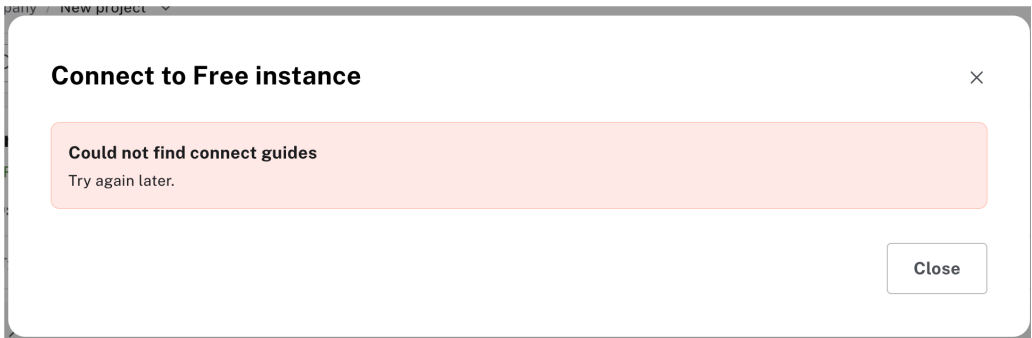


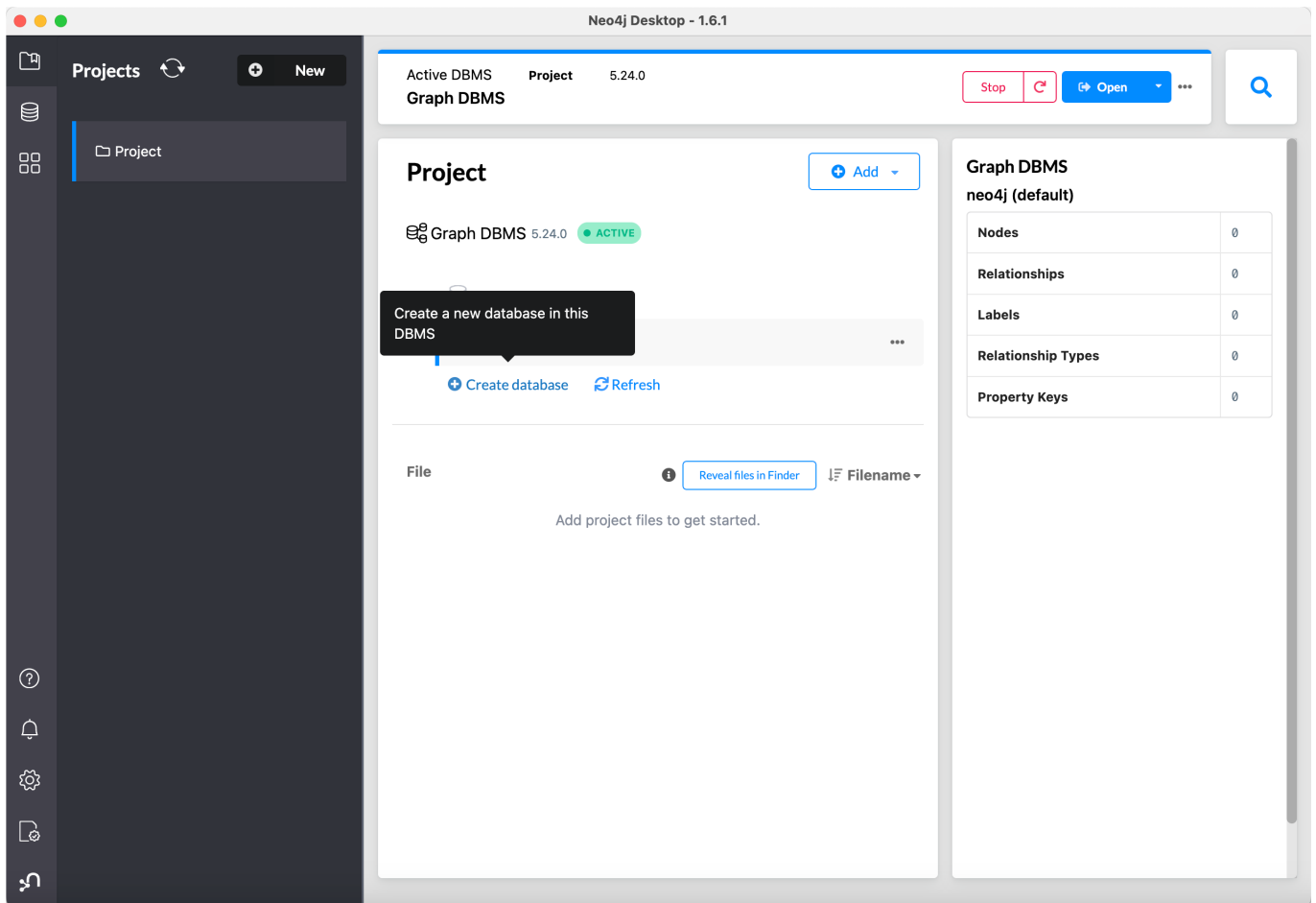
Neo4j 使用方式介绍

如何创建一个 Neo4j 服务器？

- 云端 Neo4j Aura Free 由于一些奇怪的原因无法正常连接，并且免费版云端 Neo4j 连接设备数量似乎有上限，不利于我们的项目，我建议使用本地的 Neo4j 数据库，部署在一个服务器上，然后在服务器上开放端口允许其他设备连接。



- 为了在服务器上提供 Neo4j 数据库服务，需要先下载 Neo4j 服务端 <https://neo4j.com/download/> 然后创建一个新 project:



- 这个 project 会在本机的多个端口进行服务，其使用的端口是：(7687 用于查询，7474 用于展示和监控)

Neo4j 默认用到的主要端口是这些：

用途	协议	端口
Bolt（客户端连接）	TCP	7687
HTTP（管理界面）	TCP	7474
HTTPS（管理界面SSL版）	TCP	7473

- 如果使用拥有公网 IP 的服务器，那么接下来可以直接使用 IP + 端口访问 neo4j 服务
- 如果在本地可以使用 localhost:7474 访问本地的 neo4j 服务
- 如果在本地也可以使用 NAT 穿越（我已经实现过）将端口开放至公网允许任意电脑访问。

如何使用 Python 访问 Neo4j 服务？

Neo4j 提供了官方的 python API 服务，首先需要使用 pip 下载 neo4j 包

```
pip install neo4j
```

之后可以使用 `URI`、`username` 和 `password` 访问 Neo4j 服务。

例如以下的代码可以测试本地部署的 Neo4j 服务：

其使用的 URI 为 `"bolt://localhost:7687"`
username 和 password 为 `("neo4j", "Microrust")`

```
from neo4j import GraphDatabase

URI = "bolt://localhost:7687"
AUTH = ("neo4j", "Microrust")

with GraphDatabase.driver(URI, auth=AUTH) as driver:
    driver.verify_connectivity()
    print("Connection established.")

    with driver.session() as session:
        # 插入一些 Person 节点（如果之前已经插入过，这里可以跳过）
        people = ["Alice", "Bob", "Charlie", "Diana"]
        for name in people:
            session.run(
                "MERGE (p:Person {name: $name})", # 用 MERGE，避免重复插入
                name=name
            )
        print("Nodes inserted.")

        # 插入一些关系：KNOWS
        relationships = [
```

```

        ("Alice", "Bob"),
        ("Bob", "Charlie"),
        ("Charlie", "Diana")
    ]
    for person1, person2 in relationships:
        session.run(
            """
            MATCH (a:Person {name: $person1})
            MATCH (b:Person {name: $person2})
            MERGE (a)-[:KNOWS]->(b)
            """,
            person1=person1,
            person2=person2
        )
    print("Relationships created.")

# 查询并打印所有 Person 节点及他们认识的人
result = session.run(
    """
    MATCH (a:Person)-[:KNOWS]->(b:Person)
    RETURN a.name AS from, b.name AS to
    """
)
print("Relationships in database:")
for record in result:
    print(f"{record['from']} KNOWS {record['to']}")

```