/\*\*

\* Profile：

\* Spring为我们提供的可以根据当前环境，动态的激活和切换一系列组件的功能；

\*

\* 开发环境、测试环境、生产环境；

\* 数据源：(/A)(/B)(/C)；

\*

\*

\* @Profile：指定组件在哪个环境的情况下才能被注册到容器中，不指定，任何环境下都能注册这个组件

\*

\* 1）、加了环境标识的bean，只有这个环境被激活的时候才能注册到容器中。默认是default环境

\* 2）、写在配置类上，只有是指定的环境的时候，整个配置类里面的所有配置才能开始生效

\* 3）、没有标注环境标识的bean在，任何环境下都是加载的；

\*/

@PropertySource("classpath:/dbconfig.properties")

@Configuration

public class MainConfigOfProfile implements EmbeddedValueResolverAware{

@Value("${db.user}")

private String user;

private StringValueResolver valueResolver;

private String driverClass;

@Bean

public Yellow yellow(){

return new Yellow();

}

@Profile("test")

@Bean("testDataSource")

public DataSource dataSourceTest(@Value("${db.password}")String pwd) throws Exception{

ComboPooledDataSource dataSource = new ComboPooledDataSource();

dataSource.setUser(user);

dataSource.setPassword(pwd);

dataSource.setJdbcUrl("jdbc:mysql://localhost:3306/test");

dataSource.setDriverClass(driverClass);

return dataSource;

}

@Profile("dev")

@Bean("devDataSource")

public DataSource dataSourceDev(@Value("${db.password}")String pwd) throws Exception{

ComboPooledDataSource dataSource = new ComboPooledDataSource();

dataSource.setUser(user);

dataSource.setPassword(pwd);

dataSource.setJdbcUrl("jdbc:mysql://localhost:3306/ssm\_crud");

dataSource.setDriverClass(driverClass);

return dataSource;

}

@Profile("prod")

@Bean("prodDataSource")

public DataSource dataSourceProd(@Value("${db.password}")String pwd) throws Exception{

ComboPooledDataSource dataSource = new ComboPooledDataSource();

dataSource.setUser(user);

dataSource.setPassword(pwd);

dataSource.setJdbcUrl("jdbc:mysql://localhost:3306/scw\_0515");

dataSource.setDriverClass(driverClass);

return dataSource;

}

@Override

public void setEmbeddedValueResolver(StringValueResolver resolver) {

// TODO Auto-generated method stub

this.valueResolver = resolver;

driverClass = valueResolver.resolveStringValue("${db.driverClass}");

}