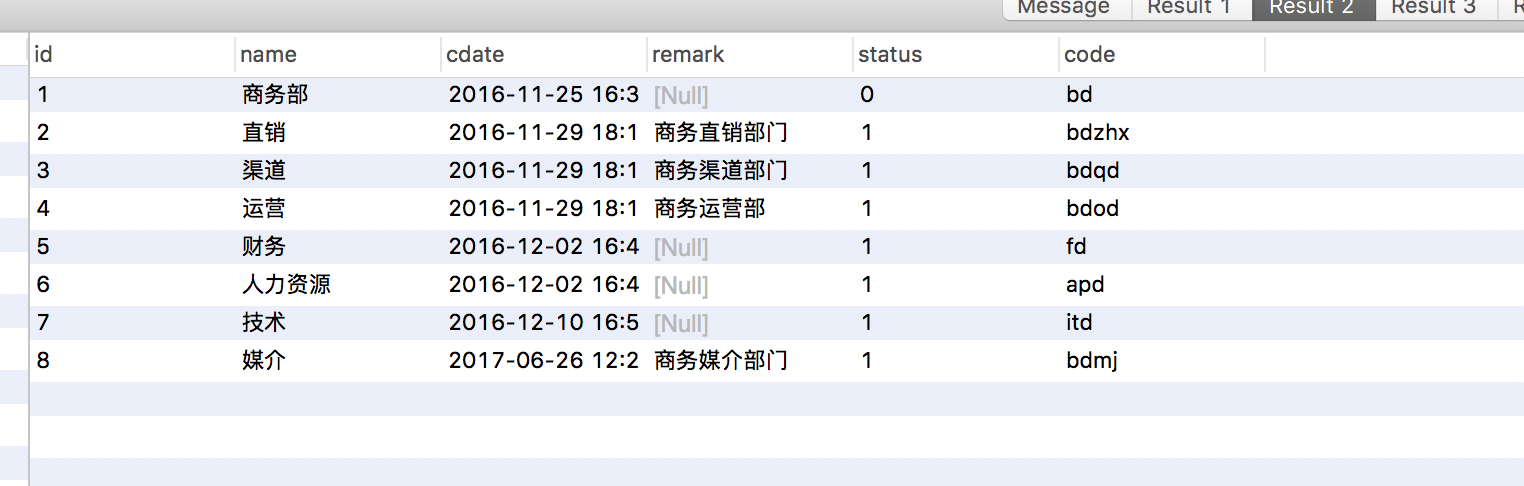
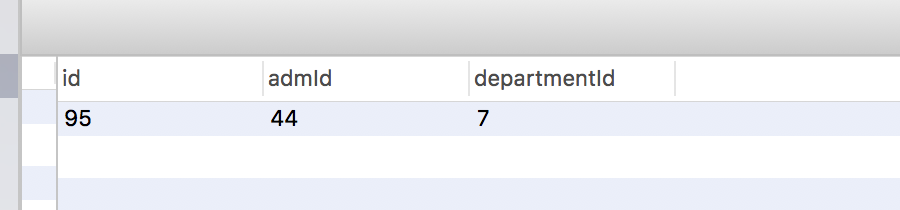
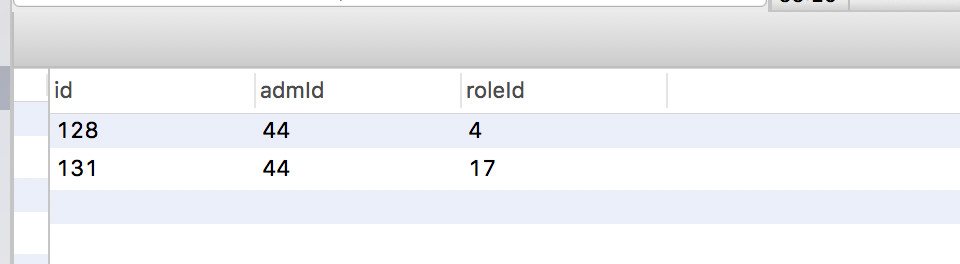
1、前端查询数据库进行菜单的显示

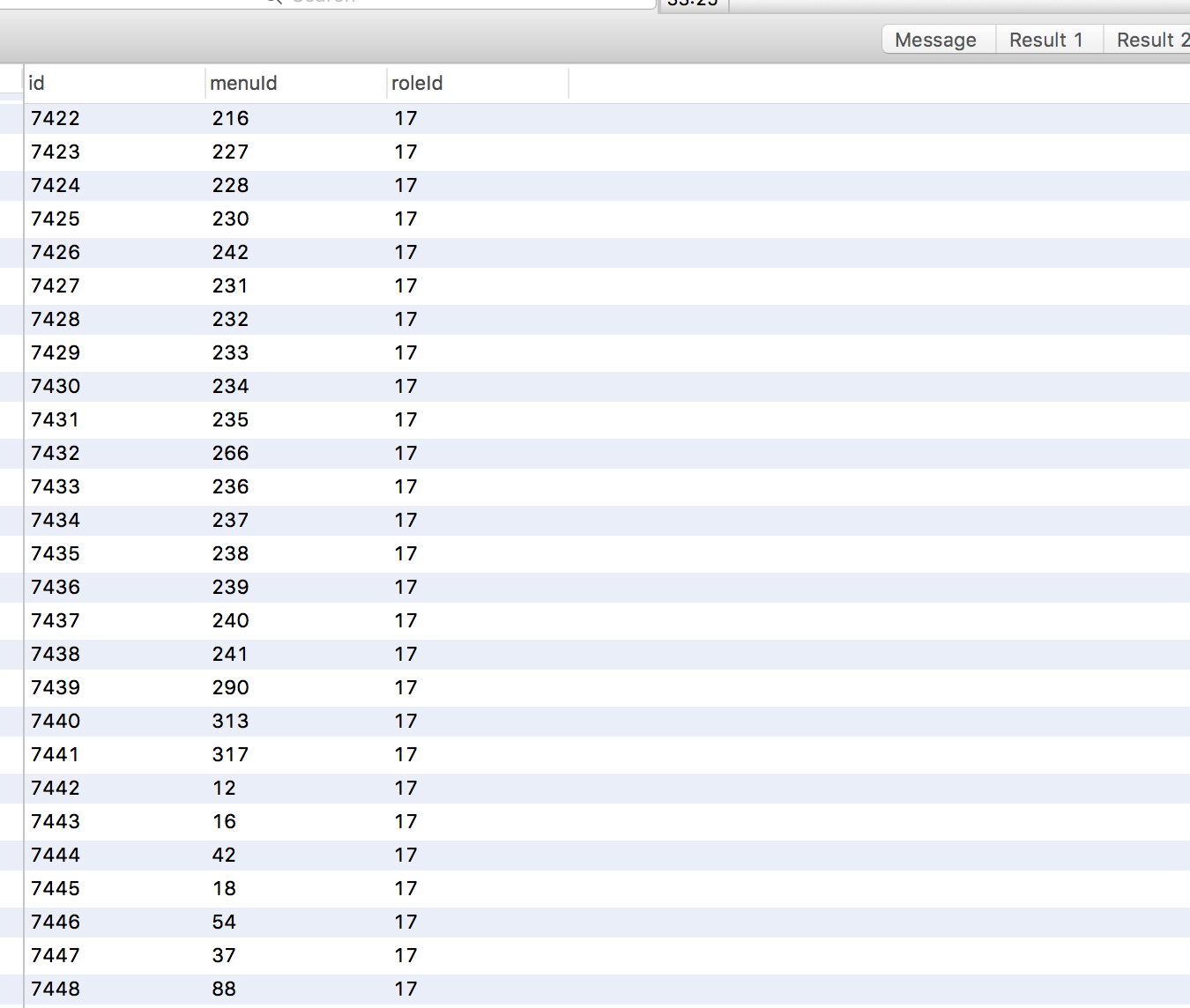
# 1、首先需要知道几个数据库表

# 1、系统管理员用户表，包括用户名和密码 ，创建管理员用户就会到这里  
 select \* from sys\_admin\_user s  
 where s.email='yujin.zhang@duodian.com';

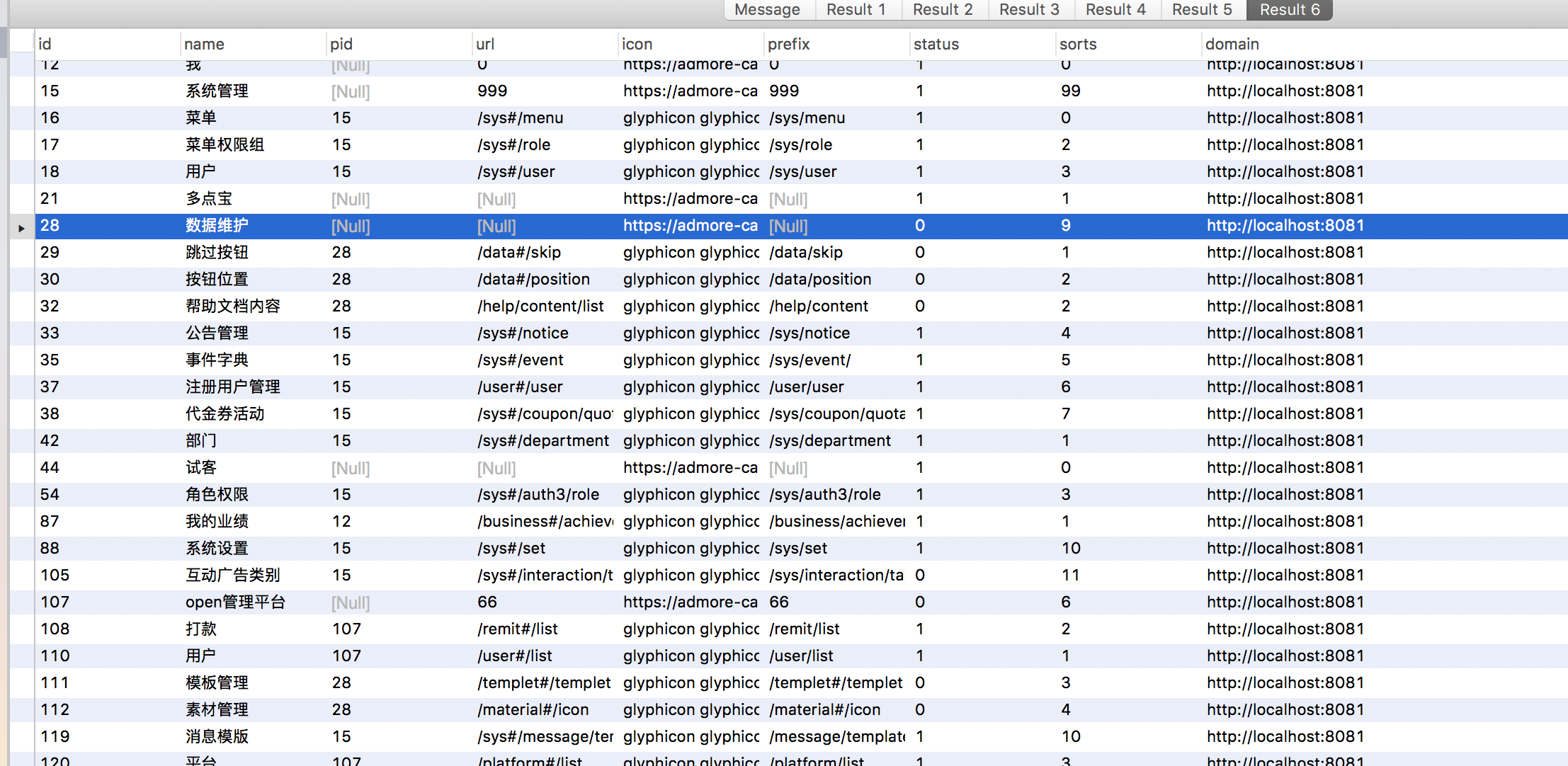
  
 -- 部门表  
 SELECT \* from sys\_department ;

  
 -- 管理员部门对应表  
 select \* from sys\_dept\_admin\_user s  
 WHERE s.admid = '44';

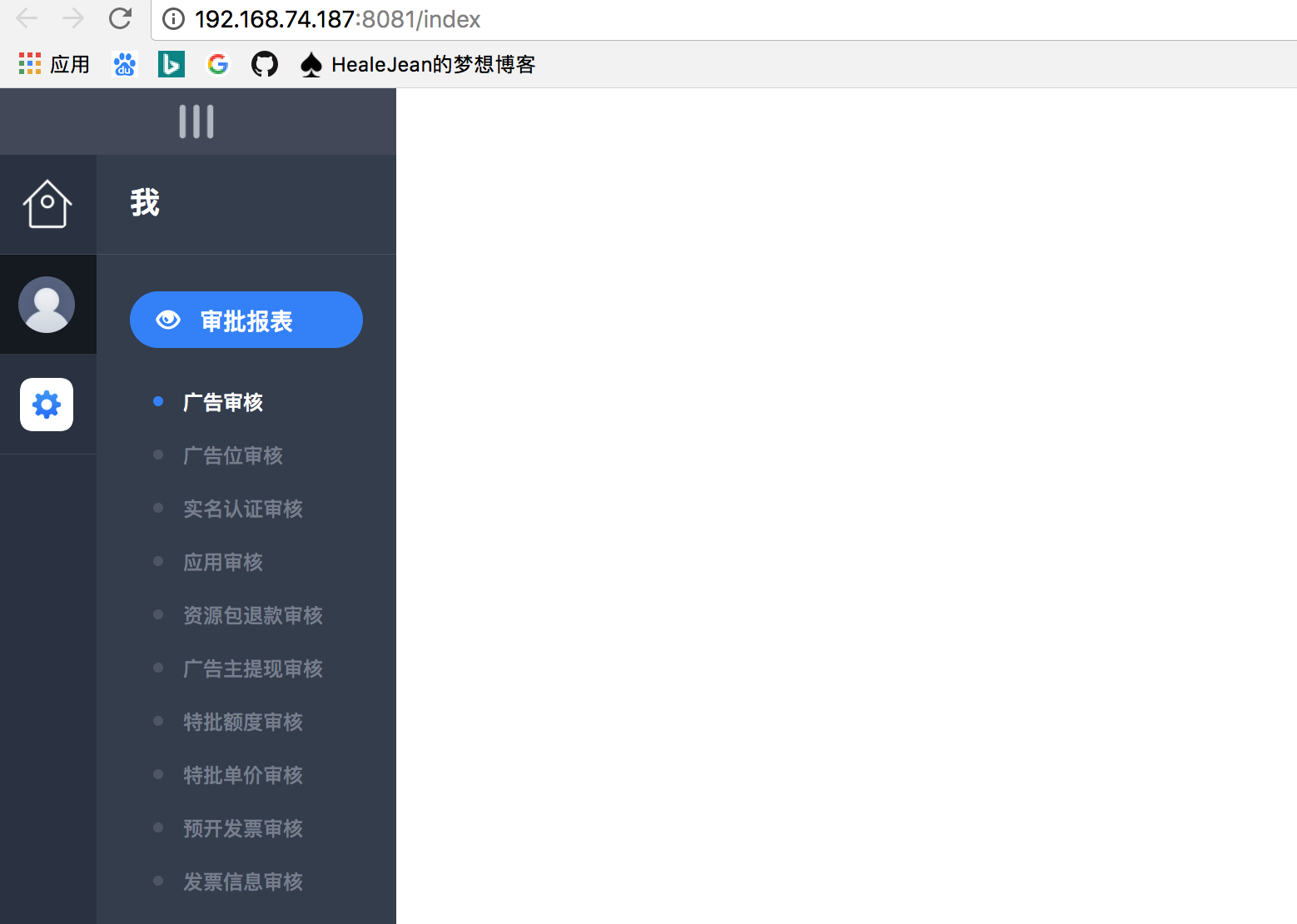
  
  
# 2、系统角色 和 用户id表  
 -- private Long id;  
 -- private Long roleId; 角色id  
 -- private Long admId; 管理员 id 如果为1为则为系统管理员  
 select \* from sys\_auth\_role\_admin\_user s  
 where s.admid = 44;  


# 3、角色菜单对应表  
 -- private Long id;  
 -- private Long roleId; //角色id  
 -- private Long menuId; //菜单id  
 select \* from sys\_auth\_role\_menu s  
 where s.roleId in(4,17) ;  


-- 4、系统菜单 pid为上级  
 select \* from sys\_menu ;



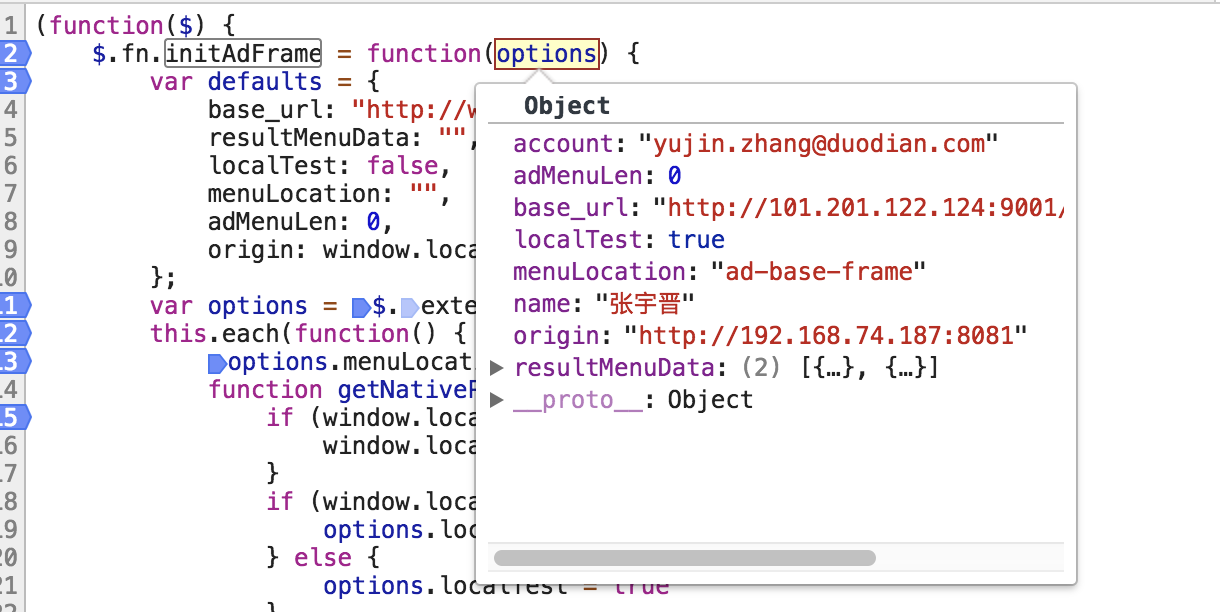
# 2、观察我设置的账户的菜单 （发现大菜单只有两个 我和系统管理）



# 3、开始展示（具体不知道angularJs 从哪里开始进入就开始调用这个js了 frame\_0.7.min.js）

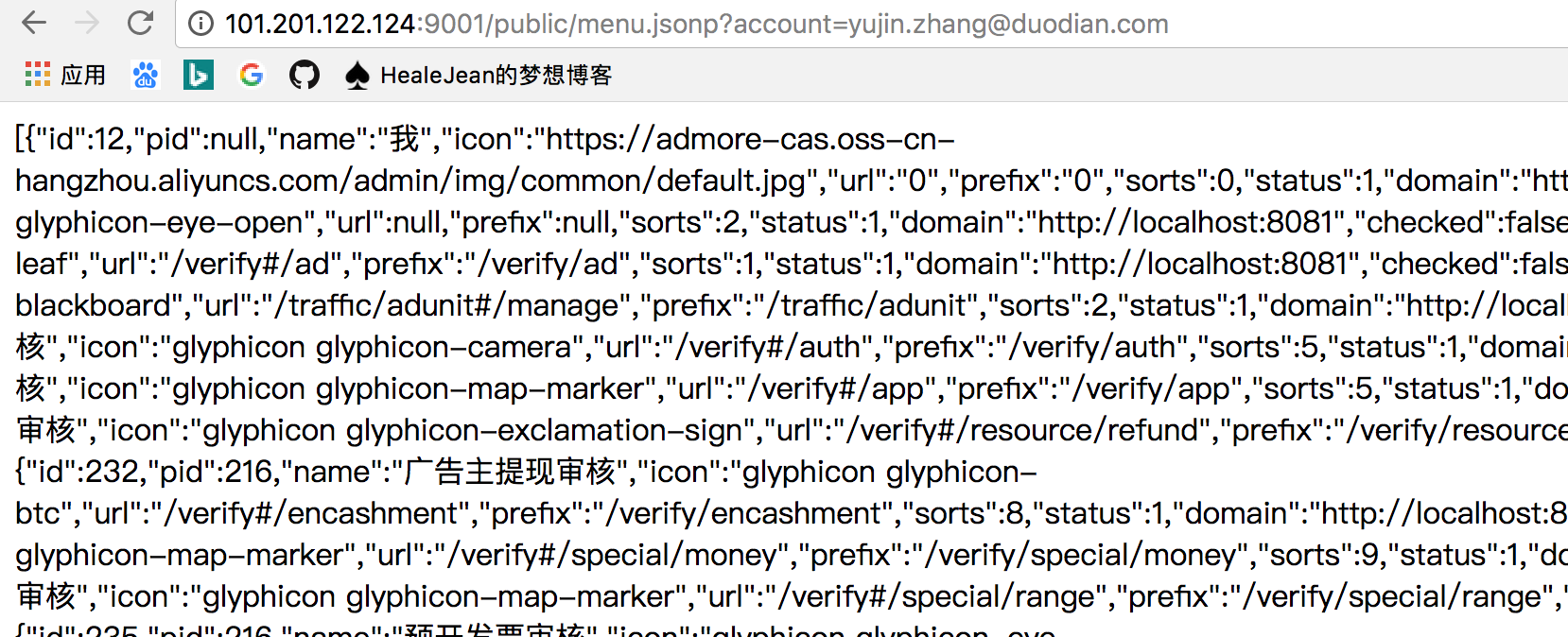
## 解释:

## 1、options:



## 2、本地访问测试环境上的101.201.122.124：9001 而正是环境则是访问下面的 <http://work.duodian.com/menu.jsonp>

## 3、展示的时候ajax访问数据 account 为登录的邮箱账户



;(function($){  
 $.fn.initAdFrame = function(options){  
 var defaults = {  
 base\_url:'http://work.duodian.com/menu.jsonp',  
 resultMenuData:'',  
 localTest:false,  
 menuLocation:'',  
 adMenuLen:0,  
 origin:***window***.location.origin  
 }  
 var options = $.extend(defaults,options);  
 this.each(function(){  
 options.menuLocation=$(this).context.className;  
 function *getNativePath*() {  
 if(***window***.location.pathname!='/index'){  
 ***window***.location.href=options.origin+'/index';  
 }  
  
 if(***window***.location.host.indexOf('duodian.com')!=-1){  
 options.localTest=false;  
 }else{  
 options.localTest=true;  
 }  
 }  
 function *adLoadUrl*() {  
 if(options.localTest){  
 options.base\_url='http://101.201.122.124:9001/public/menu.jsonp';  
 }else{  
 options.base\_url='http://work.duodian.com/menu.jsonp';  
 }  
 }

function *makeFrame*() {  
 *createFrame*();  
 *initMenu*();  
}  
function *loadMenuData*() {  
 $.ajax({  
 url: options.base\_url, dataType: 'jsonp', async: false, data: {account: options.account}, success: function (result) {  
 if(result!=''&&result!=[]){  
 options.resultMenuData = result;  
 }  
 *setToLocalStorage*();  
 *makeFrame*();  
 }, error: function (err) {  
 ***console***.log(err)  
 }  
 })  
}  
function *init*() {  
 *getNativePath*();  
 *adLoadUrl*();  
 *loadMenuData*();  
}  
*init*();

# 4、后端开始包装从菜单数据（分上下级也就是父亲和儿子）我这里看到的其实也就是上面9001的代码，为什么用9001，闲的呗，业绩就是说 下面这个就是9001了，本地8081也可以测试的哦



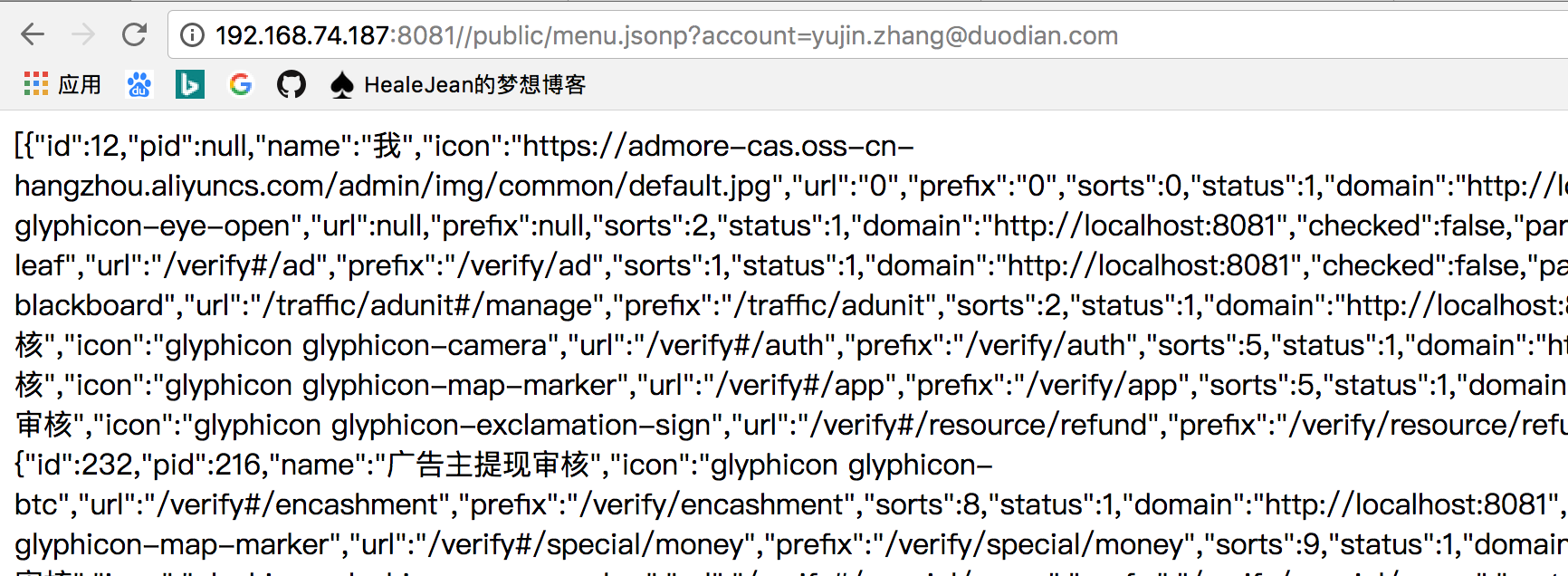
## 1、controller account 为邮箱 yujin.zhang@duodian.com

@Jsonp  
@Controller  
@Profile("prod")  
public class MenuProdController {  
  
 @Resource  
 private LoginService loginService;  
  
 @RequestMapping("menu.jsonp")  
 @ResponseBody  
 public List<SysMenu> menu(String account){  
 return loginService.getAdmUserMenu(account);  
 }  
  
}

## 2、观察实体类

@Entity  
@Table(name = "sys\_menu")  
public class SysMenu implements Serializable {  
  
 public static final int ***STATUS\_ENABLE*** = 1;  
 public static final int ***STATUS\_DISABLE*** = 0;  
  
 private static final long ***serialVersionUID*** = 1463833530703361050L;  
  
 @Id  
 @GeneratedValue(strategy = GenerationType.***IDENTITY***)  
 private Long id;  
 private Long pid; //父级id  
 private String name; //显示名称  
 private String icon; //图标  
 private String url; //跳转地址  
 private String prefix; //权限前缀  
 private Integer sorts; //排序  
 private Integer status;  
 private String domain; //菜单域名前缀  
  
  
 @Transient  
 private Boolean checked = false;  
 @Transient  
 private Boolean parent = false;  
 @Transient  
 private List<SysMenu> children;

## 3、service开始包装，这里我们还是通过debug看吧，浏览器访问



## 1、service访问

@Override  
@Cacheable(value = CacheConstants.***CACHE\_PUBLIC***, key = "'getAdmUserMenuByAccount' + #account")  
public List<SysMenu> getAdmUserMenu(String account) {  
 SysAdminUser adminUser = adminUserDAO.findTopByEmailAndStatus(account,SysAdminUser.***STATUS\_ENABLE***);  
 if (adminUser == null) {  
 return new ArrayList<>();  
 }  
  
 List<SysMenu> userMenu = new ArrayList<>();  
 Set<Long> menuIds = null;  
 if(adminUser.getIsSuper() != null && adminUser.getIsSuper().intValue() == 1){  
 //超级管理员可以查看所有菜单  
 menuIds = new HashSet<>(sysMenuDAO.findMenuIds());  
 }else{  
 menuIds = new HashSet<>(sysAuthRoleMenuDAO.findAdmUserMenuIds(adminUser.getId()));  
 }  
 # 查找到父节，也就是最左侧的菜单，我，设置等  
 List<SysMenu> rootNodes = sysMenuDAO.findByPidIsNullAndStatusOrderBySortsAsc(SysMenu.***STATUS\_ENABLE***);

#添加最高的父菜单，也就是添加好 我，设置  
 for (SysMenu sysMenu : rootNodes){  
 if (menuIds.contains(sysMenu.getId())){  
 userMenu.add(sysMenu);  
 }  
 }

# 上面添加好了父亲菜单，下面开始添加孩子，# 将每一个父级菜单和关于张宇晋账户的所有菜单传入  
 for (SysMenu sysMenu : userMenu){  
 this.setChildren(sysMenu,menuIds);  
 }  
 return userMenu;  
}

## 2、中间查询菜单id的sql

@Query("select id from SysMenu where status = 1")  
public List<Long> findMenuIds();

@Query("select menuId from SysAuthRoleMenu where roleId in (select roleId from SysAuthRoleAdminUser where admId = :admId)")  
public List<Long> findAdmUserMenuIds(@Param("admId") Long admId);

## 3、给父级菜单添加孩子，需要注意的是其实有爷爷，父亲和儿子，所以下面用到的递归

# 将每一个父级菜单和关于张宇晋账户的所有菜单传入  
private void setChildren(SysMenu menu,Set<Long> menuIds){  
 List<SysMenu> children = sysMenuDAO.findByPidAndStatusOrderBySortsAsc(menu.getId(),SysMenu.***STATUS\_ENABLE***);  
 List<SysMenu> userChildren = new ArrayList<>();

#个人认为下面这个for就是多此一举  
 for (SysMenu sysMenu : children){  
 if (menuIds.contains(sysMenu.getId())){  
 userChildren.add(sysMenu);  
 }  
 }  
  
 if (CollectionUtils.isNotEmpty(userChildren)){  
 menu.setParent(true);  
 for (SysMenu child : userChildren){  
 if (!menuIds.contains(child.getId())){  
 continue;  
 }  
 setChildren(child,menuIds);  
 }  
 menu.setChildren(userChildren);  
 } else {  
 menu.setParent(false);  
 }  
}