



链滴

Latke 学习历程

作者: [hunhuang](#)

原文链接: <https://ld246.com/article/1599797515053>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



Latke初始化

在Service类中

首先获取当前的路径(包名)

在执行init()方法

```
private static Properties localProps;

private static Properties latkeProps;

public static synchronized void init() {
    // inited : 标识符, 用来检测是否进行过初始化
    if (inited) {
        return;
    }
    inited = true;
    LOGGER.log(Level.TRACE, "Initializing Latke");

    // 读取 latke.properties文件内容, 获取后赋给上面的静态变量 latke的配置信息
    loadLatkeProps();
    // 读取 local.properties文件内容, 获取后赋给上面的静态变量 数据库的配置信息
    loadLocalProps();

    //检测latke的运行环境(可以在latke.properties中配置)
    if (null == runtimeMode) {
        //判断在latke中有没有配置运行环境
        final String runtimeModeValue = getLatkeProperty("runtimeMode");
        if (null != runtimeModeValue) {
            //获取到后使用获取的运行环境
        }
    }
}
```

```

        runtimeMode = RuntimeMode.valueOf(runtimeModeValue);
    } else {
        LOGGER.log(Level.TRACE, "Can't parse runtime mode in latke.properties, default to
PRODUCTION]");
        //没有获取到(没有配置)默认使用生产环境
        runtimeMode = RuntimeMode.PRODUCTION;
    }
}
//log日志的信息 如果你的环境是开发环境他会告诉你现在是开发环境,还有一个生产环境可以
择
    if (Latkes.RuntimeMode.DEVELOPMENT == getRuntimeMode()) {
        LOGGER.warn("!!!!Runtime mode is [" + Latkes.RuntimeMode.DEVELOPMENT + "], ple
se make sure configured it with ["
        + Latkes.RuntimeMode.PRODUCTION + "] in latke.properties if deployed on pro
duction environment!!!!");
    } else {
        //告诉你现在是生产环境
        LOGGER.log(Level.DEBUG, "Runtime mode is [{}]", getRuntimeMode());
    }
//获取数据库配置 告诉你用的是什么数据库
final RuntimeDatabase runtimeDatabase = getRuntimeDatabase();
LOGGER.log(Level.DEBUG, "Runtime database is [{}]", runtimeDatabase);
//获取缓存配置 告诉你用的是什么缓存(本地/Redis/没有)
final RuntimeCache runtimeCache = getRuntimeCache();
LOGGER.log(Level.INFO, "Runtime cache is [{}]", runtimeCache);
// Locale : 国际化对象 下面用来指定默认的语言类型为中文
Locale.setDefault(Locale.SIMPLIFIED_CHINESE);
//获取所有的类
final Collection<Class<?>> beanClasses = Discoverer.discover(Latkes.getScanPath());
//添加进容器中
BeanManager.start(beanClasses);
//初始化完成
LOGGER.log(Level.INFO, "Initialized Latke");
}

```