## 日志分为：

默认日志：

# 日志文件名，比如：roncoo.log，或者是 /var/log/roncoo.log

logging.file=roncoo.log

# 日志级别配置，比如： logging.level.org.springframework=DEBUG

logging.level.\*=info

logging.level.org.springframework=DEBUG

Lomback:

Log4g2:比log4g好

性能比较：Log4J2 和 Logback 都优于 log4j（不推荐使用）

配置方式：Logback最简洁，spring boot默认，推荐使用

## 启动

**java -jar myapp.jar** **--spring.profiles.active=dev**

## web 应用开发-Servlets, Filters, listeners

方法一：通过注册 ServletRegistrationBean、 FilterRegistrationBean 和

ServletListenerRegistrationBean 获得控制

/\*\*

\* 自定义 servlet

\*

\* @author wujing

\*/

public class CustomServlet extends HttpServlet {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

System.out.println("servlet get method");

doPost(request, response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

System.out.println("servlet post method");

response.getWriter().write("hello world");

}

}

/\*\*

\* 自定义 filter

\*

\* @author wujing

\*/

Spring Boot基础教程 作者：冯永伟

2

龙果学院：http://www.roncoo.com

public class CustomFilter implements Filter {

@Override

public void init(FilterConfig filterConfig) throws ServletException {

System.out.println("init filter");

}

@Override

public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain)

throws IOException, ServletException {

System.out.println("do filter");

chain.doFilter(request, response);

}

@Override

public void destroy() {

System.out.println("destroy filter");

}

}

/\*\*

\* 自定义 listener

\*

\* @author wujing

\*/

public class CustomListener implements ServletContextListener {

@Override

public void contextInitialized(ServletContextEvent sce) {

System.out.println("contextInitialized");

}

@Override

public void contextDestroyed(ServletContextEvent sce) {

System.out.println("contextDestroyed");

}

}

注册 bean

@Bean

public ServletRegistrationBean servletRegistrationBean() {

Spring Boot基础教程 作者：冯永伟

3

龙果学院：http://www.roncoo.com

return new ServletRegistrationBean(new CustomServlet(), "/roncoo");

}

@Bean

public FilterRegistrationBean filterRegistrationBean() {

return new FilterRegistrationBean(new CustomFilter(), servletRegistrationBean());

}

@Bean

public ServletListenerRegistrationBean<CustomListener> servletListenerRegistrationBean() {

return new ServletListenerRegistrationBean<CustomListener>(new CustomListener());

}

方法二：通过实现 ServletContextInitializer 接口直接注册

implements ServletContextInitializer {

@Override public void onStartup(ServletContext servletContext) throws ServletException { servletContext.addServlet("customServlet", new CustomServlet()).addMapping("/roncoo"); servletContext.addFilter("customFilter", new CustomFilter()) .addMappingForServletNames(EnumSet.of(DispatcherType.REQUES T), true, "customServlet"); servletContext.addListener(new CustomListener()); }

方法三：在 SpringBootApplication 上使用@ServletComponentScan 注解后，直接通过@WebServlet、

@WebFilter、@WebListener 注解自动注册