

Como usar eneo4j.py

Execução no Linux:

```
executar ./eneo4j.py > comandos_sql.txt
```

Esse comando irá fazer a extração dos dados das bases e o datacleaning. O resultado é escrito no arquivo de texto comandos_sql.txt que possui todos os comandos para inserir esses dados no banco de dados relacional. Para criar esse banco, é necessário executar outro arquivo explicado abaixo.

Além disso, essa execução gera outro arquivo de texto (comandos_cypher.txt), que será usado para inserção de dados no banco de grafos.

Como usar mysql_criacao.py

Utilizar o MySQL no Linux:

```
executar ./mysql_criacao.py
```

Antes, deve-se modificar a linha desse arquivo:

```
db = MySQLdb.connect(host="localhost", user="root", passwd="root", db="projetoBD")
```

Os valores dessas variáveis dependem de como foram definidas na criação do MySQL.

Ao executar a linha acima, irá pedir um arquivo, que é o comandos_sql (gerado pelo arquivo eneo4j explicado acima), por isso é recomendado que o deixe na mesma pasta do arquivo mysql_criacao.py. Esse arquivo possui todos os comandos para criação das tabelas e da inserção dos dados.

Pronto, o banco de dados relacional já está criado, e pode-se fazer as consultas acessando ele.

Como usar banconeo4j.py

Executar no Windows:

```
python banconeo4j.py
```

É necessário ter o arquivo comandos_cypher.txt na mesma pasta desse arquivo e deixar estabelecida a conexão com o Neo4J. Esse comando irá inserir todos os nós e relações.

Como usar equivalencia_nut.py

O arquivo é executado no Windows para efetuar a análise do grafo com a interface do Neo4j no próprio browser. Portanto basta executar o comando na linha de comando:

```
python equivalencia_nut.py
```

Ele recupera todos os nutrientes do grafo, cria vetores de alimentos e as respectivas

quantidades para cada nutriente e assim gera comandos cypher para criar as relações de equivalencia e executa os comandos utilizando a API da biblioteca py2neo.

Como usar lista_alimentos.py

Depois que o banco já está criado no Neo4j, no Windows, basta executar o comando:

```
python lista_alimentos.py
```

Assim, as relações serão criadas no banco.

Como usar histogram_data.py

Basta executar:

```
python histogram_data.py > arquivo.txt
```

O histograma foi gerado com os dados desse arquivo, com uma ferramenta de análise