第9章　程式

9-1 元件清單及其規格描述

▼表9-1-2 元件清單及其規格描述表(後端)

|  |  |  |
| --- | --- | --- |
| Python檔案(後端)－圖形資料庫(Neo4j) | | |
| 編號 | 檔案名稱 | 功能 |
| 2-3-1 | neo4j\_transfer\_and\_creation.py | 將mysql資料導入neo4j，並建立節點(房屋物件、價格、房屋類型、房間類型、坪數)以及建立關聯 |
| 2-3-2 | neo4j\_nearby\_features.py | 將整理好的.json(附近特色:學校、商店、醫院……)導入先前建好的neo4j資料庫，建立節點及關聯 |
| 2-3-3 | neo4j\_transportation.py | 將整理好的.json(附近交通:公車、捷運)導入先前建好的neo4j資料庫，建立節點及關聯 |
| 2-3-4 |  |  |
| 2-3-5 |  |  |

9-2 其他附屬之各種元件

▼表9-2-1 部分程式碼－neo4j\_transfer\_and\_creation.py

|  |  |  |  |
| --- | --- | --- | --- |
| 編號 | 對應9-1編號 | 程式名稱 | neo4j\_transfer\_and\_creation.py |
| 目的 | 將mysql資料導入neo4j，並建立節點(房屋物件、價格、房屋類型、房間類型、坪數)以及建立關聯 | | |
| 部分程式碼 | | | |
| # 连接到MySQL数据库  嘗試  建立MySQL连接，使用本地服务器、用户名、密码和数据库名稱  如果连接成功，顯示 "MySQL connection established."  捕獲錯誤  如果有錯誤，顯示錯誤訊息並退出程序  # 定义一个重试连接Neo4j的函数  函數 connect\_to\_neo4j(重试次数=5, 等待时间=2)  迭代重试次數  嘗試  建立Neo4j连接  如果连接成功，顯示 "Neo4j connection established."  返回Neo4j连接  捕獲錯誤  顯示 "Attempt {重试次数} failed: {錯誤訊息}"  等待指定的時間再重试  如果多次尝试后仍然无法连接，顯示 "Failed to connect to Neo4j after multiple attempts." 并退出程序  # 连接到Neo4j数据库  呼叫 connect\_to\_neo4j 函數建立连接  # 查询MySQL数据  嘗試  建立MySQL游標  执行SQL查询以获取数据  如果查询成功，顯示 "MySQL data successfully retrieved."  捕獲錯誤  如果有錯誤，顯示錯誤訊息，关闭MySQL连接并退出程序  # 定义价格和大小范围  定義 price\_ranges 為一個列表，其中包含價格範圍的元組  定義 size\_ranges 為一個列表，其中包含大小範圍的元組  # 定义解析大小的函数  函數 parse\_size(大小字符串)  使用正则表达式匹配大小字符串中的数字  如果匹配成功，返回匹配的数字  如果匹配失败，返回None  # 处理数据并写入Neo4j  嘗試  对每行MySQL数据进行迭代  获取行中的各个字段  呼叫 parse\_size 函数解析大小字段  如果解析失败，顯示跳过的訊息并继续下一行    # 创建房屋节点  创建一个 Property 节点并设置属性  将此节点写入Neo4j  顯示 "Created Property node: {hid}"  # 创建物件类型节点  创建一个 Type 节点  使用 merge 操作避免重複创建  顯示 "Ensured Type node: {type}"  # 创建物件類型（pattern）節點  创建一个 Pattern 节点  使用 merge 操作避免重複创建  顯示 "Ensured Pattern node: {pattern}"  # 根据价格创建价格范围节点  迭代價格範圍  如果物件的价格在範圍內  创建价格范围节点  使用 merge 操作避免重複创建  顯示 "Ensured PriceRange node: {price\_range\_name}"  # 创建与价格范围的关联  创建价格范围关系  将此关系写入Neo4j  顯示 "Created relationship: {hid} -> {price\_range\_name} (PriceRange)"  退出价格範圍迴圈  # 根据大小创建大小范围节点  迭代大小範圍  如果物件的大小在範圍內  创建大小范围节点  使用 merge 操作避免重複创建  顯示 "Ensured SizeRange node: {size\_range\_name}"  # 创建与大小范围的关联  创建大小范围关系  将此关系写入Neo4j  顯示 "Created relationship: {hid} -> {size\_range\_name} (SizeRange)"  退出大小範圍迴圈  # 创建房屋与类型及類型节点的关联  创建 "HAS\_TYPE" 和 "HAS\_PATTERN" 关联  将此关联写入Neo4j  顯示 "Created relationship: {hid} -> {type} (Type)" 和 "Created relationship: {hid} -> {pattern} (Pattern)"  顯示 "All data successfully written to Neo4j."  捕獲錯誤  如果有錯誤，顯示錯誤訊息  # 关闭MySQL连接  关闭MySQL游標和连接  顯示 "MySQL connection closed." | | | |

▼表9-2-2 部分程式碼－neo4j\_nearby\_features.py

|  |  |  |  |
| --- | --- | --- | --- |
| 編號 | 對應9-1編號 | 程式名稱 | neo4j\_nearby\_features.py |
| 目的 | 將整理好的.json(附近特色:學校、商店、醫院……)導入先前建好的neo4j資料庫，建立節點及關聯 | | |
| 部分程式碼 | | | |
| # 连接到Neo4j数据库的函数，带有重试机制  函數 connect\_to\_neo4j(重试次数=5, 等待时间=2)  迭代重试次數  嘗試  建立Neo4j连接  如果连接成功，顯示 "Neo4j connection established."  返回Neo4j连接  捕獲錯誤  顯示 "Attempt {重试次数} failed: {錯誤訊息}"  等待指定的時間再重试  如果多次尝试后仍然无法连接，顯示 "Failed to connect to Neo4j after multiple attempts." 并退出程序  # 连接到Neo4j数据库  呼叫 connect\_to\_neo4j 函數建立连接  # JSON文件路径  設定 json\_file\_path 为JSON文件的路径  # 检查文件是否存在  如果文件不存在  顯示 "File not found: {json\_file\_path}" 并退出程序  # 读取JSON文件  嘗試  打开JSON文件并读取内容  解析内容为 store\_data  捕獲 JSON 解析错误  顯示 "Error reading JSON file: {錯誤訊息}" 并退出程序  # 处理数据并写入Neo4j  嘗試  对每个条目进行迭代  获取条目中的 'hid' 和 'store' 字段  如果 'store' 不是一个列表  顯示 "Skipping property {hid} because 'store' is not a list." 并跳过该条目  # 查找对应的Property节点  查找具有 'hid' 的 Property 节点  如果未找到该节点  顯示 "Property {hid} not found in the database." 并跳过该条目  # 创建或获取每个Store节点并创建关系  对 'store' 列表中的每个店铺进行迭代  创建或获取 Store 节点  使用 merge 操作避免重复创建  创建 Property 节点与 Store 节点之间的 NEAR\_STORE 关系  顯示 "Created relationship: {hid} -> {store} (Store)"  顯示 "All store data successfully integrated into Neo4j."  捕獲錯誤  如果处理数据时出错，顯示 "Error processing data: {錯誤訊息}" | | | |

▼表9-2-3 部分程式碼－neo4j\_transportation.py

|  |  |  |  |
| --- | --- | --- | --- |
| 編號 | 對應9-1編號 | 程式名稱 | neo4j\_transportation.py |
| 目的 | 將整理好的.json(附近交通:公車、捷運)導入先前建好的neo4j資料庫，建立節點及關聯 | | |
| 部分程式碼 | | | |
| # 假设你已经连接到了Neo4j数据库的函数，带有重试机制  函數 connect\_to\_neo4j(重试次数=5, 等待时间=2)  迭代重试次數  嘗試  建立Neo4j连接  如果连接成功，顯示 "Neo4j connection established."  返回Neo4j连接  捕獲錯誤  顯示 "Attempt {重试次数} failed: {錯誤訊息}"  等待指定的時間再重试  如果多次尝试后仍然无法连接，顯示 "Failed to connect to Neo4j after multiple attempts." 并退出程序  # 连接到Neo4j数据库  呼叫 connect\_to\_neo4j 函數建立连接  # 读取JSON文件  設定 json\_file\_path 为JSON文件的路径  # 檢查文件是否存在  如果文件不存在  顯示 "File not found: {json\_file\_path}" 并退出程序  # 讀取 JSON 文件  打开 JSON 文件并读取内容到 data 变量中  # 处理数据并写入Neo4j  嘗試  对每个条目进行迭代  获取条目中的 'hid', 'subway', 和 'bus' 字段  # 查找对应的Property节点  查找具有 'hid' 的 Property 节点  如果未找到该节点  顯示 "Property {hid} not found in the database." 并跳过该条目  # 创建或获取每个Subway节点并创建关系  对 'subway' 列表中的每个地铁站进行迭代  创建或获取 Subway 节点  使用 merge 操作避免重复创建  创建 Property 节点与 Subway 节点之间的 NEAR\_SUBWAY 关系  将此关系写入Neo4j  顯示 "Created relationship: {hid} -> {subway\_station} (Subway)"  # 创建或获取每个Bus节点并创建关系  对 'bus' 列表中的每个公交站进行迭代  创建或获取 Bus 节点  使用 merge 操作避免重复创建  创建 Property 节点与 Bus 节点之间的 NEAR\_BUS 关系  将此关系写入Neo4j  顯示 "Created relationship: {hid} -> {bus\_stop} (Bus)"  顯示 "All transportation data successfully integrated into Neo4j."  捕獲錯誤  如果处理数据时出错，顯示 "Error: {錯誤訊息}" | | | |