# SpringBoot[bttel+bettlSQL环境配置]

<http://ibeetl.com/guide/#beetlsql> 中文文档

|  |
| --- |
| <dependency>  <groupId>com.ibeetl</groupId>  <artifactId>beetlsql</artifactId>  <version>2.10.30</version>  </dependency>  <!-- https://mvnrepository.com/artifact/org.beetl/beetl-core -->  <dependency>  <groupId>org.beetl</groupId>  <artifactId>beetl-core</artifactId>  <version>2.2.3</version>  </dependency> |

# SpringBoot [bettelSQL初级实现]

## 核心对象SQLManager

|  |
| --- |
| **public** **static** **final** String ***driver*** = "com.mysql.jdbc.Driver";  **public** **static** **final** String ***url*** = "jdbc:mysql://localhost:3306/mem?useUnicode=true&characterEncoding=utf-8";  **public** **static** **final** String ***userName*** = "root";  **public** **static** **final** String ***password*** = "root";  /\*\*  \* 获取到SQLManager 核心对象  \*  \* **@param** driver  \* **@param** url  \* **@param** userName  \* **@param** password  \* **@return**  \*/  **public** **static** SQLManager getSQLManager(String driver, String url, String userName, String password) {  // 连接数据库  ConnectionSource source = ConnectionSourceHelper.*getSimple*(driver, url, userName, password);  DBStyle mysql = **new** MySqlStyle();  // sql语句放在classpagth的/sql 目录下  SQLLoader loader = **new** ClasspathLoader("/sql");  // 数据库命名跟java命名一样，所以采用DefaultNameConversion，还有一个是UnderlinedNameConversion，下划线风格的，  UnderlinedNameConversion nc = **new** UnderlinedNameConversion();  // 创建一个SQLManager,DebugInterceptor  SQLManager sqlManager = **new** SQLManager(mysql, loader, source, nc);  **return** sqlManager;  } |

## CRUD+POJO生成+SQL生成

|  |
| --- |
| /\*\*  \* 增  \*  \* **@param** manager  \*/  **public** **void** insertByManager(SQLManager manager) {  User user = **new** User();  user.setAge(13);  manager.insert(User.**class**, user);  }  /\*\*  \* 根绝ID删除  \*  \* **@param** manager  \*/  **public** **void** deleteByManager(SQLManager manager) {  manager.deleteById(User.**class**, 1);  }  /\*\*  \* 根绝对象删除  \*  \* **@param** manager  \*/  **public** **void** deketeObjectByManager(SQLManager manager) {  User user = **new** User();  user.setAge(13);  manager.deleteObject(user);  }  /\*\*  \* query条件查找  \*  \* **@param** manager  \*/  **public** **void** queryByManager(SQLManager manager) {  User query = **new** User();  query.setName("xiandafu");  List<User> list = manager.template(query);  }  /\*\*  \* 根绝数据库表生成POJO  \*  \* **@param** manager  \*/  **public** **static** **void** genderCodeByBeetl(SQLManager manager) {  **try** {  manager.genPojoCodeToConsole("user");  } **catch** (Exception e) {  e.printStackTrace();  }  }  /\*\*  \* 根绝数据库表生成简单的BttelSQL.  \*  \* **@param** manager  \*/  **public** **static** **void** genderBeetlSQL(SQLManager manager) {  **try** {  manager.genSQLTemplateToConsole("user");  } **catch** (Exception e) {  // **TODO** Auto-generated catch block  e.printStackTrace();  }  } |

