1.

**QUERY 6: seguendo la logica suggerita, ecco i risultati:

```
val results_1_3 = Graph.cypher("MATCH (dis1:disease {names:'multiple myeloma'})-[r:BIO_VALUE_HIGH]->(drg1:drug) WITH dis1, drg1 MATCH (dis1)-[r:BIO_VALUE_HIGH]->(dis2:disease) RETURN dis1.names, drg1.name, dis2.names LIMIT 1")


val results_1_4 = Graph.cypher("MATCH (dis1:disease {names:'multiple myeloma'})-[r:BIO_VALUE_HIGH]->(drg1:drug) WITH dis1, drg1 MATCH (dis1)-[r:BIO_VALUE_HIGH]->(m)-[c:BIO_VALUE_HIGH]->(dis2:disease) RETURN dis1.names, drg1.names, m.names, dis2.names LIMIT 1")


val results_2_3 = Graph.cypher("MATCH (dis1:disease {names:'multiple myeloma'})-[r:BIO_VALUE_HIGH]->(n)-[c:BIO_VALUE_HIGH]->(drg1:drug) WITH dis1, n, drg1 MATCH (dis1)-[r:BIO_VALUE_HIGH]->(dis2:disease) RETURN dis1.names, n.names, drg1.name, dis2.names LIMIT 1 ")


val results_2_4 = Graph.cypher("MATCH (dis1:disease {names:'multiple myeloma'})-[r:BIO_VALUE_HIGH]->(n)-[c:BIO_VALUE_HIGH]->(drg1:drug) WITH dis1, n, drg1 MATCH (dis1)-[r:BIO_VALUE_HIGH]->(m)-[c:BIO_VALUE_HIGH]->(dis2:disease) RETURN dis1.names, n.names, drg1.names, m.names, dis2.names LIMIT 1 ")

val results_final= results_1_3 + " " + results_1_4 + " " + results_2_3 + " " + results_2_4
```

2.

3 Query → OTTIENE RISULTATO

Sostituendo la variabile p, il problema viene risolto

PRIMA: MATCH p=(dis:disease)-[:BIO_VALUE_HIGH]->(gen:gene) WHERE dis.names CONTAINS 'io' RETURN p LIMIT 10;

DOPO: val results_3_query = Graph.cypher("MATCH (dis:disease)-[r:BIO_VALUE_HIGH]->(gen:gene) WHERE dis.names CONTAINS 'io' RETURN dis.name, gen.name LIMIT 10;")

3.

Script richiesto:

- Link video tutorial: https://drive.google.com/file/d/1MIbktOPqobpocxUwrDVYPggoaAc6L_nN/view?usp=sharing

- Link allo script: https://drive.google.com/file/d/18zMhjY11xCunt_7bGa_oL7LOHS-VZHtk/view?usp=sharing