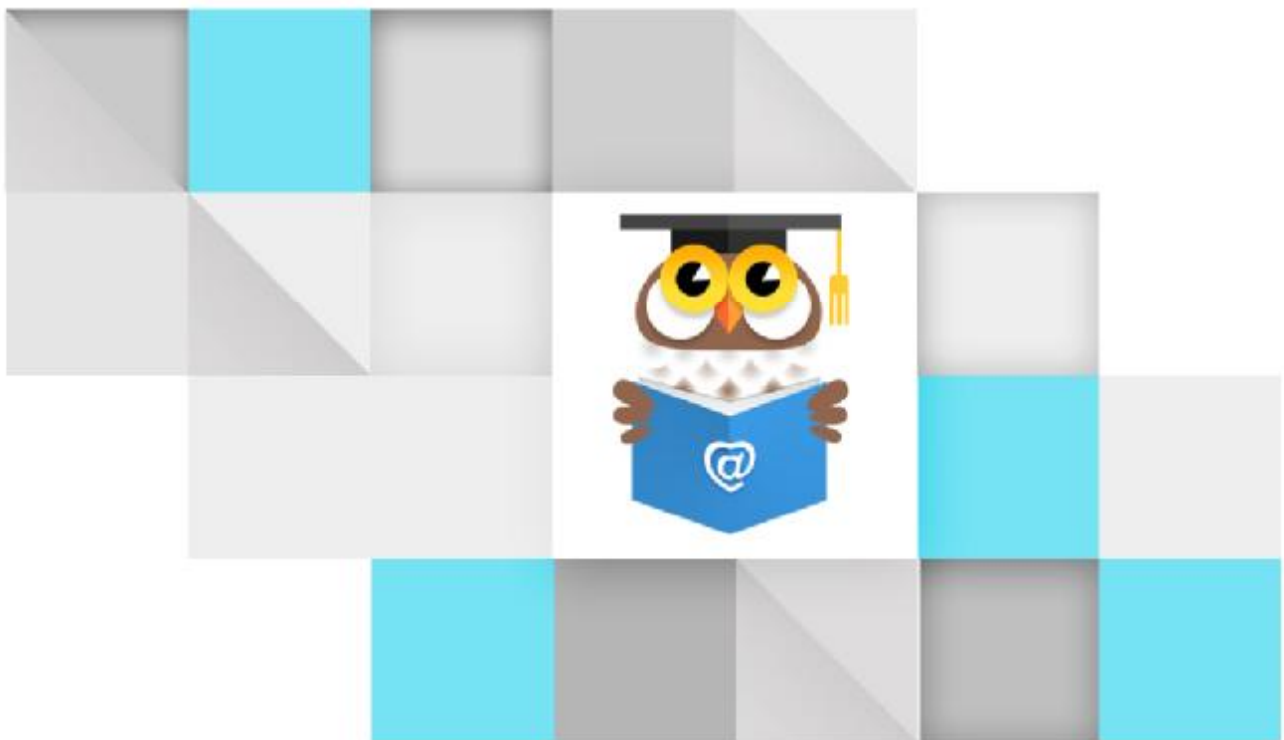


# 消灭泡泡糖 ( Java )

## 实训指导手册

### 实训场景 009 – 在游戏中合理利用日志



**Campus** Solution Group

## 目 录

一、任务编号：PRJ-BU2-JAVA-009 .....	1
1、实训技能 .....	1
2、涉及知识点 .....	1
3、实现效果 .....	1
4、场景说明 .....	2
5、快速开始 .....	4
6、任务 1 – 确认日志环境 .....	5
7、任务 2 – 为泡泡糖业务类创建日志 .....	6
8、任务 3 – 设置日志等级过滤条件 .....	10
9、场景总结 .....	12

## 一、任务编号：PRJ-BU2-JAVA-009

### 1、实训技能

- I Java 面向对象编程技能

### 2、涉及知识点

- I 全局日志记录
- I 日志管理器配置
- I  $\log_4 j$ (\*)

\* 为扩展或体验用知识点

### 3、实现效果



图 3-1

## 4、场景说明

### 1、业务说明：

本场景将利用Java日志技术体验企业级项目开发过程中的日志管理方法，本场景选择的日志框架为使用极为普遍的Log4j框架。

日志管理主要用于实现项目运行中的日志输出功能，如果说游戏界面是用来实现游戏与用户之间对话的平台，那么日志就是游戏与开发人员之间对话的平台，开发人员能根据观察日志信息，找出程序运行过程中出现的BUG及BUG相应的位置，大大方便了程序的测试和维护。

企业级项目开发过程的基础日志管理流程如下：

- 1-1. 使用日志技术监测程序各主要功能类或组件的执行流程。
- 1-2. 设定不同的日志的输出级别，常用的有DEBUG、INFO、WARN和ERROR等。
- 1-3. 设定日志不同的显示介质：控制台显示日志，文件存储日志等。

注意：日志是应用程序调试阶段与部署上线阶段重要的环节，日志管理的优劣直接决定了应用程序维护的成本与代价。

### 2、实现思路：

- 2-1. 搭建Log4j框架所必需的技术环境。
- 2-2. 给游戏业务类、界面类添加各种必要的日志输出信息。
- 2-3. 设置并体验各种日志显示介质：控制台显示日志，文件存储日志。
- 2-4. 通过修改【日志输出级别】管理日志。

### 3、核心组件介绍：

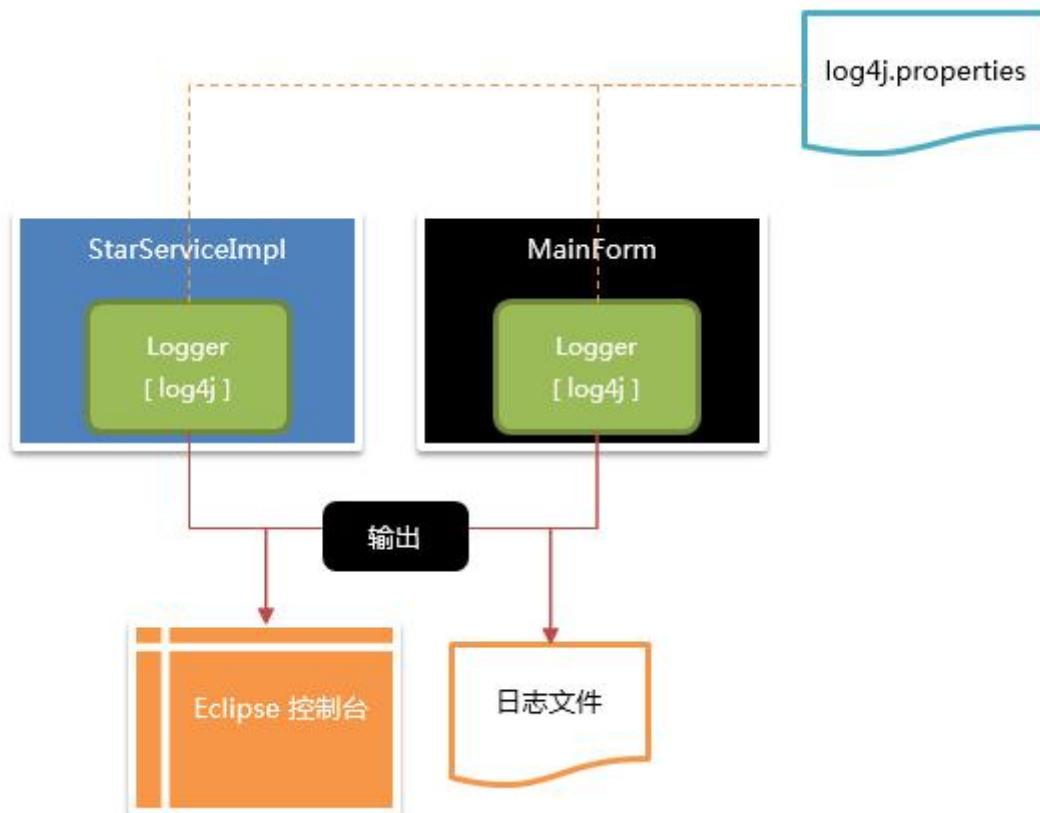


图 4-1

### 3-1. MainForm - 游戏界面类：

负责游戏数据显示、响应用户在界面上的各类操作。

### 3-2. StarServiceImpl – 游戏业务类：

负责游戏相关逻辑计算，例如：泡泡糖移动、消除、分数计算等操作。

### 3-3. Logger – 日志类：

负责向指定【显示介质】输出固定格式、固定等级的日志信息。

### 3-4. log4j.properties – 日志的配置文件：

负责定义日志的一些标准，如：日志的【显示介质】、日志的输出格式、日志的输出级别等。

### 3-5. Eclipse控制台 和 日志文件：

Log4j日志的【显示介质】，日志显示在哪个介质中，由3-4的配置决定。

#### 4、了解更多：

请参考《消灭泡泡糖 - 需求说明文档》。

#### 5、前置条件：

5-1. 前置场景：PRJ-BU2-JAVA-008 – 在游戏中合理利用异常。

5-2. 必备知识与技能：

5-2.1. Java面向对象编程技能（类、方法调用、日志的使用）。

## 5、快速开始

#### 1、开发环境：

1-1. Oracle JDK8.x 以上版本

1-2. Eclipse Luna ( 4.4.x ) 以上版本

1-3. 工程包：PRJ\_BU2\_JAVA\_009

#### 2、进入开发环境：

详见SPOC平台上《PRJ-BU2-JAVA-009 前置任务：进入开发环境》



图 5-1

## 6、任务 1 – 确认日志环境

### 1、任务描述：

为游戏程序添加日志管理的开发环境，包括：

- 1-1. 添加使用日志的核心jar包【log4j-1.2.15.jar】。
- 1-2. 添加使用日志的核心配置文件【log4j.properties】。

### 2、推荐步骤：

- 2-1. 定位到项目工程下的文件夹：resource。
- 2-2. 把【log4j-1.2.15.jar】文件复制到【lib】文件夹下。
- 2-3. 引用该日志的jar包。

+ **提示：** 软件程序引用第三方jar包的方法：右键【lib】文件夹下的【log4j-1.2.15.jar】文件，选择【Build Path】 -> 【Add to Build Path】。

#### + 业务说明

- 1 ) jar包说明： log4j-1.2.15.jar负责输出、管理、显示日志，项目如果需要增加日志管理功能，就必须引用该jar包来完成日志的输出。
- 2 ) 第三方jar包：企业级项目开发过程中，并非所有的代码都需要自己完成，引用其他团队或公司开发的成熟的jar包，就被称之为引用第三方jar包。
- 3 ) 当前软件行业的主流开发模式是：尽可能的选择由第三方公司研发的各类功能jar包，软件开发人员主要负责调用jar包中包含的功能类，来以此降低系统业务的开发难度，提升开发效率。

### 2-4. 添加日志的核心配置文件

- 2-4.1. 把【resource】文件夹下的【log4j.properties】文件复制到【src】文件夹下

### + 业务说明

1) 日志配置文件：在使用log4j输出日志信息时，我们需要控制日志信息输出的目的地、输出格式等，还需要控制日志信息的输出优先级。这些需求，需要通过文件来进行相应的配置。

2) 文件主要配置属性说明：

**log4j.rootLogger：**

2-1. 指定日志输出的级别，本场景中的五个级别分别为DEBUG、INFO、WARN、ERROR和FATAL，优先级为DEBUG < INFO < WARN < ERROR < FATAL。

2-2. 设置日志文件的显示介质（数据来自于appenderName）

**log4j.appender.appenderName：**

2-3. 日志显示介质名，本场景日志显示介质，可选【控制台】A1；【本地文件】R；

注意：appenderName为显示介质的代称，可自定义。

**log4j.appender.appenderName.layout.ConversionPattern：**

2-4. 指定日志输出的格式。

### 3、验证与测试：

3-1. 确保log4j-1.2.15.jar文件已存储于lib目录下。

3-2. 确保log4j-1.2.15.jar已被系统引用。

3-3. 确保log4j.properties文件已存储于src目录下。

### 4、参考答案：

无

## 7、任务 2 – 为泡泡糖业务类创建日志

### 1、任务描述：



- 1-1. 为业务类StarServiceImpl增加日志类对象。
- 1-2. 分别为公共函数tobeClearedStars、createStars添加【调试】日志。
- 1-3. 根据业务需要为业务类StarServiceImpl增加【信息】日志、【警告】日志。
- 1-4. 将日志信息输出到：【Eclipse 控制台】显示。
- 1-5. 将日志保存于日志文件中。

## 2、推荐步骤：

- 2-1. 定位到业务类：cn.campsg.practical.bubble.service.StarServiceImpl。
- 2-2. 创建一个日志类对象。

### + 提示

- 1) 可通过以下方式获得指定类的日志对象：

Logger.getLogger(业务类的Class对象)

- 2) 日志对象主要通过以下方法输出相应级别的信息：

fatal方法：【FATAL级别】

error方法：【ERROR级别】

warn方法：【WARN级别】

info方法：【INFO级别】

debug方法：【DEBUG级别】

- 2-3. 定位到当前类的createStars方法，添加日志信息：

- 2-3.1. 在方法开始处输出【DEBUG级】信息，日志内容如下：

开始通过createStars函数创建10\*10泡泡糖。

- 2-3.2. 在循环结束后输出【INFO级】信息，日志内容如下：

10\*10泡泡糖创建结果：完整的泡泡糖集合StarList（数据取自于循环）。

2-3.3. 在方法返回前输出【DEBUG级】信息，日志内容如下：

10\*10泡泡糖创建完毕

2-4. 定位到当前类的lookupByPath方法，添加日志信息：

2-4.1. 当左侧为相同类型泡泡糖时，即“star.getType() == type”判断内，添加

【DEBUG级】，日志内容如下：

被点击的泡泡糖：被点击的泡泡糖（数据取自于参数）左侧存在同色泡泡糖。

2-4.2. 同样，当右、上和下侧为相同类型泡泡糖时，也添加【DEBUG级】信息。

2-5. 定位到当前类的tobeClearedStars方法，添加日志信息：

2-5.1. 在方法开始处添加【DEBUG级】信息，日志内容如下：

开始通过tobeClearedStars函数获取被点击

2-5.2. 添加【INFO级】信息，显示当前点击的泡泡糖的相关信息，日志内容如下：

被点击的泡泡糖为：被点击的泡泡糖（数据取自于参数）

2-5.3. 当没有待消除泡泡糖时，添加【WARN级】信息，警告当前泡泡糖周围没有可消

除泡泡糖，日志内容如下：

被点击的泡泡糖:[ 被点击的泡泡糖（数据取自于参数）]周边没有待消除的泡泡糖

2-5.4. 在方法返回前添加【INFO级】信息，显示待消除的泡泡糖列表内容，日志内容

如下：

待清除的泡泡糖内存列表：通过程序获取的待清除泡泡糖集合StarList

2-5.5. 添加【DEBUG级】信息，提示获取待消除列表的方法执行完毕，日志内容如下：

成功通过tobeClearedStars获取待消除的泡泡糖

+ 提示：日志输出内容中：

1) 蓝色字体：输出内容。

2) **红色字体**：变量、对象、集合。

### 3、验证与测试：

3-1. 定位到程序入口类：cn.campsg.practical.bubble.MainClass。

3-2. 运行项目工程并点击游戏界面的任意泡泡糖，观察控制台是否有日志内容输出：



图 7-1

3-3. 更改日志输出目的地：本地文件

3-3.1. 定位到log4j的配置文件中。

3-3.2. 修改【log4j.rootLogger】属性，把控制台显示日志改为本地文件显示日志。

+ **提示**：该配置文件中有两个代称：A1、R，通过配置文件的注释可知：

1) A1是控制台输出的代称

2) R是本地文件输出的代称

3-4. 在配置文件中，指定了日志文件的保存位置为：C:/log/bubble.log。

3-5. 运行项目工程并点击游戏界面的任意泡泡糖。

3-6. 到3-4的路径下找到日志文件并打开，能看到相应的日志信息：

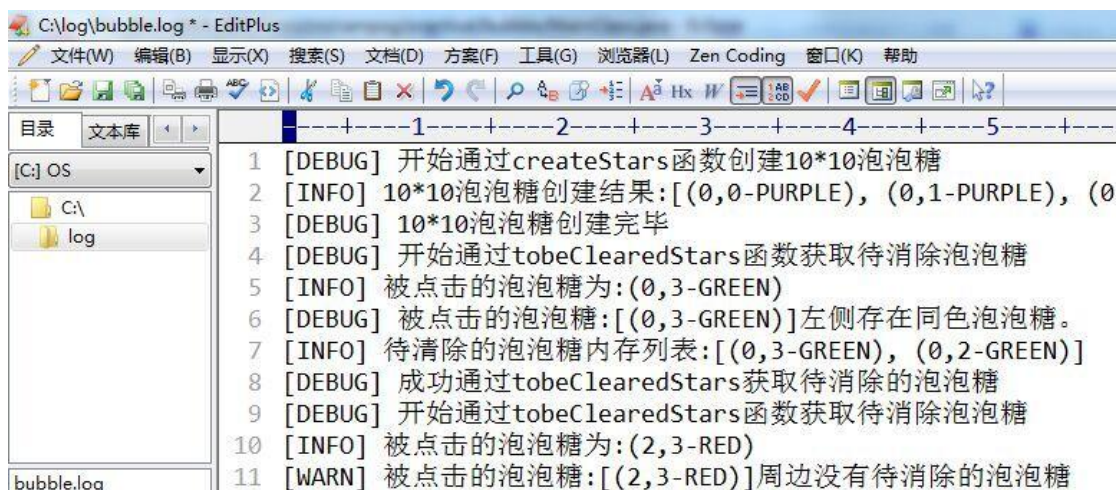


图 7-2

## 8、任务 3 – 设置日志等级过滤条件

### 1、任务描述：

- 1-1. 为方便测试，任务将为每个界面泡泡糖设置可显示的坐标文字（见完成效果）。
- 1-2. 为区分调试状态和发布状态，可显示的坐标文字仅在调试状态可见。
- 1-3. 通过日志等级控制技术完成本场景任务。

### 2、推荐步骤：

- 2-1. 定位到业务类：cn.campsg.practical.bubble.MainForm。
- 2-2. 为窗体类创建一个日志对象。
- 2-3. 定位到方法initGameStars并找到相应注释。
- 2-4. 调用showStarsCoordinate方法显示界面泡泡糖坐标文字。
- 2-5. 确保显示界面泡泡糖坐标文字的方法仅在调试模式下有效：

2-5.1. 判断当前的日志输出级别是否为【DEBUG级】。

#### + 提示

1) 判断日志输出级别：通过isDebugEnabled方法进行判断。

2) showStarsCoordinate方法：在游戏界面上显示泡泡糖的坐标：

鼠标在游戏【关卡】菜单上点下并保持时，可显示界面泡泡糖坐标文字，当松开鼠标按键时，隐藏泡泡糖坐标。



### 3、验证与测试：

3-1. 定位配置文件：src/log4j.properties。

3-2. 将log4j.rootLogger的日志等级修改为：DEBUG。

3-3. 定位到程序入口类：cn.campsg.practical.bubble.MainClass。

3-4. 运行该项目，鼠标在游戏【关卡】菜单上点下并保持，观察是否能显示泡泡糖坐标：



图 8-1

3-5. 再次定位src/log4j.properties，修改【日志输出级别】为【INFO级】，再运行项目

工程，此时点击菜单处将不会再显示泡泡糖坐标：

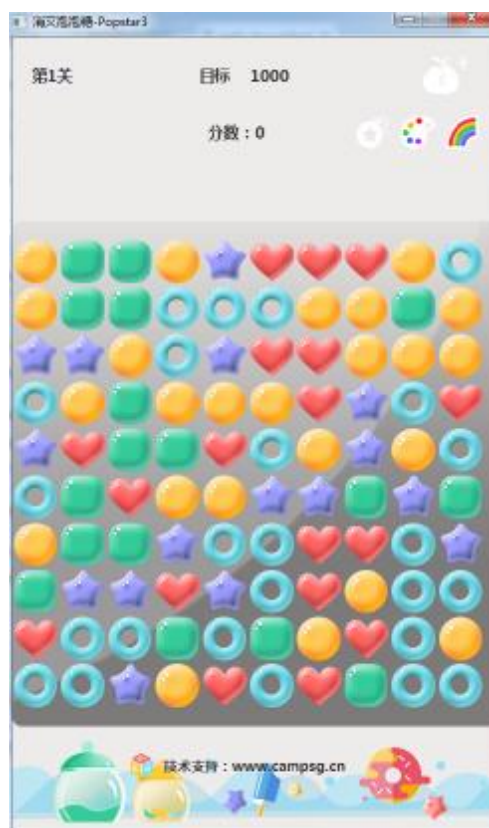


图 8-2

## 9、场景总结

Q1. 在您的项目中，您使用何种日志框架？日志的作用是什么？

1. 项目中使用的是Log4j日志框架。

2. Log4j配置简单，它可以分别配置日志等级、日志输出格式、日志输出容器（文件、控制台）。

3. 日志是软件系统的重要组成部分，当系统测试完成上线或部署到用户现场后，系统发生的任何错误都不可能再通过调试程序解决，只能通过系统级日志来跟踪错误情况，判断错误的原因。

扩展1：任何系统都应具有完善的日志系统，在企业级系统，日志打印一般由AOP技术负责。

扩展2：日志不但需要打印还需要备份，大型电商公司为了分析用户行为都以日志作为数据基础。