**Grails用法**

# 为什么要用Grails

<http://www.grails.org/>

1. 采用groovy 开发，基于springboot+ hibernate ，groovy 语言层面运行效率稍微差点，但开发效率比java快很多。具体可以参见groovy 语言介绍。

最显然的一个特征是：

bean?.age 这种写法 ?. 可以避免java NullPointer Exception .

1. java 开发人员能非常快速上手。
2. 简单的增删查改页面，基本上可以完全自动生成。
3. 对数据库操作，增强了hibernate .
4. 对于类 Person (id,name,age),可以直接使用 Person.finByName(),Person. findByNameIlikeAndAge 非常强大。
5. 同时grails 还提供了丰富的插件，如

spring-security ：权限控制

migration: 数据库版本管理，写好实体类，可以自动完成与数据库的同步更新。

# 如何使用

1. **下载grails 安装包，配置环境变量。**
2. Grails 提供了一系列内置命令，可以直接生成代码或者运行项目：

[create-app](http://docs.grails.org/latest/ref/Command%20Line/create-app.html) 穿件一个新的空应用。

[create-controller](http://docs.grails.org/latest/ref/Command%20Line/create-controller.html) 新建 spring-mvc 的controller

[create-domain-class](http://docs.grails.org/latest/ref/Command%20Line/create-domain-class.html) 创建数据库实体

[create-interceptor](http://docs.grails.org/latest/ref/Command%20Line/create-interceptor.html) 创建web 拦截器

[create-service](http://docs.grails.org/latest/ref/Command%20Line/create-service.html) 创建service层

[create-taglib](http://docs.grails.org/latest/ref/Command%20Line/create-taglib.html) 创建tag库

….

1. **一般使用流程如下：**
2. 创建entity类(表结构)
3. 用migration 同步表结构。
4. 用命令generate-all 生成controller 和 gsp页面。
5. 修改相关部分代码，满足自己需要。