Vo:值对象 value/ˈvæljuː/ object的缩写 ep:gVo

got:得到 ep:gGot

desc：描述 describe/dɪˈskraɪb/的缩写 ep:gDesc

flag:标记 /flæɡ/ ep:已领取

mgr:管理 Manager/ˈmænɪdʒə(r)/的缩写 ep:gMgr

reward:奖励展示 /rɪˈwɔːd/ ep:gReward

model:模型 /ˈmɒdl/ >> 特效 ep:gModel

skin:皮肤 /skɪn/ >> 装载bg ep:gShinSp

list:列表  /lɪst/

pool:池  /puːl/

exp:经验 experience /ɪkˈspɪəriəns/ 的缩写

item:项 /ˈaɪtəm/

ep:例如 example/ɪɡˈzɑːmpl/ 的缩写

content:内容  /kən'tent/ ep:\_contentSp

parse:解析  /pɑːz/ ep:parseData

gain:奖励  /ɡeɪn/

loss:消耗  /lɒs/  ep:parseGainLossVos

param:参数  /ˌpærəˈm/

cus:参数名前缀 custom /ˈkʌstəm/ 的缩写

cur:当前 current /ˈkʌrənt/ 的缩写

mask:遮罩 /mɑːsk/

dic:字典 dictionary /ˈdɪkʃənri/ 的缩写

dis:距离 distance/ˈdɪstəns/的缩写

time:时间

roll:摇 /rəʊl/

grow:发展 生长 /ɡro?/

game:游戏

dispatcher:触发 /d??sp?t??r/

menu:菜单

gCurInd:当前位置 current+index的缩写

fly:飞

boot:靴子 /bu?t/

Filter:过滤 /?f?lt?r/

attr:属性 attribute /??tr?bju?t/的缩写

enough:足够 /??n?f/

cond:条件 /'k?nd/

temp:临时

find:查找

sort:排序

lib:库 library /?la?breri/的缩写

comm:通用 common/?kɑ?m?n/的缩写

component:组件 /k?m?po?n?nt/

sex:性别 /seks/

role:角色 /ro?l/

info:信息 /??nfo?/

male:男性 /me?l/

female:女性 /?fi?me?l/

panel:面板 /?p?nl/

register:注册，登记/?red??st?(r)/ ep:registerPanelClass()

price:价格

getInstance():获取单例

updata():更新

clearItems():清除选项 >> 自定义方法

init():初始化(一般重写) >> UIEditContainer.as的方法

open():打开界面(一般重写) >> PopPanel.as的方法

close():关闭界面(一般重写) >> PopPanel.as的方法

getStartPos():获取界面开始点(默认为 null) >> PopPanel.as的方法

addDisposeItem(do:DisplayObject):添加到待销毁列表 >> UIEditContainer.as的方法

addNoDisposeItem(do:DisplayObject):添加到不销毁列表 >> UIEditContainer.as的方法

registerStartController():注册需要启动的控制器 >> StartController.as的方法

initController(cusController:Controller, cusGameStartCallBack:Boolean = false):初始化列表 >> registerStartController()

ArrayUtil.as:数组工具

ChangeSceneScaleUtil.as:改变屏幕缩放比例工具

ColorTransformUtil.as:颜色转换工具

ColorUtil.as:颜色工具类

ComponentUtil.as:UI组件工厂

ConfigUtil.as:配置解析工具

DateUtil.as:时间处理工具类

DisplayObjectUtil.as:显示对象工具

GraphicUtil.as:图形绘制工具

HtmlUtil.as:超文本工具类

IDCreatorUtil.as:唯一ID创建工具

LanguageUtil.as:语言包工具类

LayoutUtil.as:排版布局工具

MathUtil.as:计算工具

NumberUtil.as:数字类工具

ScriptUtil.as:脚本工具

TweenLiteUtil.as:缓动工具类

UIEditUtil.as:UI布局解析工具

ResourceName.as:动态加载模块资源名swf路径

consoler.as:测试控制台

CusJsonParser.as:json数据解析器

BackActEvent.as:后台活动事件 ep:BackActEvent.CHARGE\_DISCOUNT\_INFO

GameConfigDataManager.as:配置表解析代理 ep: >> parseConfig(name:String, data:Object) >> else if(name =="act\_charge\_discount\_data") > {ChargeDiscountManager.getInstance().parseData(data);}

HotUpdateItem.as:重载配置item ep: >> onLoadConfigCompleteHandler(evt:Event) >> case HotUpdateManager.CHARGE\_DISCOUNT: > ChargeDiscountManager.getInstance().parseData(\_json);

HotUpdateManager.as:在线热更配置 ep:HotUpdateManager.CHARGE\_DISCOUNT

OpenView.as:打开界面 ep:OpenView.open(uiName:int, openType:uint = 1, openData:Object = null)

UIEditContainer.as:基础UI编辑容器

UIManage.as:面板实例管理器 ep: >> registerPanelClass() >> registerData();registerData(uiName:int, popPanel:Class, res:Array = null, title:String = "", wid:uint = 300, hei:uint = 400, style:uint = 1, closeDispose:Boolean = false, dragAble:Boolean = true, layerType:uint = 1, isFollowStageResize:Boolean = false):void

UINameManager.as:UI名字

RoleManager.as:角色数据管理器

FrameTimerManager.as:FrameTimer对象的管理类,提供静态接口全局调用,您不应该 new FrameTimer实例，应该调用FrameTimerManager.getInstance(),或者

PopPanel.as:二级弹框 继承UIEditContainer

PopPanelManager.as:二级弹框管理器 继承EventDispatcher

Controller.as:所有Controller的基类

GameDispatcher.as:此框架结构中的mvc中的事件发送中心

StartController.as:启动流程控制器

BackActController.as:后台活动控制器

ObjectManager.as:对象池类别 ep:ObjectManager.createSimpleEffect() retun >> SimpleEffect

RightTopView.as:右上角图标面板

SmartCacher.as: 智能Cache, 用于缓存经常使用的对象 \* 如数字图片等

DirQuakeCtrl.as:方向振动控制器

Cookie.as:Cookie类

OperateUtil.as:操作类工具：寻路，传送，打坐等

StateUtil.as:状态机工具

MapUtil.as:地图工具类

TileUtil.as:地图格子信息转换基本工具

FilterWordUtil.as:敏感字过滤

FightUtil.as:战斗相关工具类

CameraUtil.as:摄像机工具类

FindPathUtil.as:寻路工具类

SceneUtil.as:场景工具类

UpdateAttUtil.as:更新属性工具类

AS3.0 关键字：

语句 属性关键字 定义关键字 指令 命名空间 主表达式关键字

break 出现在循环（for、for..in、for each..in、do..while 或 while）内，或出现在与 switch 语句中的特定情况相关联的语句块内。

case 定义 switch 语句的跳转目标。

continue 跳过最内层循环中所有其余的语句并开始循环的下一次遍历，就像控制正常传递到了循环结尾一样。

default 定义 switch 语句的默认情况。

do..while 与 while 循环类似，不同之处是在对条件进行初始计算前执行一次语句。

else 指定当 if 语句中的条件返回 false 时要运行的语句。

for 计算一次 init（初始化）表达式，然后开始一个循环序列。

for..in 遍历对象的动态属性或数组中的元素，并对每个属性或元素执行 statement。

for each..in 遍历集合的项目，并对每个项目执行 statement。

if 计算条件以确定下一条要执行的语句。

label 将语句与可由 break 或 continue 引用的标识符相关联。

return 导致立即返回执行调用函数。

super 调用方法或构造函数的超类或父版本。

switch 根据表达式的值，使控制转移到多条语句的其中一条。

throw 生成或抛出（使用 throw 语句）一个可由 catch 代码块处理或捕获的错误。

try..catch..finally 包含一个代码块，在其中可能会发生错误，然后对该错误进行响应。

while 计算一个条件，如果该条件的计算结果为 true，则会执行一条或多条语句，之后循环会返回并再次计算条件。

with 建立要用于执行一条或多条语句的默认对象，从而潜在地减少需要编写的代码量。

dynamic 指定类的实例可具有在运行时添加的动态属性。

final 指定不能覆盖方法或者不能扩展类。

internal 指定类、变量、常量或函数可用于同一包中的任何调用者。

native 指定函数或方法由 Flash Player 以本机代码的形式实现。

override 指定用一种方法替换继承的方法。

private 指定变量、常量、方法或命名空间仅可供定义它的类使用。

protected 指定变量、常量、方法或命名空间只可用于定义它的类及该类的任何子类。

public 指定类、变量、常量或方法可用于任何调用者。

static 指定变量、常量或方法属于类，而不属于类的实例。

... (rest) parameter 指定函数将接受任意多个以逗号分隔的参数。

class 定义一个类，它允许实例化共享您定义的方法和属性的对象。

const 指定一个常量，它是只能赋值一次的变量。

extends 定义一个作为其它类的子类的类。

function 包含为执行特定任务而定义的一组语句。

get 定义一个 getter，它是一种可像属性一样读取的方法。

set 定义一个 setter，它是一种在公共接口中作为属性出现的方法。

implements 指定一个类可实现一个或多个接口。

interface 定义接口。

namespace 允许您控制定义的可见性。

package 允许您将代码组织为可由其它脚本导入的离散组。

var 指定一个变量。

default xml namespace default xml namespace 指令可将默认的命名空间设置为用于 XML 对象。

import 使外部定义的类和包可用于您的代码。

include 包括指定文件的内容，就像该文件中的命令是调用脚本的一部分一样。

use namespace 使指定的命名空间添加到打开的命名空间集中。

AS3 定义核心 ActionScript 类的方法和属性，将其作为固定属性而非原型属性。

flash\_proxy 定义 Proxy 类的方法。

object\_proxy 定义 ObjectProxy 类的方法。

false 表示 false 的布尔值。

null 一个可以分配给变量的或由未提供数据的函数返回的特殊值。

this 对方法的包含对象的引用。

true 表示 true 的布尔值。

驼峰式命名法：

单词之间不以空格、连接号或者底线连结(例如不应写成:camel case、camel-case或camel\_case形式)。共有两种格式:

1、小驼峰式命名法(lower camel case):

第一个单字以小写字母开始，第二个单字的首字母大写。例如:firstName、lastName。

2、大驼峰式命名法(upper camel case):

每一个单字的首字母都采用大写字母，例如:FirstName、LastName、CamelCase，也被称为 Pascal 命名法。

变种:StudlyCaps，是"驼峰式大小写"的变种。

补充说明，在JAVA中:类名的标识符一般用大驼峰式书写格式，方法和变量的标识符则多用小驼峰式书写格式。

administrator