

海外藏中国文物知识管理与服务平台

用户使用手册

计科 2102

1.1 概述

本文旨在说明海外藏中国文物知识管理与服务平台的用户使用方法。系统一共分为 5 个子系统：知识图谱构建子系统、知识服务子系统（Web 端）、掌上博物馆（手机端）、问答子系统、后台管理子系统。

2.1 子系统功能说明

知识图谱构建子系统：

1) 数据爬取：爬取海外博物馆网站的中国文物信息（至少爬取指定的 5 家博物馆的所有中国文物，博物馆网址列表附后），包括文物名字、文物图片、年代、介绍等信息。按照规定格式保存下载的数据。具体数据格式附后。

2) 数据建模：将爬取的数据转化为三元组形式。

3) 数据补充：根据需要从互联网爬取数据，对现有数据进行补充，如相关数据的基础信息缺失，需要从其他来源进行爬取或人工补充。例如，从百度百科下载数据，补充现有数据，如书画作家信息等。

4) 数据存储：将根据步骤 2 中建模好的三元组数据保存到 Virtuoso 或 Neo4j 图数据库中，发布成链接开放数据（选做），可用于关系图谱、时间轴等知识图谱可视化、问答等功能开发；全部数据（用户数据和文物数据等）需要保存到数据库（mysql 或其他数据库）中。

海外文物知识服务子系统：

1) 数据浏览：支持多种形式展示的浏览功能。①提供基本的筛选、排序功能，可按照文物类型、文物年代、博物馆等多种基础信息进行索引、筛选、排序方式浏览文物信息以方便用户的使用。②提供查看文物详情功能，显示文物的详细数据，如文本、图像等信息，点击文物图片，可以进行放大缩小。③相关文物推荐功能，在该文物页面显示相关文物，相关规则自定，如相似主题、同一作者、图像内容相似等。可参考克利夫兰、大英博物馆等网站。

2) 数据查询：支持文物的简单查询功能和高级查询功能。简单查询根据输入的关键字，如文物名称、博物馆名称、文物年代等进行查询。高级查询可以对文物的多个字段进行限定查询，可参考克利夫兰博物馆。

3) 数据可视化显示：将构建的文物知识图谱可视化，以从不同角度等可视化效果，展示文物知识。可参考的可视化方案如下：①包含结点、边的力导向图知识图谱展示，可参考全历史网站、历史人文大数据平台、Neo4j 图数据库等。②文物时间轴：按照时间轴的方式、展示各个时段的文物信息、时间等信息。

4) 用户个人信息管理：用户可以注册登录该系统，设置用户名密码等个人信息。可扩展针对用户使用的多种功能，如最近浏览记录、收藏、评论等功能（选做）。

知识问答子系统：

1) 简单问答：支持单实体单属性的问答，至少支持 10 类以上问题，例如文物的收藏地、时代、材质、类型、介绍、书画作者等。

2) 复杂问答：选做。

3) 闲聊：选做。

掌上博物馆：

1) 文物浏览：显示文物的基本信息、图片，播放讲解等（如果提供了讲解音视频）；可以进行按照关键字搜索的简单搜索。

2) 用户交互：用户可以对单个文物点赞、留言评论等功能、用户可以为单个文物上传自己拍摄的相关照片（例如，用户参观博物馆，拍摄了该文物的照片，然后上传）、可以上传针对某个文物的讲解（选做）、视频（选做）等。

3) 以图搜图：可以上传图片或直接拍摄一个照片，根据上传图片特征搜索相关文物。

4) 用户个人信息管理：用户可以注册登录该系统，设置用户名密码等个人信息。用户信息可以和海外文物知识服务子系统共用或单独使用一套用户系统。

5) 用户个人动态（选做）：用户可以发表动态，上传文字和图片，分享文物或博物馆等，其他用户可以点赞、评论等。（类似微博）

后台管理子系统：

1) 用户管理：管理后台管理子系统用户、掌上博物馆用户、知识服务子系统用户的基础信息。如：对用户信息的增删查改功能；用户权限的管理，如登录、点赞、评论等用户功能的管理。例如，如果掌上博物馆用户发表不良评论，则停止其发表评论的权限。

2) 信息审核：审核用户发表的留言、图片、音视频等内容。对于审核不通过的内容会被屏蔽，同时对该用户进行一定程度惩罚，如禁止发评论等。自动审核功能：对于留言，可以设置敏感词，当用户提交留言出现敏感词时，进行自动屏蔽；对于图片，需要判断是否为不良图片，如果审核不通过，则不显示。人工审核功能：后台管理员或审核人员人工检查用户提交的文本、图片、音视频。

3) 数据管理：管理 1-3 中涉及的所有数据，对所有数据可以进行增删改查等操作，支持单个操作和批量操作。

4) 数据备份和恢复：支持数据库的备份和恢复。实现手动备份恢复或定时备份功能。支持查看所有的备份和恢复记录，显示记录时间等相关功能。可通过点击备份记录来将数据库恢复到该备份记录点上。

5) 日志管理：查看和检索后台管理子系统的操作日志。记录包括管理员等用户对后台数据的操作记录，数据库的备份还原等记录。

3.1 知识图谱构建子系统用户使用手册

MySQL

为了正确连接到文物管理系统的数据库，我们需要进行以下操作：

1.安装数据库管理工具

在本文中，我们使用 MySQL Workbench 对数据库进行连接。

为了安装 MySQL Workbench，我们需要首先访问其官方网站：

<https://dev.mysql.com/downloads/workbench/>

[General Availability \(GA\) Releases](#)[Archives](#)[i](#)

MySQL Workbench 8.0.36

Select Operating System:

Microsoft Windows

Select Operating System...

Microsoft Windows

Ubuntu Linux

Red Hat Enterprise Linux / Oracle Linux

Fedora

macOS

Source Code

for Windows

All MySQL Products. For All Windows Platforms.
In One Package.

Starting with MySQL 5.6 the MySQL Installer package replaces the standalone MSI packages.

Windows (x86, 32 & 64-bit), MySQL Installer MSI

Go to Download Page >

在下拉菜单中选择对应的操作系统版本，并点击右下角 Go to Download Page。

[General Availability \(GA\) Releases](#)[Archives](#)[i](#)

MySQL Installer 8.0.37

!

Note: MySQL 8.0 is the final series with MySQL Installer. As of MySQL 8.1, use a MySQL product's MSI or Zip archive for installation. MySQL Server 8.1 and higher also bundle MySQL Configurator, a tool that helps configure MySQL Server.

Select Version:

8.0.37

Select Operating System:

Microsoft Windows

Windows (x86, 32-bit), MSI Installer (mysql-installer-web-community-8.0.37.0.msi)	8.0.37	2.1M	Download
	MD5: 398f1365f2bd43af9f6ece9add565c1b Signature		
Windows (x86, 32-bit), MSI Installer (mysql-installer-community-8.0.37.0.msi)	8.0.37	296.1M	Download
	MD5: ae605e4f62aaf8bb1adef684d62a49f2 Signature		

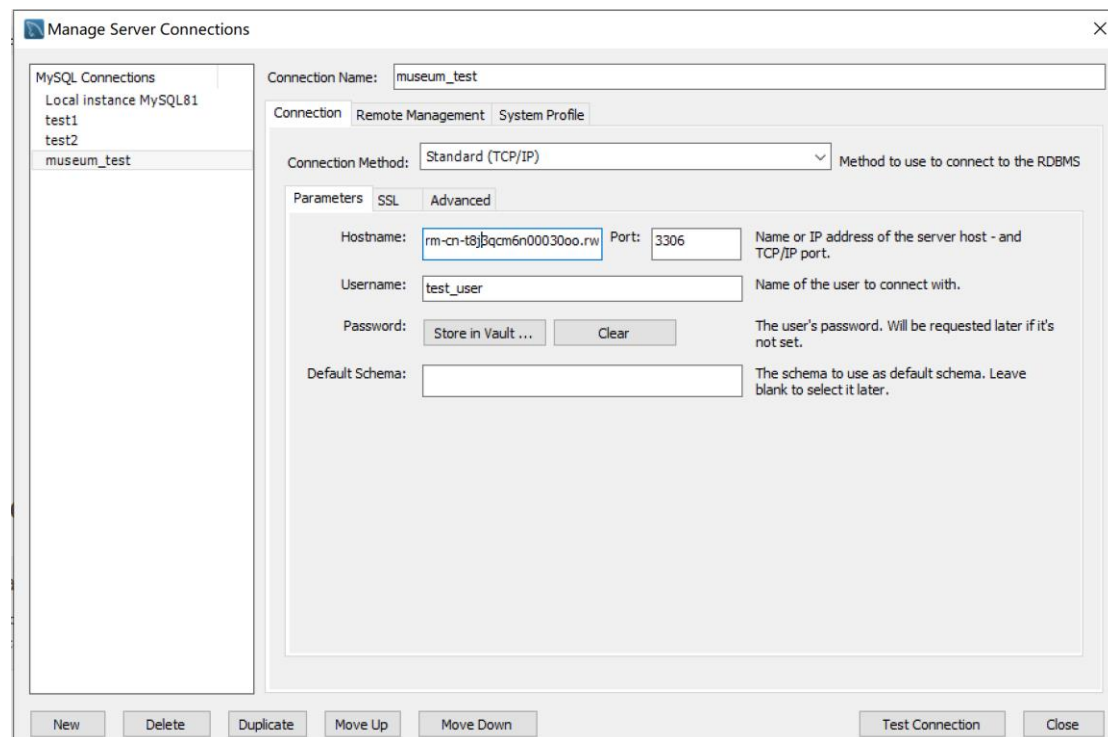
选择一个版本，点击 Download 进行下载。在下载安装程序后，按照其引导进行自动安装，等待安装完成。

2.连接实例

在本地打开 MySQL Workbench 后，单击 MySQL Connections 后的 \oplus 。

MySQL Connections

随后，在 Setup New Connection 对话框中配置 Connection Name、Hostname、Port 和 Username。



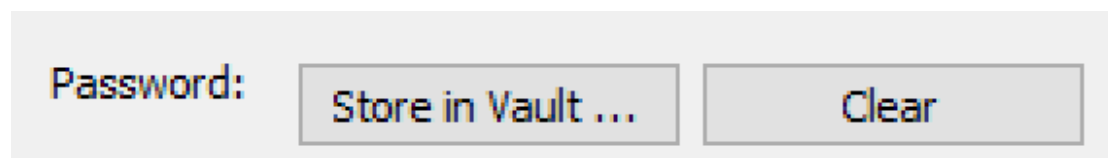
The screenshot shows the 'Manage Server Connections' window. On the left, a list of connections includes 'Local instance MySQL81', 'test1', 'test2', and 'museum_test'. The main area is for configuring a new connection. The 'Connection Name' is 'museum_test'. The 'Connection Method' is 'Standard (TCP/IP)'. The 'Parameters' tab is active, showing fields for 'Hostname' (rm-cn-t8j3qcm6n00030oo.rw), 'Port' (3306), 'Username' (test_user), 'Password' (stored in vault), and 'Default Schema' (blank). The 'Remote Management' and 'System Profile' tabs are also visible.

其中，connection name 用于标识一个连接，设置为 museum_test。

hostname 为 RDS 实例的连接地址，填写实例的外网地址，设置为 rm-cn-t8j3qcm6n00030oo.rwlb.rds.aliyuncs.com。

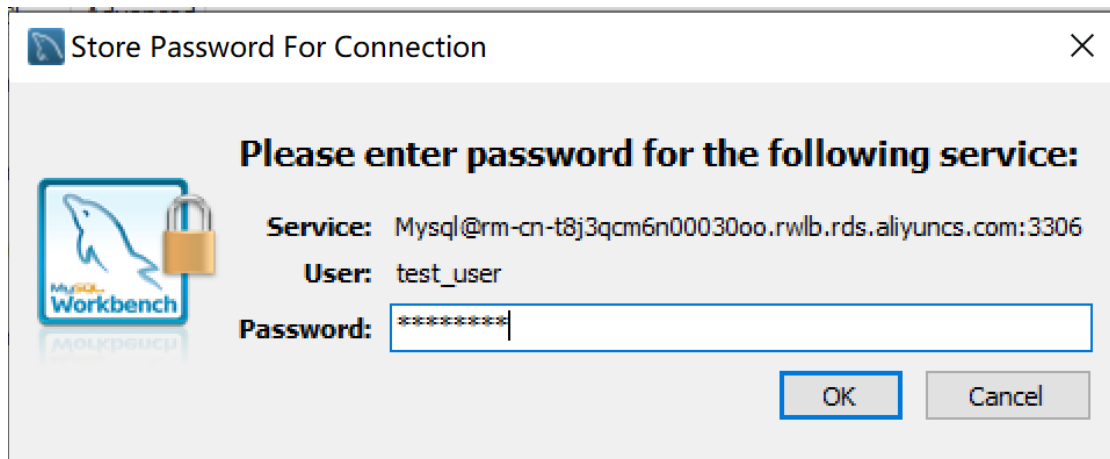
Port 为 RDS 实例的端口，填写 3306。

Username 为文物管理数据库最高权限用户名，设置为 test_user。

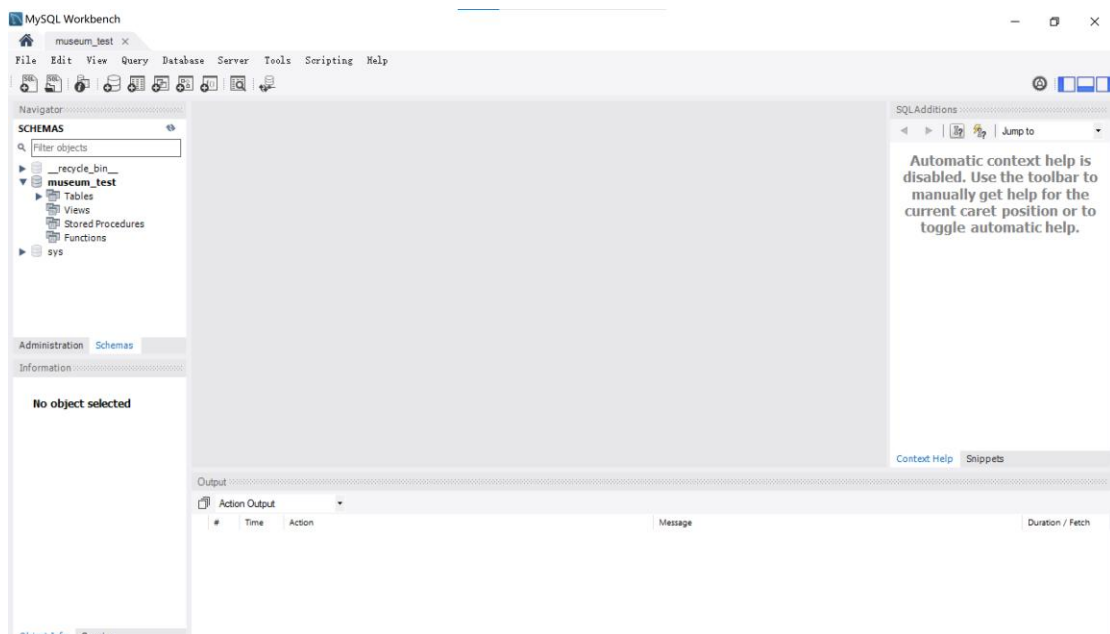


This is a close-up of the password field. It shows the label 'Password:' and two buttons: 'Store in Vault ...' and 'Clear'.

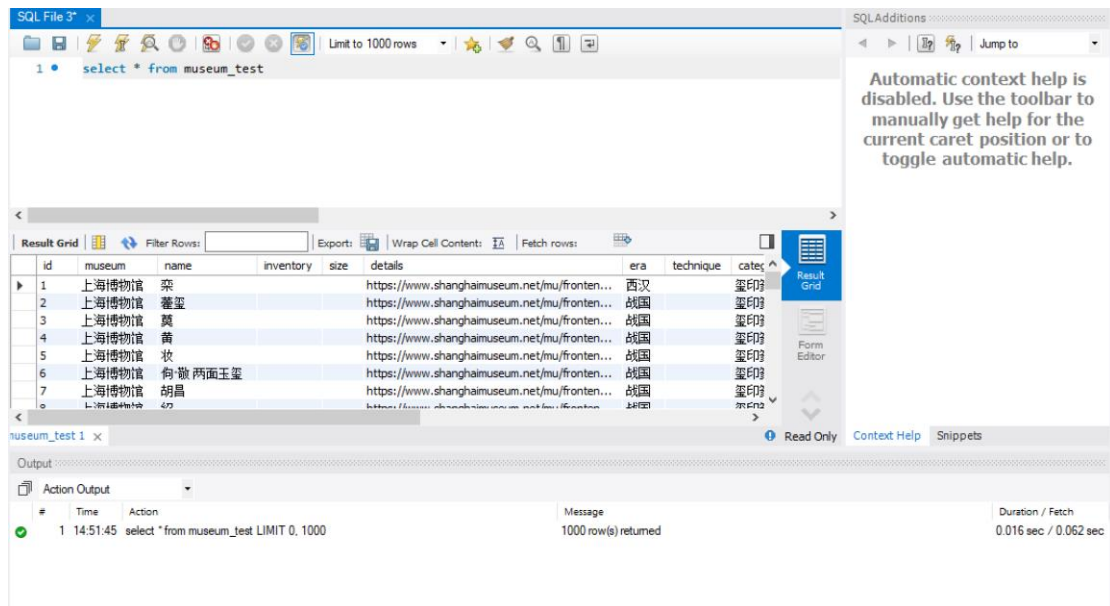
随后，单击 Store in Vault 按钮，输入本数据库密码 Jy123456，并点击 OK。



随后，单击 Test connection，测试能否进行正常连接，如果测试连接成功，则单击确认窗口中的 OK 后，再单击 Setup New Connection 窗口中的 OK。
之后，在 MySQL Workbench 首页找到已创建的连接，单击进入管理页面。出现如图页面，即代表连接成功：

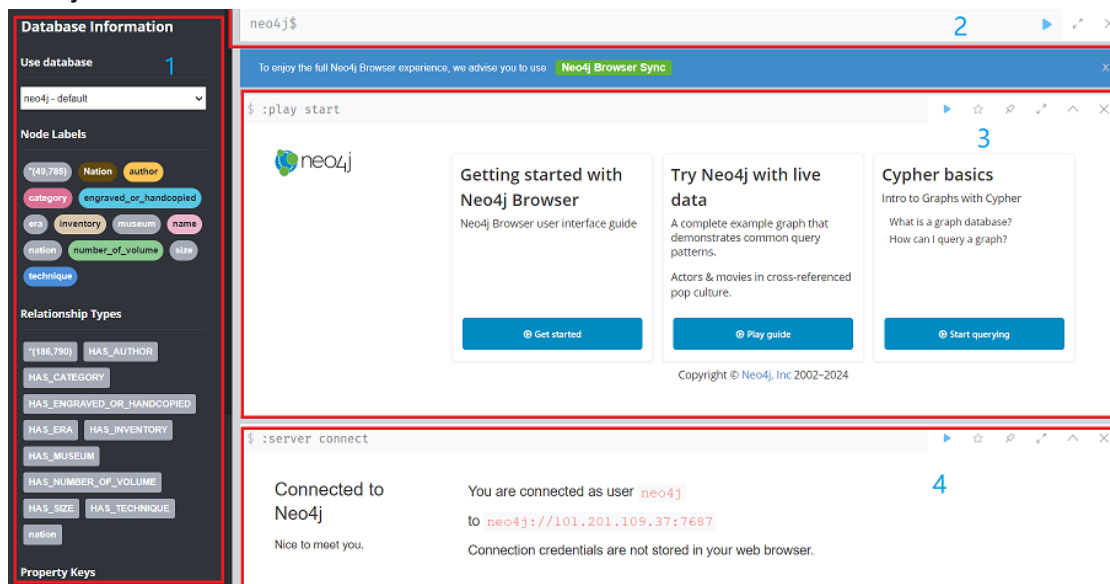


在此界面，您可以使用 MySQL Workbench 对数据库内容进行操作。



如图为连接成功后可进行的查询操作。

neo4j



如上图所示，简单介绍一下各部分：

区域 1：菜单栏

区域 2：代码命令行

区域 3：运行例子演示

区域 4：服务器连接信息

1 菜单栏

1.1 Database Information（数据库信息）

Database Information

Use database

neo4j - default



Node Labels

*(49,785)

Nation

author

category

engraved_or_handcopied

era

inventory

museum

name

nation

number_of_volume

size

technique

Relationship Types

*(186,790)

HAS_AUTHOR

HAS_CATEGORY

HAS_ENGRAVED_OR_HANDCOPIED

HAS_ERA

HAS_INVENTORY

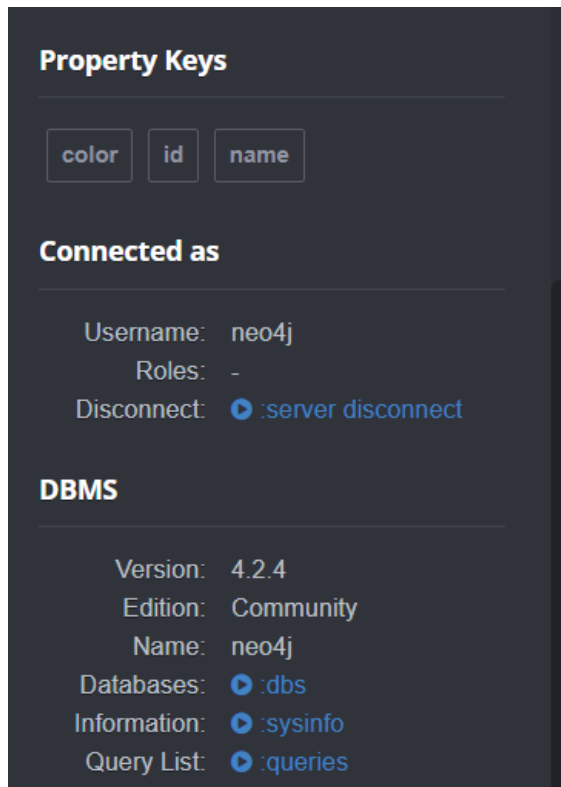
HAS_MUSEUM

HAS_NUMBER_OF_VOLUME

HAS_SIZE

HAS_TECHNIQUE

nation



这个展示了本地 neo4j 数据库的相关信息（可以进行点击，会在右边进行展示相关内容）：

Node Labels：创建的节点信息，Neo4j 里面都是节点（相当于 mysql 里面的表信息），点击可以在右边显示出 49785 个节点信息内容

Relationship Type：关系类型，目前创建了知识文物图谱的 11 种关系，一共 186790 条，包含种类，时间，国家，技巧等关系。

Property Keys：属性关键词

Connected as：数据库连接信息

server user list：点击后，会执行该命令，右边显示出当前数据库的用户列表

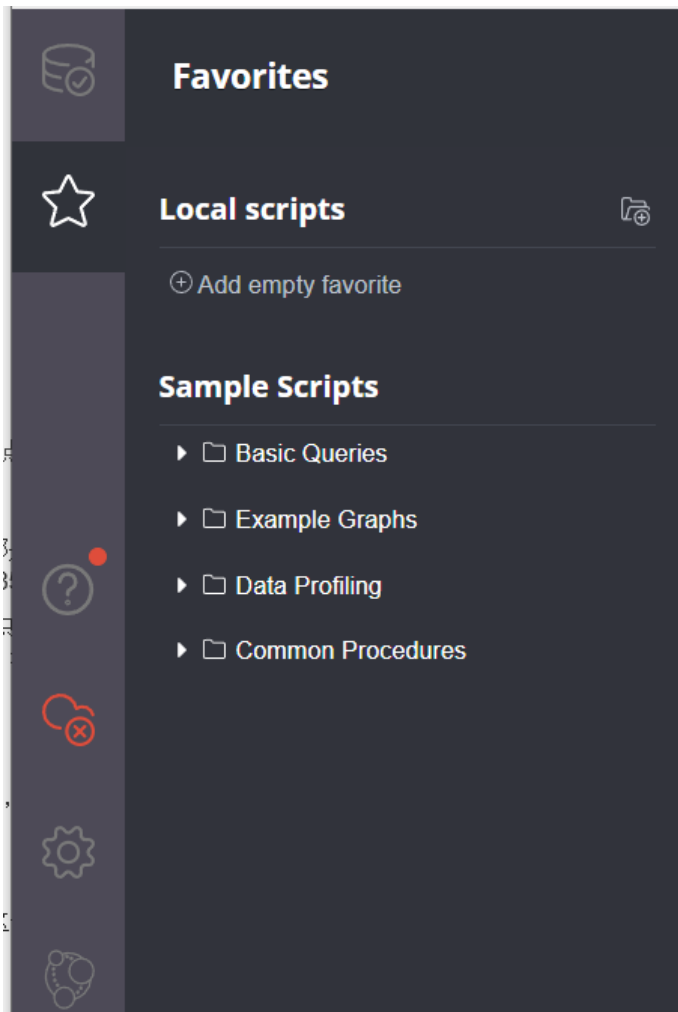
server user add：添加用户

server disconnect：这个最好不要点，这个是断开当前数据库连接

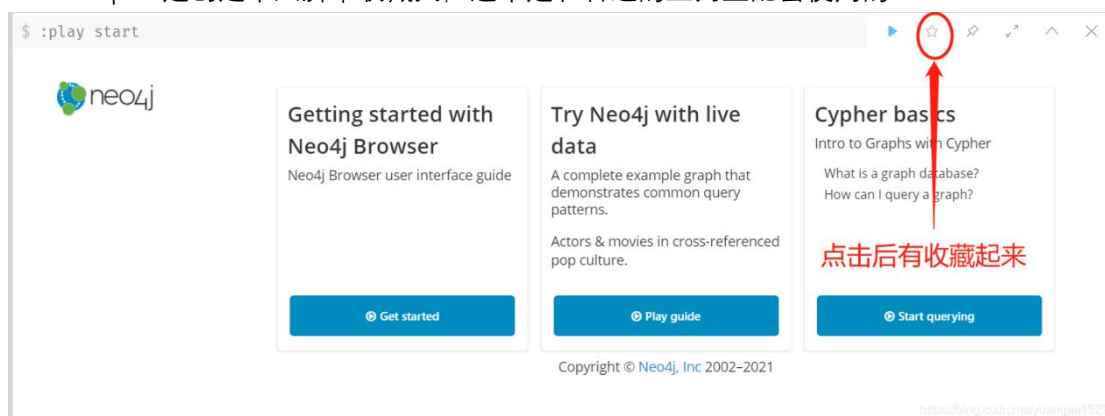
DBMS：数据库信息介绍

这个菜单项最大的用处，就是将 Neo4j 数据库信息罗列出来了，通过点击操作可以在右边可视化数据信息。

1.2 Favorites（收藏夹）



local scripts 是创建个人脚本收藏夹，这个是和右边的五角星配套使用的：

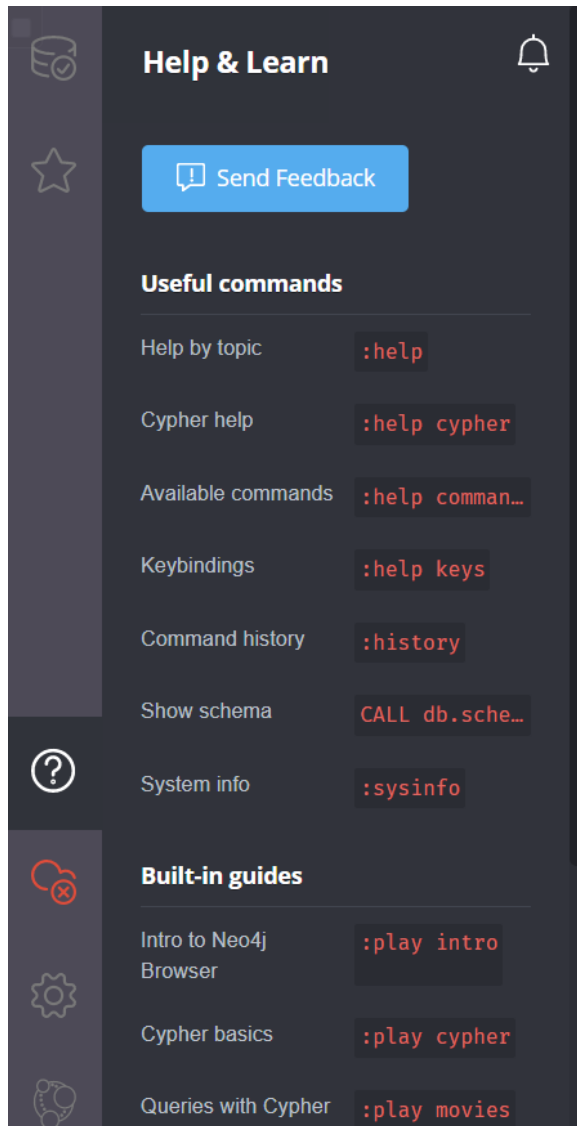


记住，上面收藏起来的其实是一个命令：

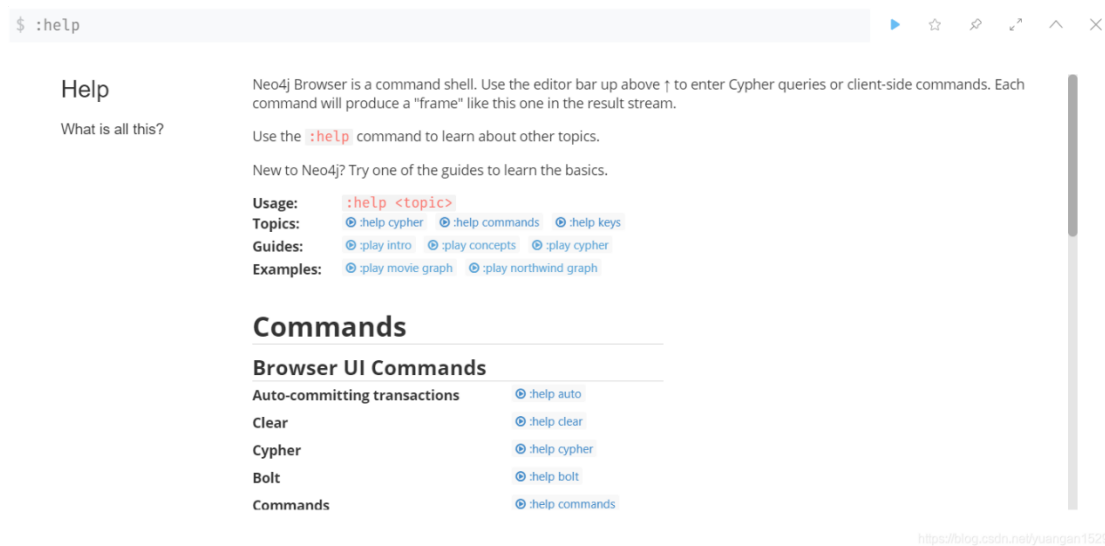
```
:play start
```

Sample scripts 是 neo4j 自带的收藏夹，里面有很多功能代码实例，通过收藏名称就可以知道其用途，可以点击运行——在右边可视化展示出来

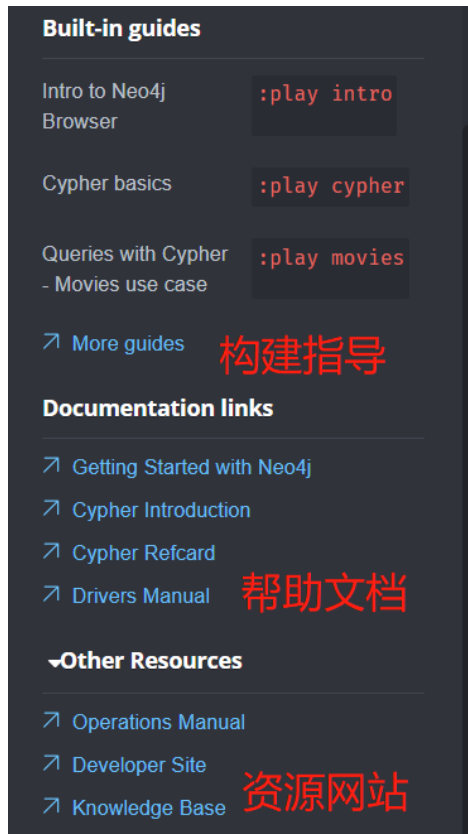
1.3 帮助与学习



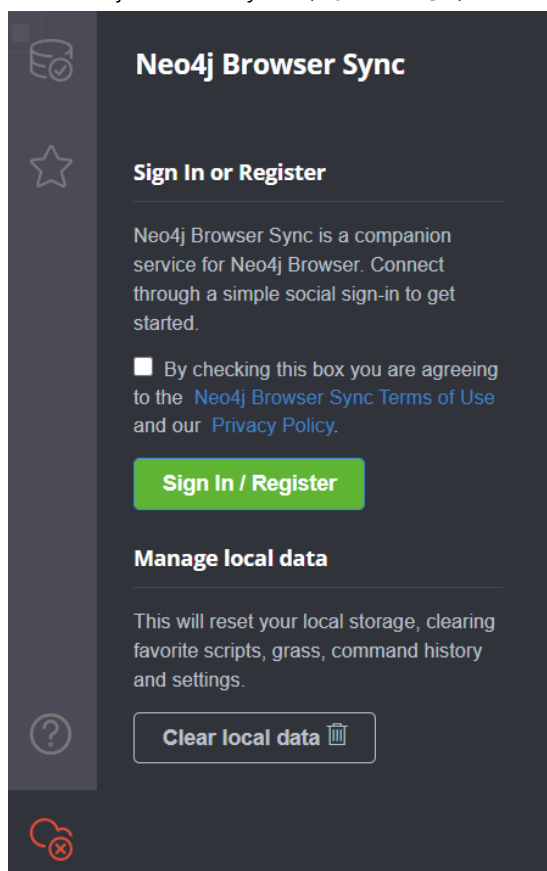
这里给出了很多有用的命令，比如":help"，在右边命令行输入，可以得到很多信息：



最后几个区域是学习区：



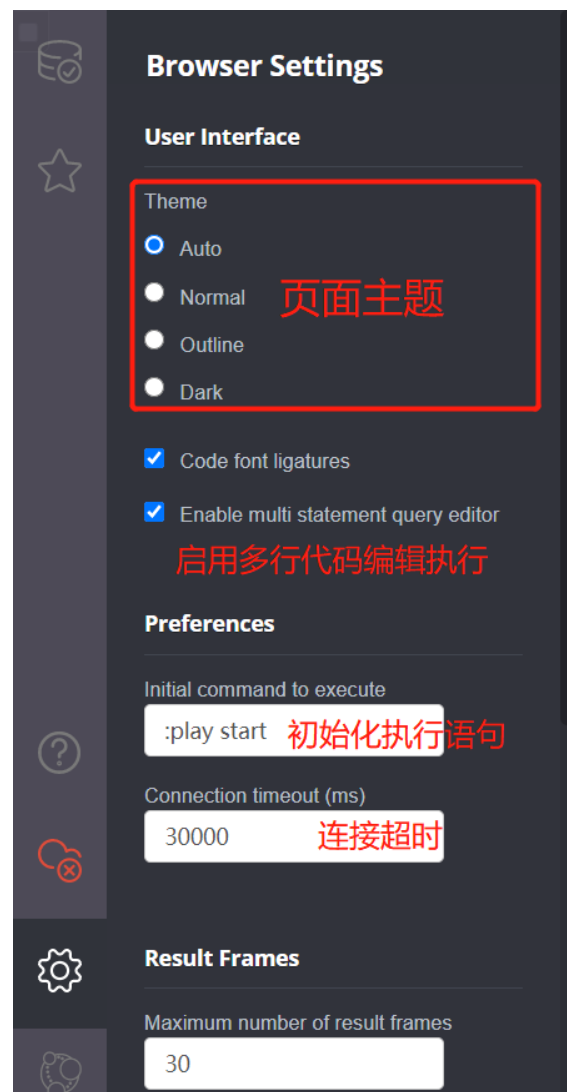
1.4 Neo4j browser sync (浏览器同步)



这个和 neo4j 数据库没什么关系，这个登录的是 Neo4j browser 账户，该账户连接了本地

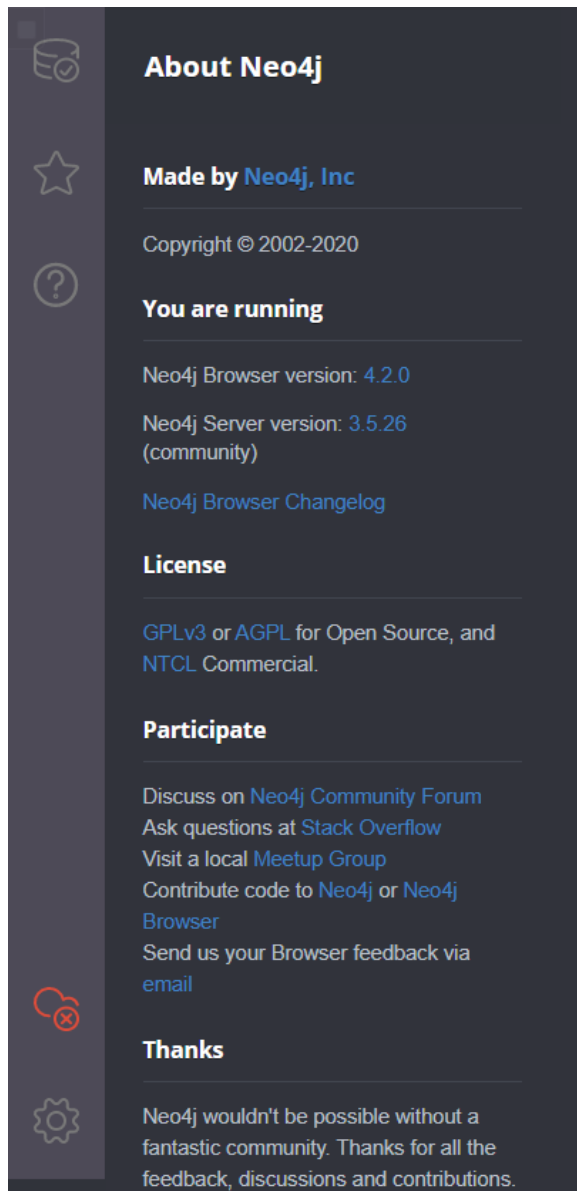
Neo4j 数据库, 它也可以连接其他数据库, 这个要分清楚; 该账户里面存有一些收藏夹信息, 到时候可以同步, 可以看到其左边图标有一个红色叉, 表明当前并没有登录账户。
Manage local data 可以对操作记录进行管理, 删除历史记录。

1.5 浏览器设置



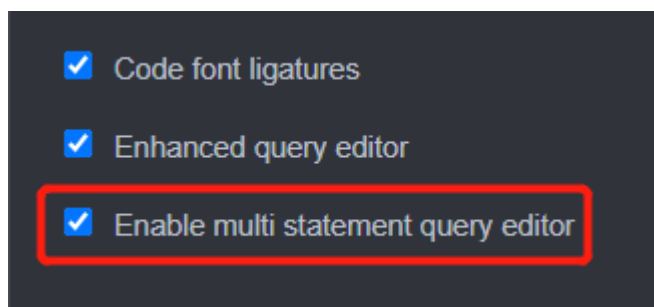
初始化执行语句, 就是指打开 Neo4j 浏览器时, 会自动执行该语句, 然后右边会显示出该语句执行结果, 上面也提到了该语句执行后会给出 8 个小页面——一个完整的代码实例。

2 关于 neo4j



2.1 代码输入命令行

如果在设置中勾选了多行语句：



如果一次执行多行语句，如：

```
CREATE (n:Location {city:'Miami', state:'FL'})
CREATE (n:Location {city:'Boston', state:'MA'})
CREATE (n:Location {city:'Lynn', state:'MA'})
CREATE (n:Location {city:'Portland', state:'ME'})
```

```
CREATE (n:Location {city:'San Francisco', state:'CA'})
```

则会报错如下：

```
$ CREATE (n:Location {city:'Miami', state:'FL'}) CREATE (n:Location {city:'Boston', state:'MA'}) CREATE (n:Loca...
ERROR Neo.ClientError.Statement.SyntaxError
Variable 'n' already declared (line 2, column 9 (offset: 55))
"CREATE (n:Location {city:'Boston', state:'MA'})"
^
```

更正方法有两种。

第一种：将 n 的命名更改

每个 create 里的 n 赋值成不同的变量：

```
CREATE (n1:Location {city:'Miami', state:'FL'})
CREATE (n2:Location {city:'Boston', state:'MA'})
CREATE (n3:Location {city:'Lynn', state:'MA'})
CREATE (n4:Location {city:'Portland', state:'ME'})
CREATE (n5:Location {city:'San Francisco', state:'CA'})
```

第二种：增加分号

需要给每行语句增加分号：

```
CREATE (n:Location {city:'Miami', state:'FL'});
CREATE (n:Location {city:'Boston', state:'MA'});
CREATE (n:Location {city:'Lynn', state:'MA'});
CREATE (n:Location {city:'Portland', state:'ME'});
CREATE (n:Location {city:'San Francisco', state:'CA'})
```

再次执行，则不会报错：

```
$ CREATE (n:Location {city:'Miami', state:'FL'}); CREATE (n:Location {city:'Boston', state:'MA'}); CREATE (n:Locatio...
$ CREATE (n:Location {city:'Miami', state:'FL'})
$ CREATE (n:Location {city:'Boston', state:'MA'})
$ CREATE (n:Location {city:'Lynn', state:'MA'})
$ CREATE (n:Location {city:'Portland', state:'ME'})
$ CREATE (n:Location {city:'San Francisco', state:'CA'})
```

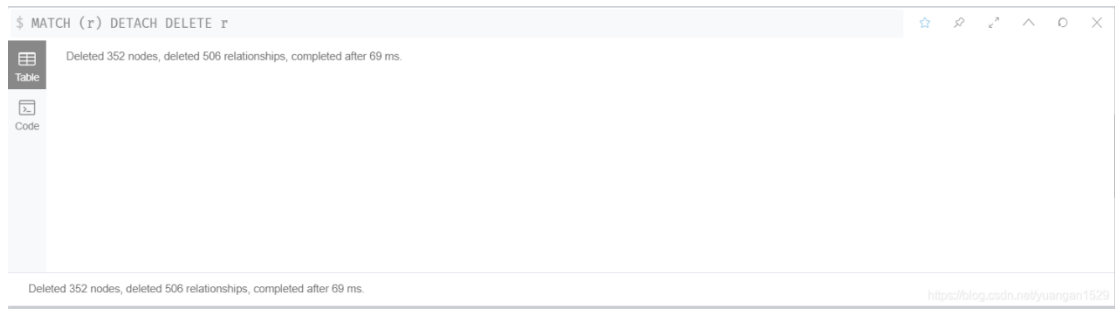
3 play start

这个区域可以说是每行代码执行结束后的显示区域，只是不同的代码性质不一样，其执行结果显示区域也不同。

比如执行结果：

```
$ CREATE (n:Location {city:'Miami', state:'FL'}); CREATE (n:Location {city:'Boston', state:'MA'}); CREATE (n:Locatio...
$ CREATE (n:Location {city:'Miami', state:'FL'})
$ CREATE (n:Location {city:'Boston', state:'MA'})
$ CREATE (n:Location {city:'Lynn', state:'MA'})
$ CREATE (n:Location {city:'Portland', state:'ME'})
$ CREATE (n:Location {city:'San Francisco', state:'CA'})
```

删除结果：



3.2 海外文物知识服务子系统用户使用手册

主要功能

数据浏览：

提供基本的筛选、排序功能，可按照文物类型、文物年代、博物馆等多种基础信息进行筛选、排序

提供查看文物详情功能，显示文物的详细数据，点击文物图片，可以进行放大缩小

数据查询：

根据输入的关键词，如文物名称、博物馆名称、文物年代等进行查询

时间轴与登录功能：

文物时间轴：按照时间轴的方式、展示各个时段的文物信息、时间等信息

用户可以注册登录该系统

主要功能介绍：

平台首页：

轮播图展示文物图片 用户可通过点击使用文物浏览（文物排序）、查询、可视化、时间轴以及用户信息管理功能



文物浏览（文物排序）：

可浏览数据库中的全部文物 并对每个文物实现查看文物详情功能


可选择按朝代、类型、博物馆三种排序文物

海外藏中国文物知识管理与服务平台

首页 文物浏览 查询 可视化 时间轴 用户信息管理

共 6819 条 10条/页 < 1 2 3 4 5 6 ... 682 > 前往 2 页 请选择排序方式 排序


箕侯簋



朝代：西周
尺寸：暂无数据
类型：铜器
所属博物馆：上海博物馆
描述：
<https://www.shanghaimuseum.net/mu/frontend/pg/article/>

查看详情

鼎



朝代：商
尺寸：暂无数据
类型：铜器

查看详情

文物详情：
展示文物详细资料 提供根据文物类型推荐相关文物功能


海外藏中国文物知识管理与服务平台

首页 文物浏览 查询 可视化 时间轴 用户信息管理

返回

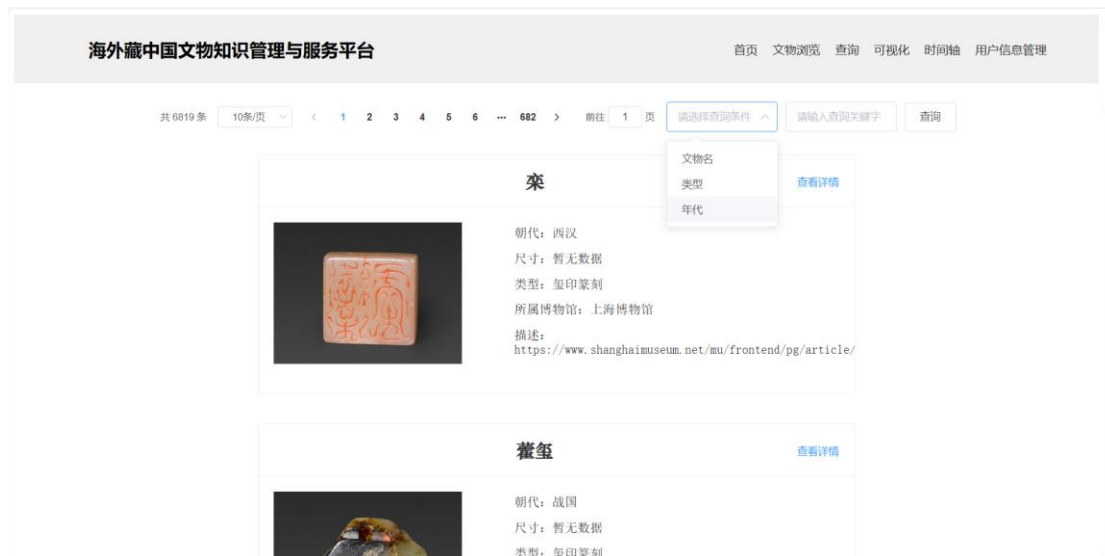
文物详情

相关推荐

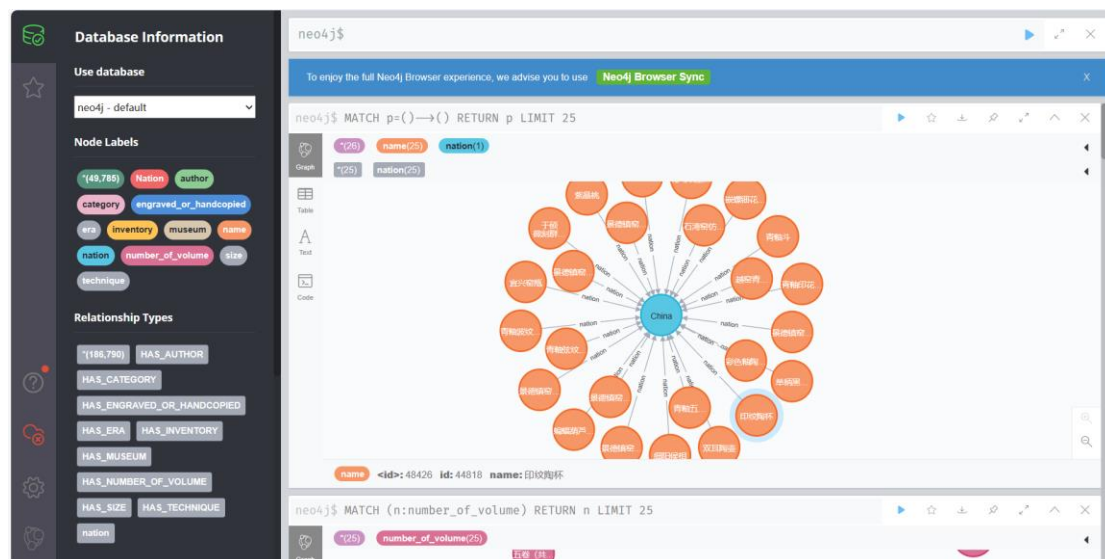


文物ID: 11
文物名称: 箕侯簋
朝代: 西周
作者: 暂无数据
尺寸: 暂无数据
类型: 铜器
所属博物馆: 上海博物馆
库存单: 暂无数据
介绍: <https://www.shanghaimuseum.net/mu/frontend/pg/article/id/CI00159002>
技艺: 暂无数据
卷数: 暂无数据
雕刻或手工复制: 暂无数据

查询：
可按照文物名、类型、年代对文物进行模糊查询



可视化:
将博物馆数据可视化处理



时间轴:
按照时间轴的方式、展示各个时段的文物信息、时间等信息



用户信息管理：
可进行登陆注册操作



3.3 知识问答子系统用户使用手册

博物馆查询

在主界面，您可以输入关键词或选择感兴趣的博物馆，系统将返回相关的博物馆历史信息和时间。比如我想要获得上海博物馆的相关信息时，可以进行如下对话：



您还可以查看相关博物馆过去和未来的展览计划，并获取相关的详细介绍和参观信息。比如想要得知上海博物馆最近的展览计划时可以进行如下对话：



藏品查询

您可以搜索特定的藏品名称、类别或历史背景等关键词，系统将返回相关的藏品信息。您可以通过阅读详细介绍来了解博物馆的各种珍贵藏品。比如说想要得知四羊方尊的相关信息时，可以发生如下对话：



比如说想要得知青铜器的相关信息时，可以发生如下对话：



他将从历史背景与起源、种类与用途、制作工艺、特点与价值、收藏与发现这几个方面为我们做出详细描述。

如果我们想要得知具体朝代的某一类型藏品时，比如：唐代的陶瓷时，可以发生如下对话：



如果我们想要得知与某个藏品类似的藏品时，比如说想知道和四羊方尊相似的产品时，可以发生如下对话：

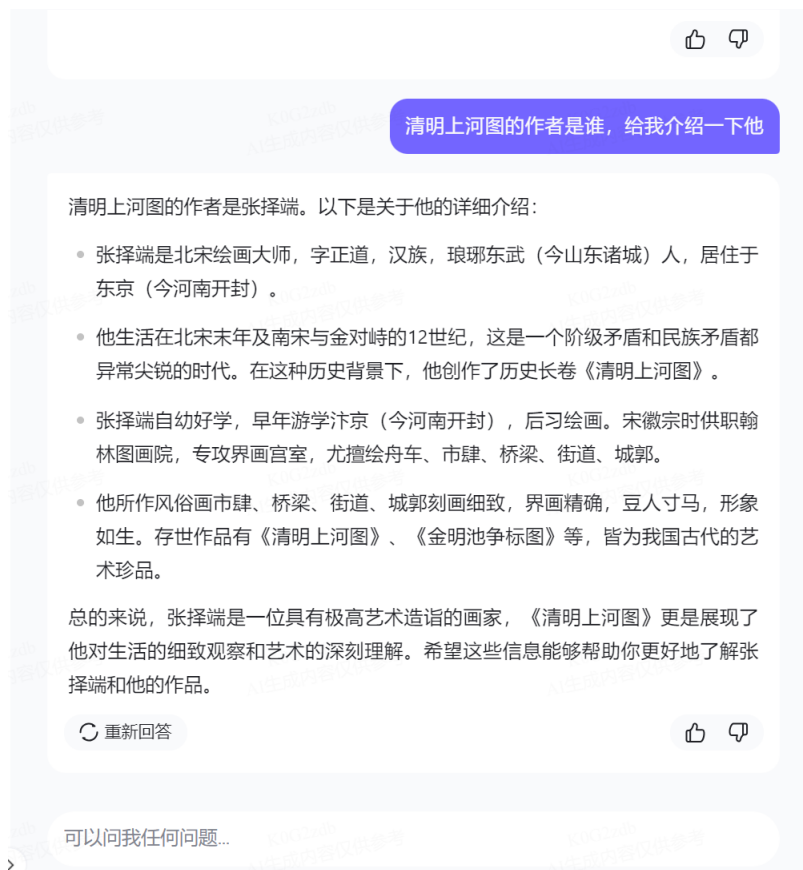


我们也可以根据作者来查询他相关的作品,比如说想要了解齐白石的几个著名作品时可以发生如下对话:



作者查询

我们可以输入藏品的名称及其相关信息得知该藏品作者的相关信息,比如说我想知道《清明上河图》的作者相关信息时可以发生如下对话:



位置查询

我们可以输入藏品的相关信息从而得知其目前展览的位置，如下图所示：



3.4 掌上博物馆用户使用手册

详细内容另附

3.5 后台管理子系统用户使用手册

内置功能：

- 1. 用户管理：用户是系统操作者，该功能主要完成系统用户配置。
- 2. 部门管理：配置系统组织机构（公司、部门、小组），树结构展现支持数据权限。
- 3. 岗位管理：配置系统用户所属担任职务。
- 4. 菜单管理：配置系统菜单，操作权限，按钮权限标识等。
- 5. 角色管理：角色菜单权限分配、设置角色按机构进行数据范围权限划分。
- 6. 字典管理：对系统中经常使用的一些较为固定的数据进行维护。
- 7. 参数管理：对系统动态配置常用参数。
- 8. 通知公告：系统通知公告信息发布维护。
- 9. 操作日志：系统正常操作日志记录和查询；系统异常信息日志记录和查询。
- 10. 登录日志：系统登录日志记录查询包含登录异常。
- 11. 在线用户：当前系统中活跃用户状态监控。
- 12. 定时任务：在线（添加、修改、删除）任务调度包含执行结果日志。
- 13. 代码生成：前后端代码的生成（java、html、xml、sql）支持 CRUD 下载 。
- 14. 系统接口：根据业务代码自动生成相关的 api 接口文档。
- 15. 服务监控：监视当前系统 CPU、内存、磁盘、堆栈等相关信息。
- 16. 缓存监控：对系统的缓存信息查询，命令统计等。
- 17. 在线构建器：拖动表单元元素生成相应的 HTML 代码。
- 18. 连接池监视：监视当前系统数据库连接池状态，可进行分析 SQL 找出系统性能瓶颈。

环境要求：

IntelliJ IDEA 2024.1, Navicat Premium, Visual Studio Code, Windows10 及以上的系统, redis 中间件（在此之前需要在 IDEA 内搭建好 springboot 框架所需要的环境）

主要功能介绍：

1、博物馆信息管理：

若依管理系统

首页

博物馆信息

博物馆信息...

用户管理

系统管理

系统监控

系统工具

博物馆信息管理

新增

重置

新增

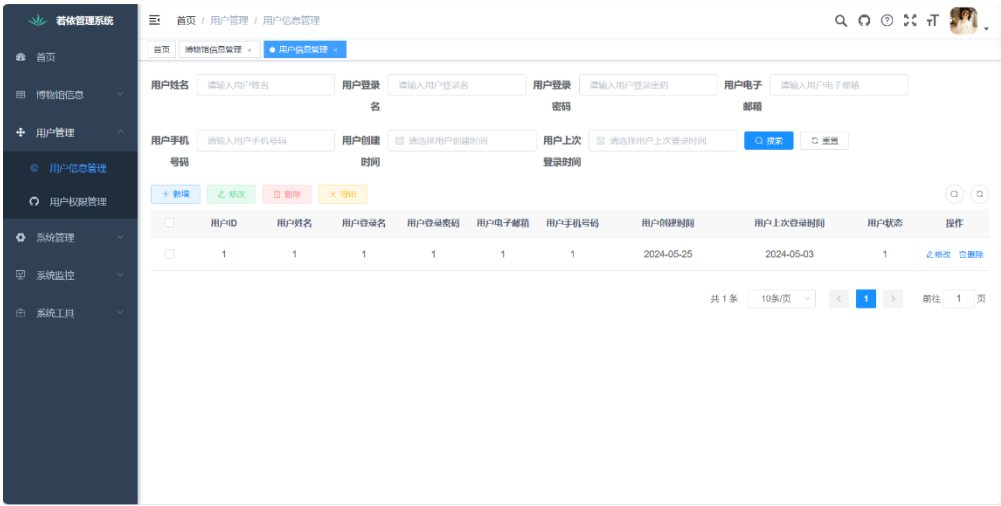
修改

删除

导出

<input type="checkbox"/>	博物馆名称	文物名称	存货	大小	源地址	朝代	工艺	种类	图片url	作者	数量	雕刻或手工	复制
<input type="checkbox"/>	上海博物馆	宋			https://www.shanghaimuseum.net/multimedia/asset/159252	西汉		玺印篆刻	https://www.shanghaimuseum.net/multimedia/asset/202302/a486051c-004a-49ff-a8d4-d59744973395.jpg			复制	修改 删除
<input type="checkbox"/>	上海博物馆	铜			https://www.shanghaimuseum.net/multimedia/asset/159238	战国		玺印篆刻	https://www.shanghaimuseum.net/multimedia/asset/202302/7803c28b-3c75-413fb1c9-5b084caa5d.jpg			复制	修改 删除

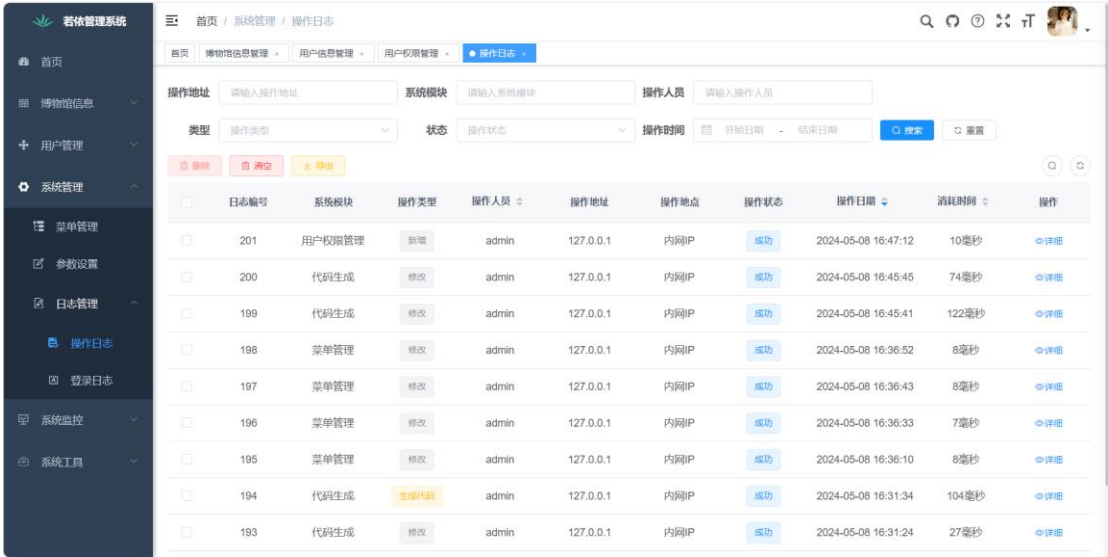
2、用户信息管理：



3、用户权限管理：



4、操作日志：



5、登录日志：

若依管理系统

首页

博物馆信息

用户管理

系统管理

菜单管理

参数设置

日志管理

操作日志

登录日志

系统监控

系统工具

三 首页 / 系统管理 / 登录日志

博物馆信息管理 用户信息管理 用户权限管理 操作日志 登录日志

登录地址 请输入登录地址 用户名 请输入用户名 状态 登录状态

登录时间 开始日期 结束日期 搜索 重置

删除 清空 刷新 导出

10 20

<input type="checkbox"/>	访问编号	用户名	登录地址	登录地点	浏览器	操作系统	登录状态	操作信息	登录日期
<input type="checkbox"/>	114	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-15 14:40:06
<input type="checkbox"/>	113	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 16:43:40
<input type="checkbox"/>	112	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 16:20:51
<input type="checkbox"/>	111	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 15:48:47
<input type="checkbox"/>	110	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 15:35:41
<input type="checkbox"/>	109	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 14:00:36
<input type="checkbox"/>	108	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 13:52:16
<input type="checkbox"/>	107	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	成功	登录成功	2024-05-08 13:14:32
<input type="checkbox"/>	106	admin	127.0.0.1	内网IP	Chrome 12	Windows 10	失败	验证码错误	2024-05-08 13:14:29

6、相关参数管理：

在 application-druid.yml 中, 可以对进行连接的数据库和控制台用户名, 密码进行修改, 如下图所示：

```
# 主库数据源
master:
  url: jdbc:mysql://localhost:3306/ruoyidb?useUnicode=true&characterEncoding=utf8&zeroDateTimeBehavior=convertToNull
  username: root
  password: 123456
```

```
# 控制台管理用户名和密码
login-username: ruoyi
login-password: 123456
```