转换工具使用说明

# 项目文件说明

* 项目git地址：
* **test** ：文件夹下存放代码转换相关脚本
* okg文件夹存放图谱构建脚本，其中**knowledgeGraph.py**为图谱构建执行脚本
* main.py用于执行代码转换，每次执行**main.py**文件得到的转换结果项目存在**test/tmp**临时文件夹下
* unitTest.py：用于进行模型的单元测试
* conversion中存有一些提供给迁移程序使用的辅助迁移函数。
* dao：文件夹存放图谱数据，用于图谱构建
* **origin**：存放需要转换的原Pytorch框架的十个模型。
* Image\_Classification：原Pytorch框架下的8个分类模型，用于转换。模型训练方式见项目下readme.md文件。
* ISTD：原Pytorch框架下的2个分割模型，用于转换。模型训练方式见项目下readme.md文件。
* **target：**转换后的Mindspore项目，用于示例与单元测试。模型训练方式见项目下readme.md文件。
* **datasets**：存放转换前后两种模型所需数据集，
* **result** ：存放十个模型在pytorch与mindspore框架下的训练结果，包括训练log与部分checkpoint
* requirements.txt：项目依赖的python模块

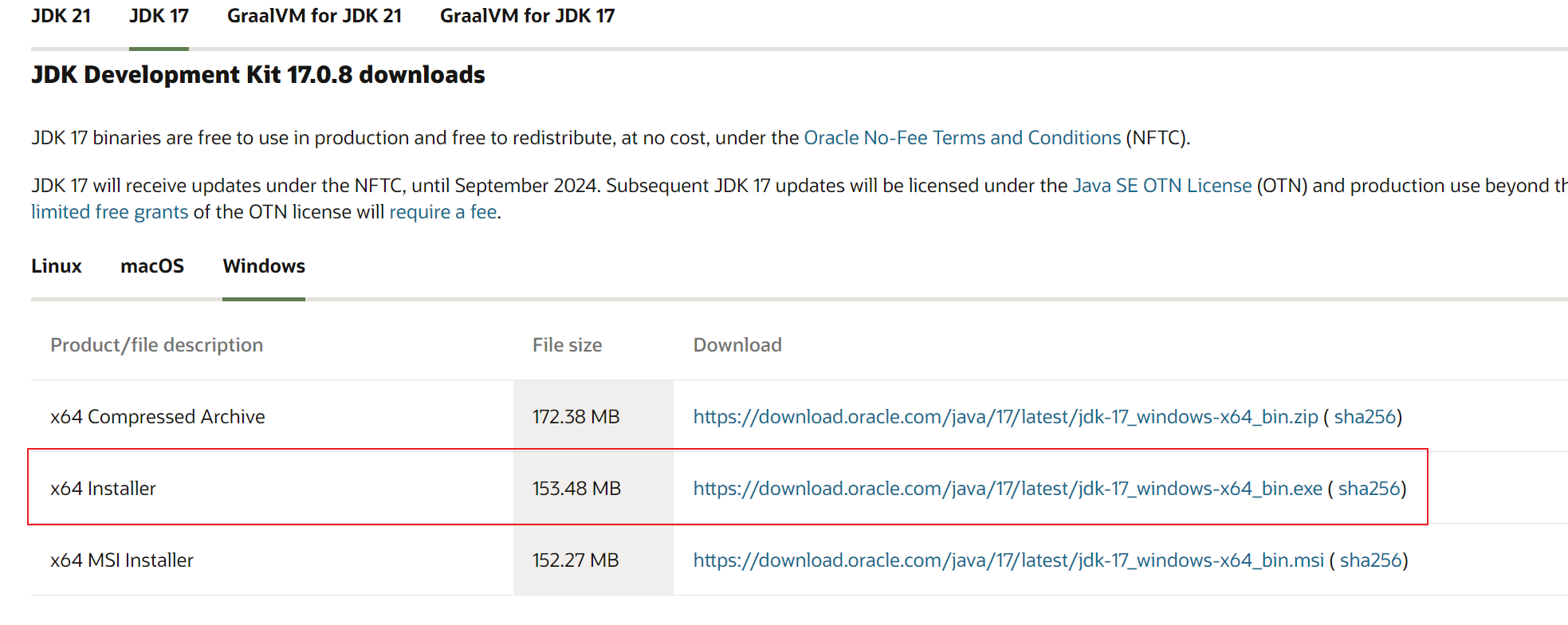
# 环境搭建

## 1.1 图谱环境搭建：Windows

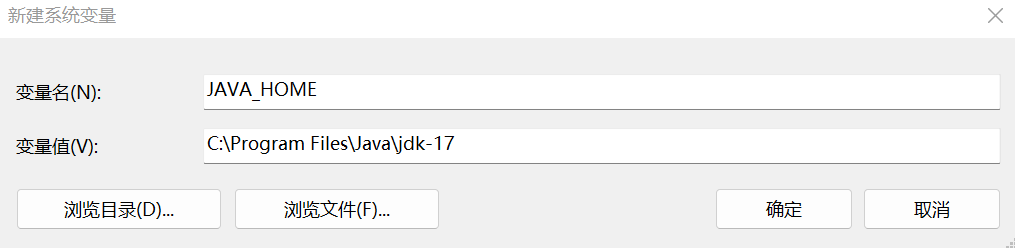
### JDK安装

* Neo4j是基于Java的图形数据库，运行Neo4j需要启动JVM进程，因此必须安装JAVA SE的JDK
* 步骤：

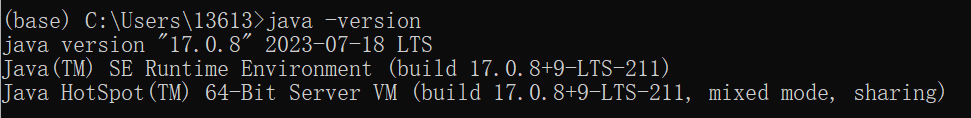
1. 从Oracle官方网站下载 Java SE JRE：[https://www.oracle.com/java/technologies/downloads/，推荐使用Java 17](https://www.oracle.com/java/technologies/downloads/，推荐使用Java%2017)



1. 配置新建JAVA\_HOME变量，变量值填写jdk的安装目录

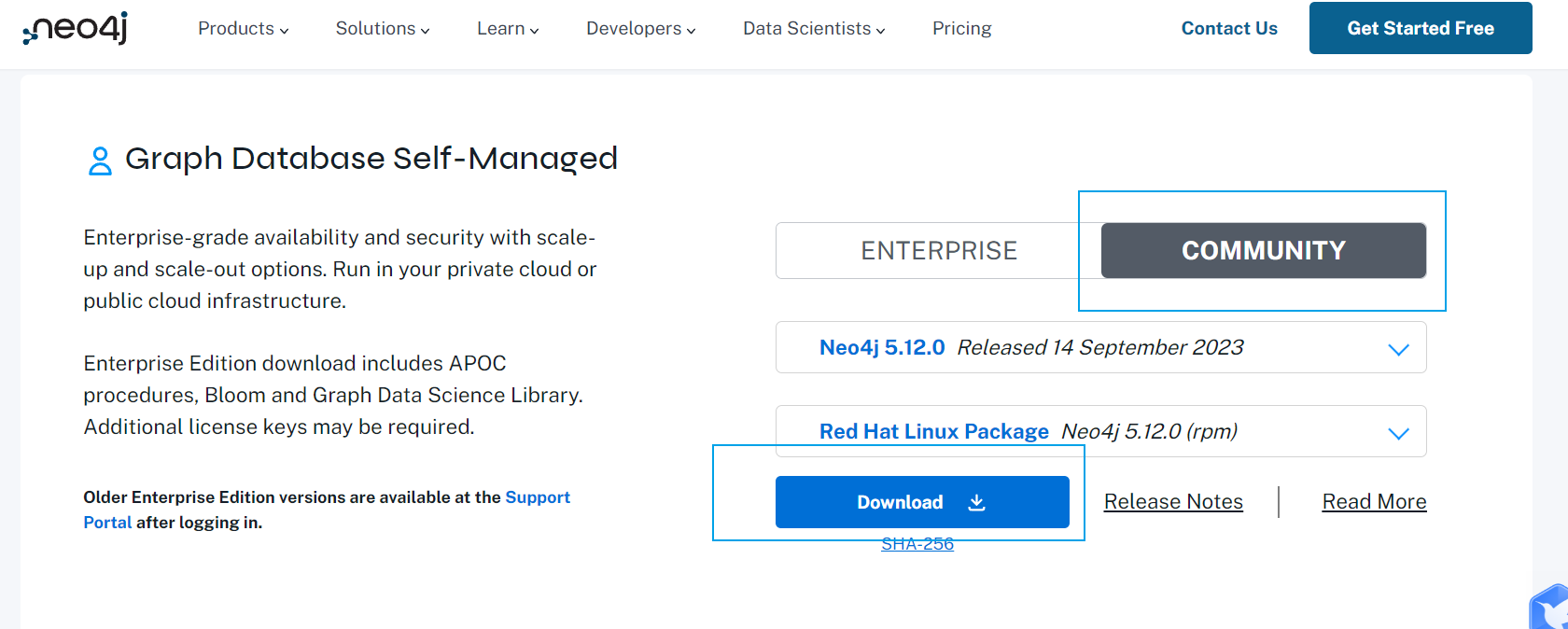


1. 编辑Path变量，在Path变量值的最后输入：%JAVA\_HOME%\bin;%JAVA\_HOME%\jre\bin;
2. 检查是否配置成功：java -version

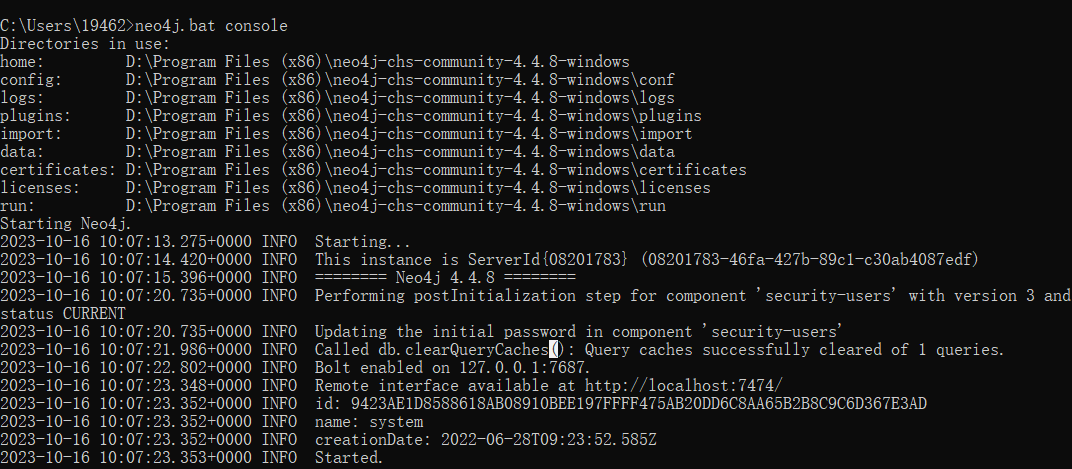


### Neo4j安装

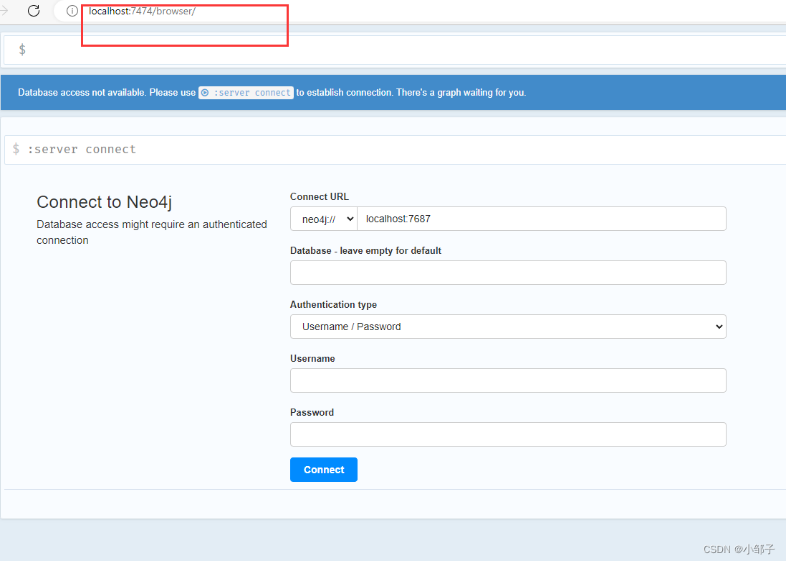
1. Neo4j官网下载：[https://neo4j.com/deployment-center/#community](https://neo4j.com/deployment-center/" \l "community)



1. 解压
2. 创建主目录环境变量NEO4J\_HOME，变量值设置为主目录路径。如D:\sorftware\neo4j-community-5.10.0的环境变量。
3. 编辑系统变量区的Path，点击新建，然后输入 **%NEO4J\_HOME%\bin**
4. 启动neo4j：以系统用户身份通过命令行neo4j.bat console运行Neo4j。



1. Neo4j集成浏览器：浏览器打开：<http://localhost:7474/>。默认的用户名和密码均为neo4j。登录后需重置密码，如设为八位12345678，记住密码

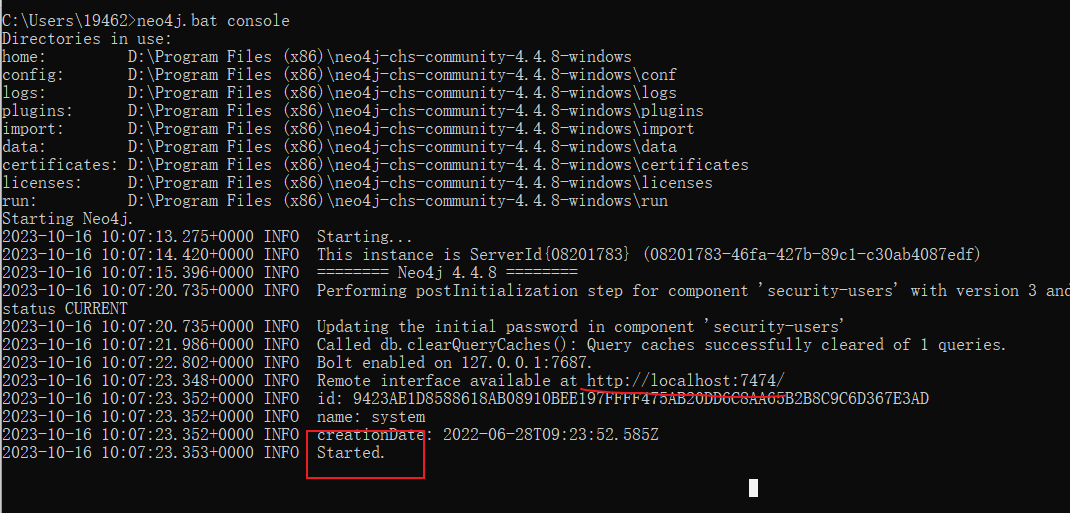


## 1.2 Python环境

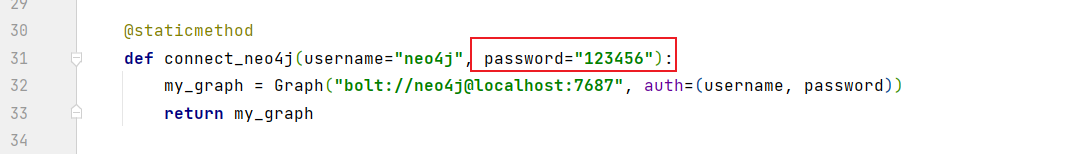
1. 安装python
2. 创建conda新环境
   1. conda create -n okgct python=3.7
   2. conda activate okgct
3. 进到项目目录，执行pip install -r requirements.txt 安装项目所需依赖包
4. 注意安装torch1.8.1，mindspore2.2.0, 根据cuda版本，参考：torch==1.13.0+cu116 torchvision==0.14.0+cu116 torchaudio==0.13.0

# 运行转换程序

1. **在执行转换程序前需保证neo4j处在运行状态**：运行neo4j：终端输入指令：**neo4j.bat console**。出现 started 即运行成功 ，注意需要一直后台运行。



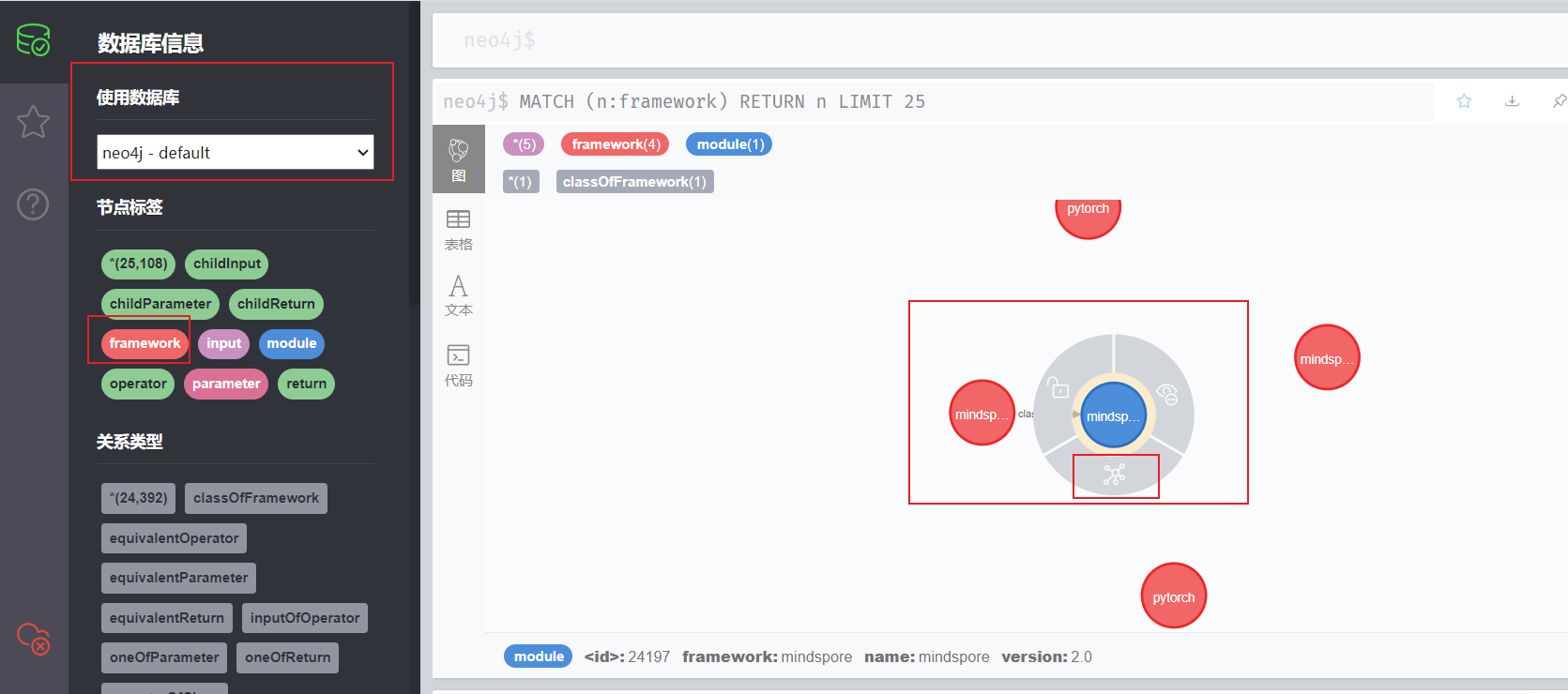
2. 修改okgct/test/ okg/knowledgeGraph.py文件类knowledgeGraph的函数connect\_neo4j为搭建neo4j环境时在登陆<http://localhost:7474/> 时修改的用户名和密码【可在浏览器登录数据库验证用户名和密码】

 3. 运行 okgct/ test/okg/knowledgeGraph.py构建图谱：

cd okgct/ test/okg

python knowledgeGraph.py

执行完可打开<http://localhost:7474/> 重新登录查看图谱是否构建成功。（选中左方节点标签点击节点即可在右方查看节点，点击节点可展开查看节点连接信息，如下图）



1. 执行okgct/test/main.py转换okgct/origin项目，转换结果位于okgct/test/tmp:

cd okgct/ test

python main.py

1. 将转换后的mindspore项目okgct/test/tmp，复制到mindspore环境训练。

# 3. 模型训练

两个框架下模型的训练说明见转换前后**readme.md**文件。原始数据集位于okgct/datasets下，训练时需根据readme文件解压到对应位置

# 4. 单元测试

## 相关文件

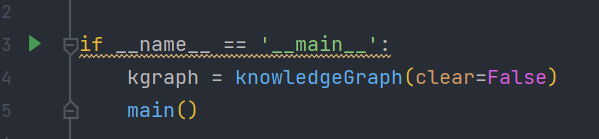
* okgct/test/unitTest.py ：单元测试运行脚本
* okgct/test/checkpoints：进行单元测试所需预训练模型存放文件夹。其中modelName\_torch.pth 为训练好的对应模型的torch框架参数文件。modelName\_mindspore.ckpt 为执行单元测试过程中按对应torch预训练模型生成的对应mindspore框架模型的参数文件

## 运行

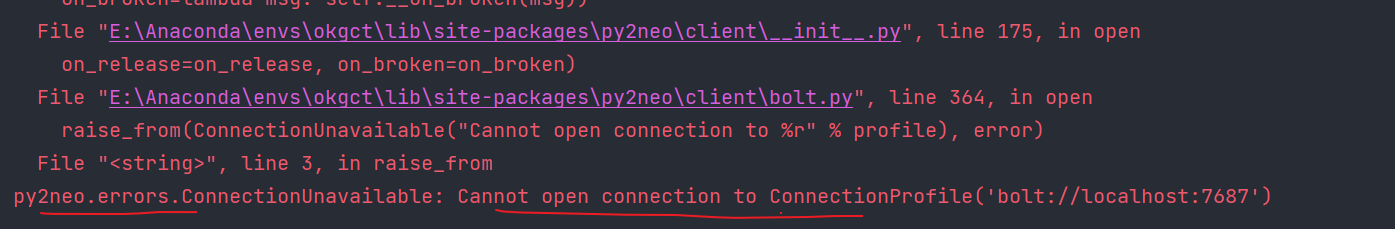
* cd okgct/test
* python unitTest.py
* 10个模型一起测试，控制台、终端观察输出 diff

# 注意事项

1. 图谱构建：knowledgeGraph.py，参数clear=True时会先清空图谱再构建



1. 执行main.py转换程序，转换后的结果会存放在 test/tmp下，每次重新执行前可把tmp下历史项目删除
2. Py2neo 连接报错：



解决：打开终端运行：neo4j.bat console, 后台运行，不要关闭

1. py2neo.errors.ConnectionUnavailable: Connection has been closed

解决：用户名密码错误，参考文档前面更改knowledgeGraph.py

中的用户名和密码