Name: Prem Kumar Yadav
Jaguar ID: J01280187

To run the program:

```
# Create virtual environment
python -m venv venv
# Activate virtual environment (on mac)
source venv/bin/activate

#Install the requirements
pip install -r requirements.txt

# Start Neo4j using Docker Compose
docker-compose up -d

# After 30s,  Check if it's running
docker-compose ps
```

Then, Open browser and
http://localhost:7474
Login ID: neo4j
Password: password

Then run the command to get thw results:

```
python3 deadcode.py ./sample_code –output sample_code_results.txt
```

Cypher Queries

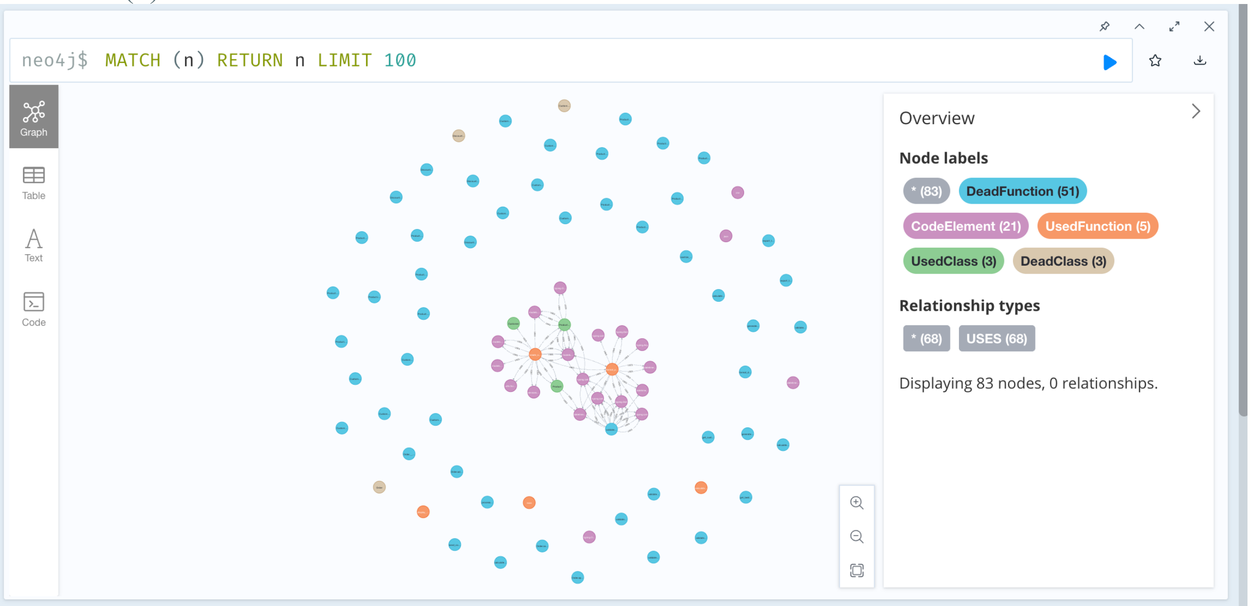
If output (Graph DB) does not appears color coded then run the below queries:

```
// Set GREEN for used functions
MATCH (n:CodeElement)
WHERE n.type = 'function' AND n.is_used = true
SET n:UsedFunction
REMOVE n:CodeElement;

// Set RED for dead functions
MATCH (n:CodeElement)
WHERE n.type = 'function' AND n.is_used = false
SET n:DeadFunction
REMOVE n:CodeElement;

// Set BLUE for used classes
MATCH (n:CodeElement)
WHERE n.type = 'class' AND n.is_used = true
SET n:UsedClass
REMOVE n:CodeElement;

// Set RED for dead classes
MATCH (n:CodeElement)
WHERE n.type = 'class' AND n.is_used = false
SET n:DeadClass
REMOVE n:CodeElement;
```

Then Use these Cypher Queries to view the results:

1. To view graph database of the sample code

```
MATCH (n) RETURN n LIMIT 100
```

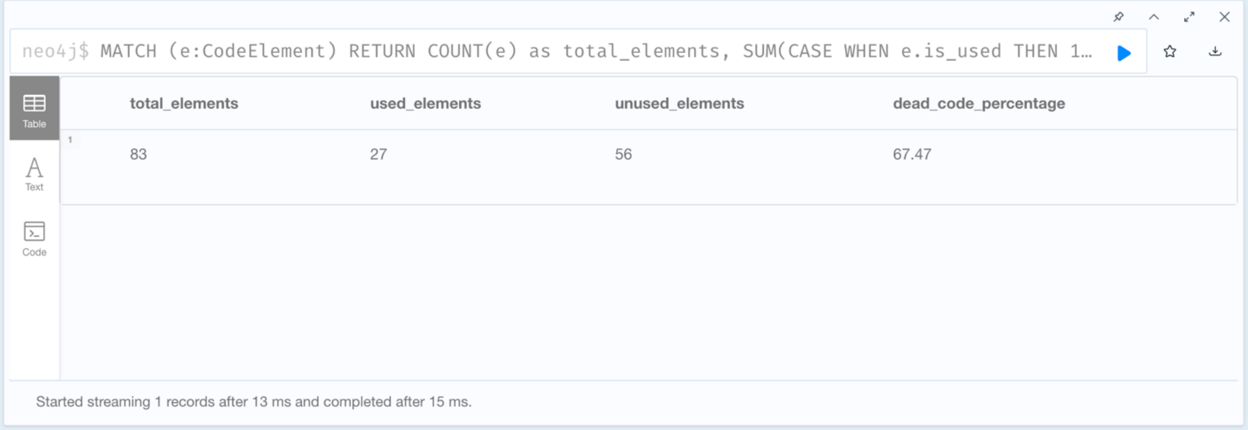


```
neo4j$ MATCH (n) RETURN n LIMIT 100
```

Displaying 83 nodes, 0 relationships.

2. Get overall statistics of the codebase

```
MATCH (e:CodeElement)
RETURN
COUNT(e) as total_elements,
SUM(CASE WHEN e.is_used THEN 1 ELSE 0 END) as used_elements,
SUM(CASE WHEN e.is_used THEN 0 ELSE 1 END) as unused_elements,
ROUND(100.0 * SUM(CASE WHEN e.is_used THEN 0 ELSE 1 END) / COUNT(e), 2) as
dead_code_percentage;
```



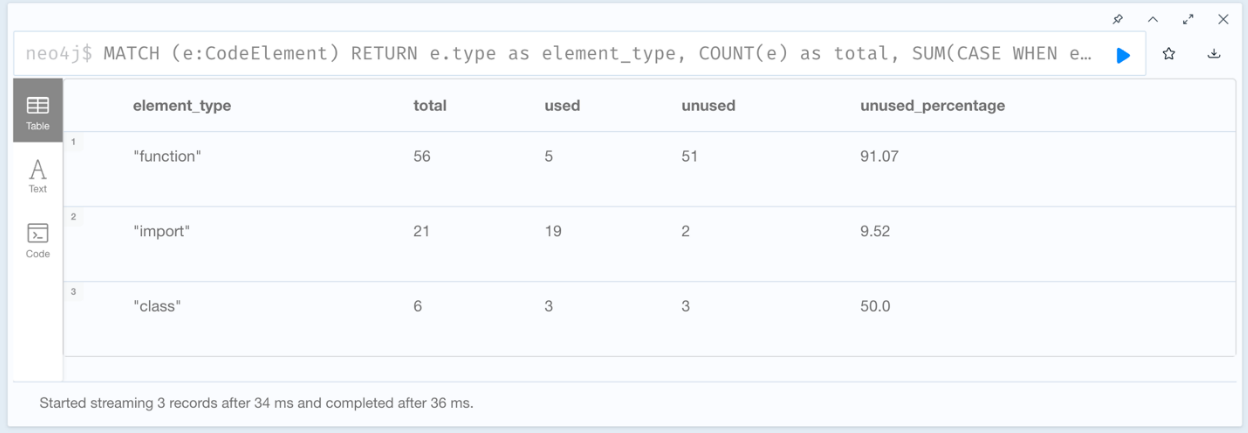
```
neo4j$ MATCH (e:CodeElement) RETURN COUNT(e) as total_elements, SUM(CASE WHEN e.is_used THEN 1…
```

	total_elements	used_elements	unused_elements	dead_code_percentage
1	83	27	56	67.47

Started streaming 1 records after 13 ms and completed after 15 ms.

3. Break down statistics by functions, classes, imports

```
MATCH (e:CodeElement)
RETURN
e.type as element_type,
COUNT(e) as total,
SUM(CASE WHEN e.is_used THEN 1 ELSE 0 END) as used,
SUM(CASE WHEN e.is_used THEN 0 ELSE 1 END) as unused,
ROUND(100.0 * SUM(CASE WHEN e.is_used THEN 0 ELSE 1 END) / COUNT(e), 2) as unused_percentage
ORDER BY total DESC;
```



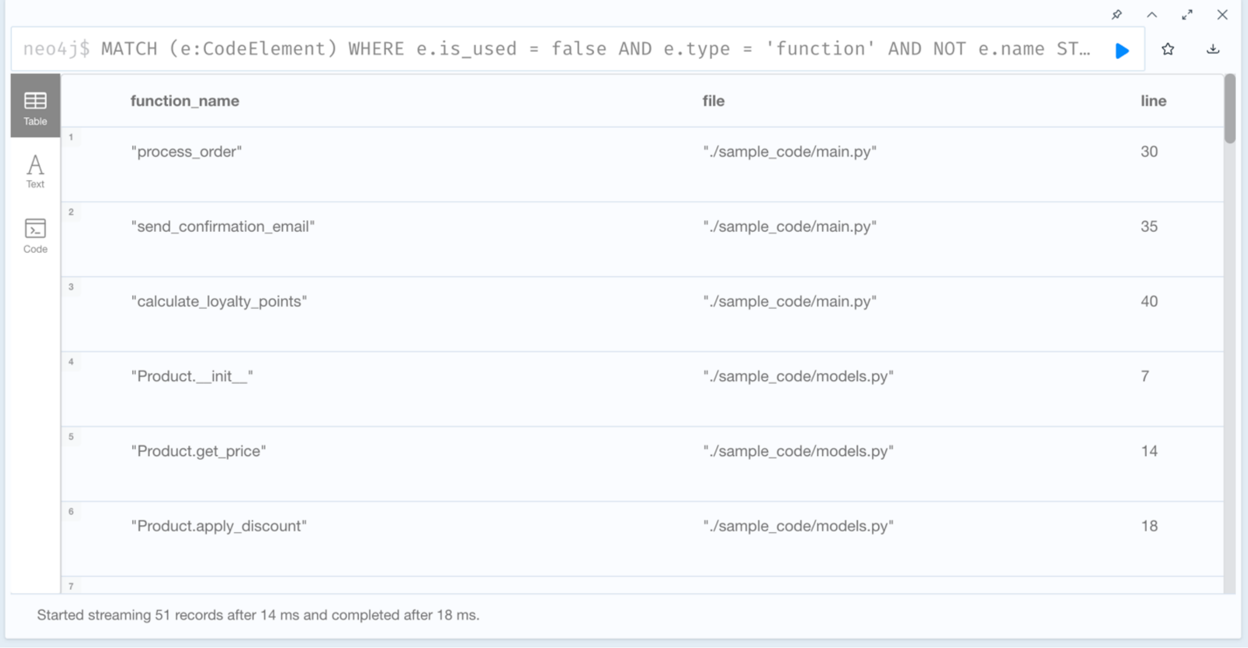
```
neo4j$ MATCH (e:CodeElement) RETURN e.type as element_type, COUNT(e) as total, SUM(CASE WHEN e…
```

	element_type	total	used	unused	unused_percentage
1	"function"	56	5	51	91.07
2	"import"	21	19	2	9.52
3	"class"	6	3	3	50.0

Started streaming 3 records after 34 ms and completed after 36 ms.

4. *List all unused functions with their locations*

```
MATCH (e:CodeElement)
WHERE e.is_used = false
AND e.type = 'function'
AND NOT e.name STARTS WITH '_'
AND NOT e.name IN ['main', '__init__']
RETURN
e.name as function_name,
e.file_path as file,
e.line_number as line
ORDER BY e.file_path, e.line_number;
```



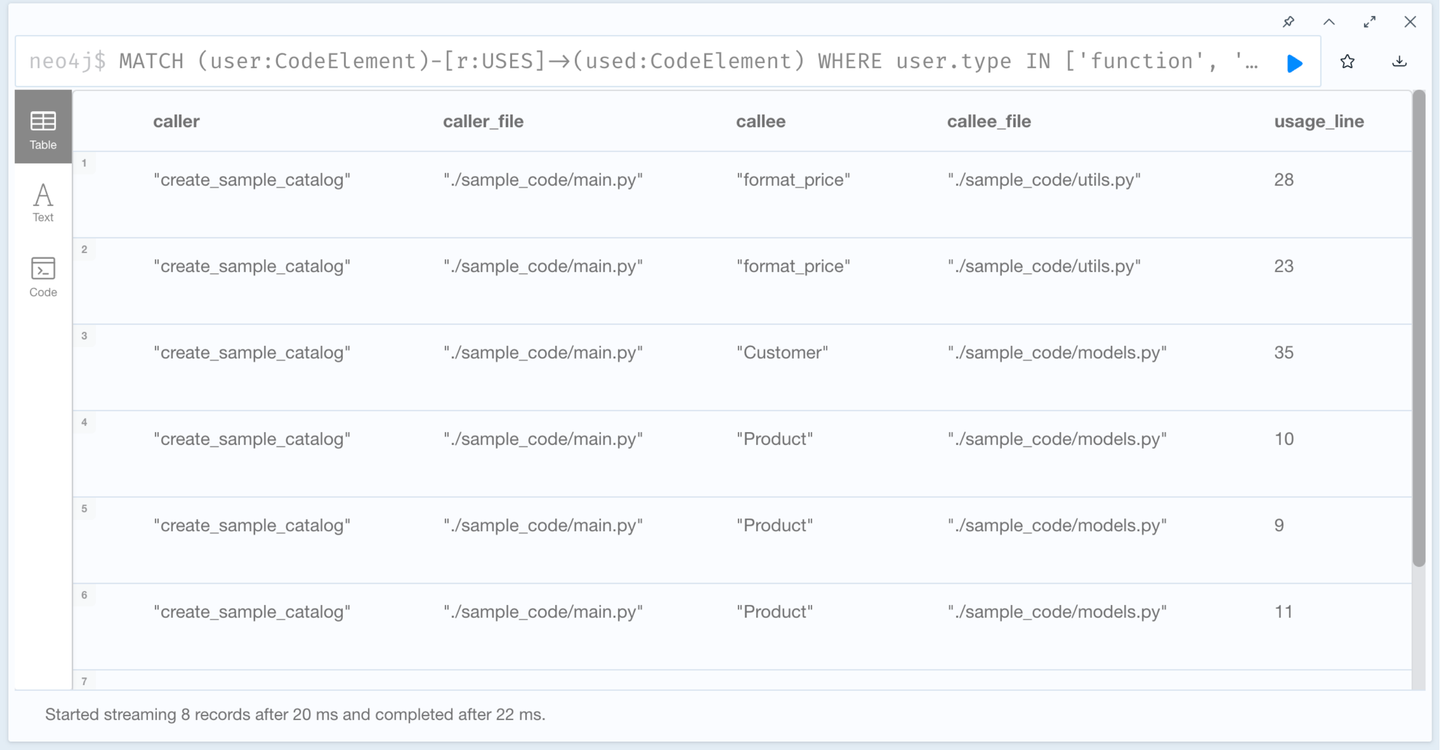
```
neo4j$ MATCH (e:CodeElement) WHERE e.is_used = false AND e.type = 'function' AND NOT e.name ST…
```

	function_name	file	line
1	"process_order"	"./sample_code/main.py"	30
2	"send_confirmation_email"	"./sample_code/main.py"	35
3	"calculate_loyalty_points"	"./sample_code/main.py"	40
4	"Product.__init__"	"./sample_code/models.py"	7
5	"Product.get_price"	"./sample_code/models.py"	14
6	"Product.apply_discount"	"./sample_code/models.py"	18
7			

Started streaming 51 records after 14 ms and completed after 18 ms.

5. *Show who uses what (for graph visualization)*

```
MATCH (user:CodeElement)-[r:USES]->(used:CodeElement)
WHERE user.type IN ['function', 'class']
AND used.type IN ['function', 'class']
RETURN
user.name as caller,
user.file_path as caller_file,
used.name as callee,
used.file_path as callee_file, r.line_number as usage_line
LIMIT 50;
```



```
neo4j$ MATCH (user:CodeElement)-[r:USES]→(used:CodeElement) WHERE user.type IN ['function', '…
```

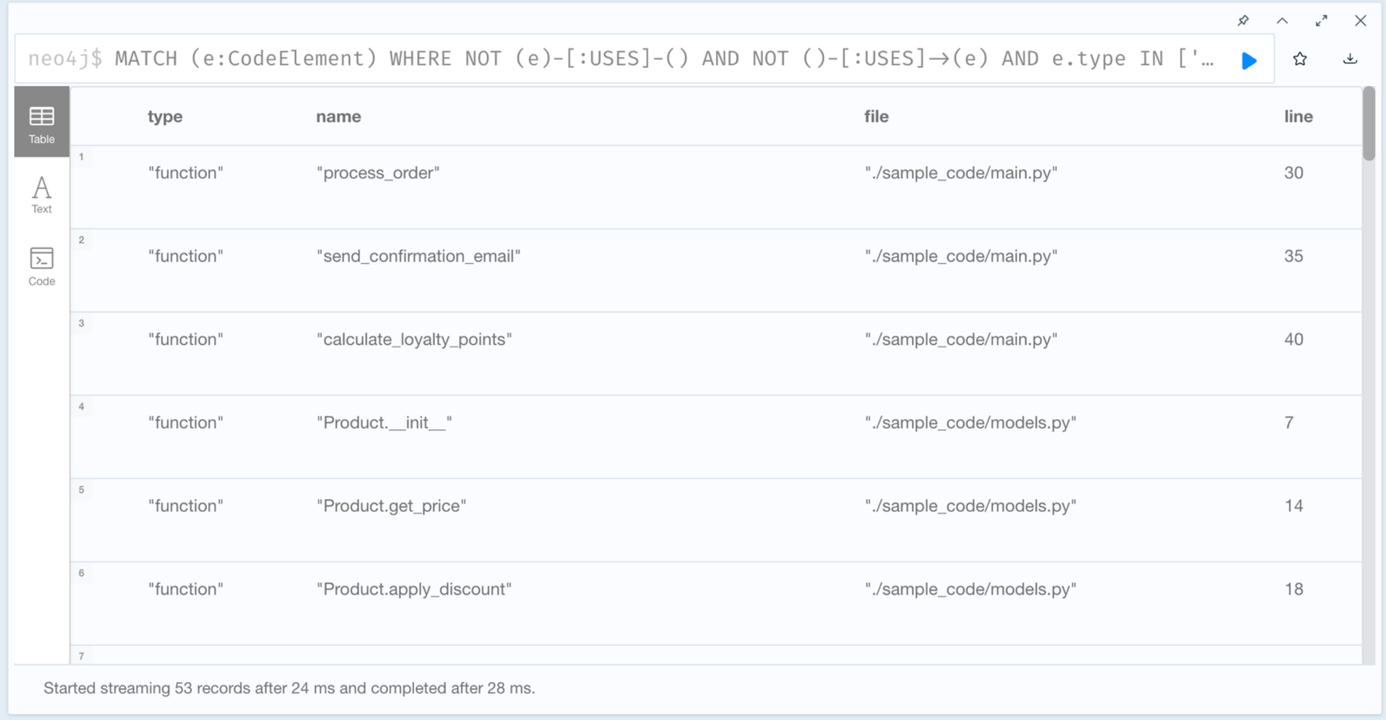
	caller	caller_file	callee	callee_file	usage_line
1	"create_sample_catalog"	"./sample_code/main.py"	"format_price"	"./sample_code/utils.py"	28
2	"create_sample_catalog"	"./sample_code/main.py"	"format_price"	"./sample_code/utils.py"	23
3	"create_sample_catalog"	"./sample_code/main.py"	"Customer"	"./sample_code/models.py"	35
4	"create_sample_catalog"	"./sample_code/main.py"	"Product"	"./sample_code/models.py"	10
5	"create_sample_catalog"	"./sample_code/main.py"	"Product"	"./sample_code/models.py"	9
6	"create_sample_catalog"	"./sample_code/main.py"	"Product"	"./sample_code/models.py"	11
7					

Started streaming 8 records after 20 ms and completed after 22 ms.

6. *Find code with no relationships (neither using nor being used)*

```
MATCH (e:CodeElement)
WHERE NOT (e)-[:USES]-()
AND NOT ()-[:USES]->(e)
AND e.type IN ['function', 'class']
AND e.is_used = false
RETURN
e.type as type,
e.name as name,
e.file_path as file,
e.line_number as line
```

```
ORDER BY e.file_path;
```



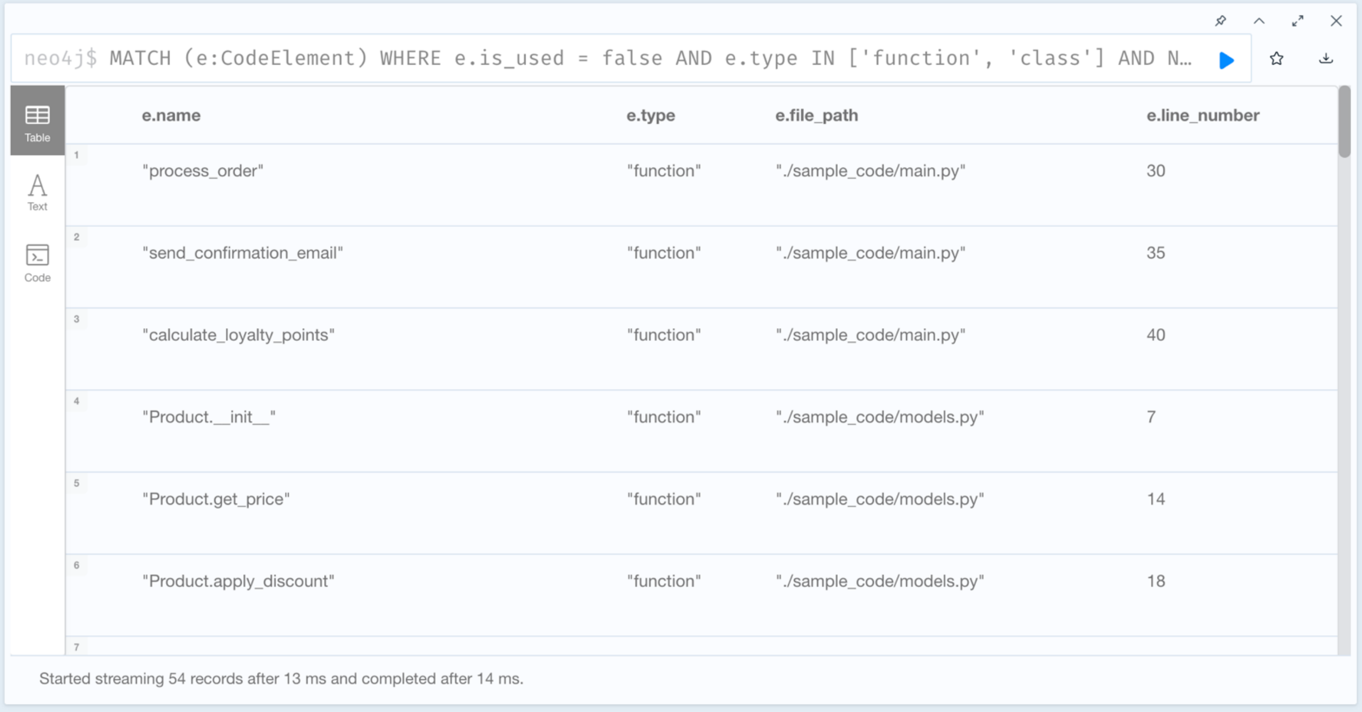
```
neo4j$ MATCH (e:CodeElement) WHERE NOT (e)-[:USES]-() AND NOT ()-[:USES]→(e) AND e.type IN ['…
```

	type	name	file	line
1	"function"	"process_order"	"./sample_code/main.py"	30
2	"function"	"send_confirmation_email"	"./sample_code/main.py"	35
3	"function"	"calculate_loyalty_points"	"./sample_code/main.py"	40
4	"function"	"Product.__init__"	"./sample_code/models.py"	7
5	"function"	"Product.get_price"	"./sample_code/models.py"	14
6	"function"	"Product.apply_discount"	"./sample_code/models.py"	18
7				

Started streaming 53 records after 24 ms and completed after 28 ms.

7. *Find all unused functions and classes*

```
MATCH (e:CodeElement)
WHERE e.is_used = false
AND e.type IN ['function', 'class']
AND NOT e.name STARTS WITH '_'
AND NOT e.name IN ['main', '__init__']
RETURN e.name, e.type, e.file_path, e.line_number
ORDER BY e.file_path, e.line_number;
```



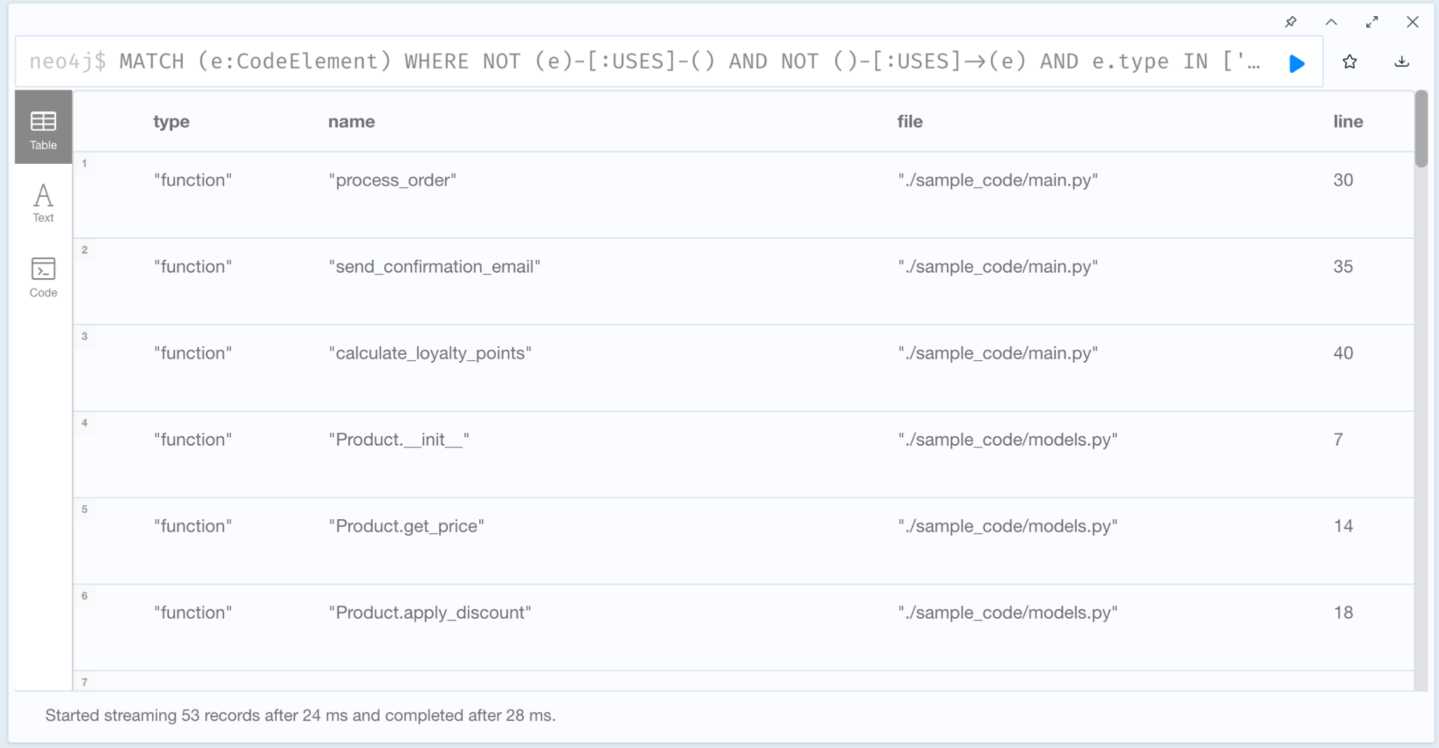
```
neo4j$ MATCH (e:CodeElement) WHERE e.is_used = false AND e.type IN ['function', 'class'] AND N…
```

	e.name	e.type	e.file_path	e.line_number
1	"process_order"	"function"	"./sample_code/main.py"	30
2	"send_confirmation_email"	"function"	"./sample_code/main.py"	35
3	"calculate_loyalty_points"	"function"	"./sample_code/main.py"	40
4	"Product.__init__"	"function"	"./sample_code/models.py"	7
5	"Product.get_price"	"function"	"./sample_code/models.py"	14
6	"Product.apply_discount"	"function"	"./sample_code/models.py"	18
7				

Started streaming 54 records after 13 ms and completed after 14 ms.

8. *Find orphaned code elements (no incoming or outgoing relationships)*

```
MATCH (e:CodeElement)
WHERE NOT (e)-[:USES]-() AND NOT ()-[:USES]->(e)
AND e.type IN ['function', 'class']
RETURN e.name, e.type, e.file_path, e.line_number
ORDER BY e.file_path;
```



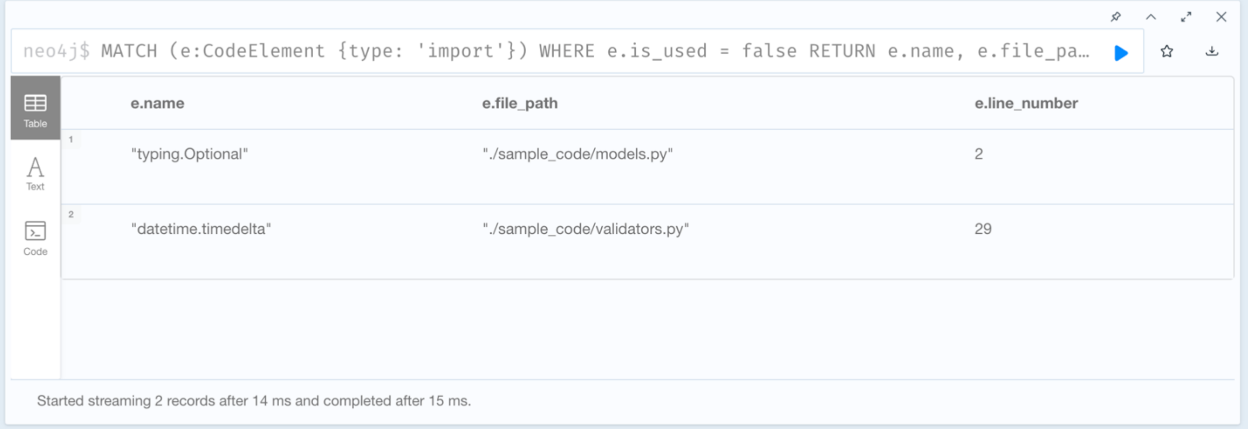
```
neo4j$ MATCH (e:CodeElement) WHERE NOT (e)-[:USES]-() AND NOT ()-[:USES]→(e) AND e.type IN ['…
```

type	name	file	line
"function"	"process_order"	"./sample_code/main.py"	30
"function"	"send_confirmation_email"	"./sample_code/main.py"	35
"function"	"calculate_loyalty_points"	"./sample_code/main.py"	40
"function"	"Product.__init__"	"./sample_code/models.py"	7
"function"	"Product.get_price"	"./sample_code/models.py"	14
"function"	"Product.apply_discount"	"./sample_code/models.py"	18

Started streaming 53 records after 24 ms and completed after 28 ms.

9. *Find unused imports*

```
MATCH (e:CodeElement {type: 'import'})
WHERE e.is_used = false
RETURN e.name, e.file_path, e.line_number
ORDER BY e.file_path, e.line_number;
```



```
neo4j$ MATCH (e:CodeElement {type: 'import'}) WHERE e.is_used = false RETURN e.name, e.file_pa…
```

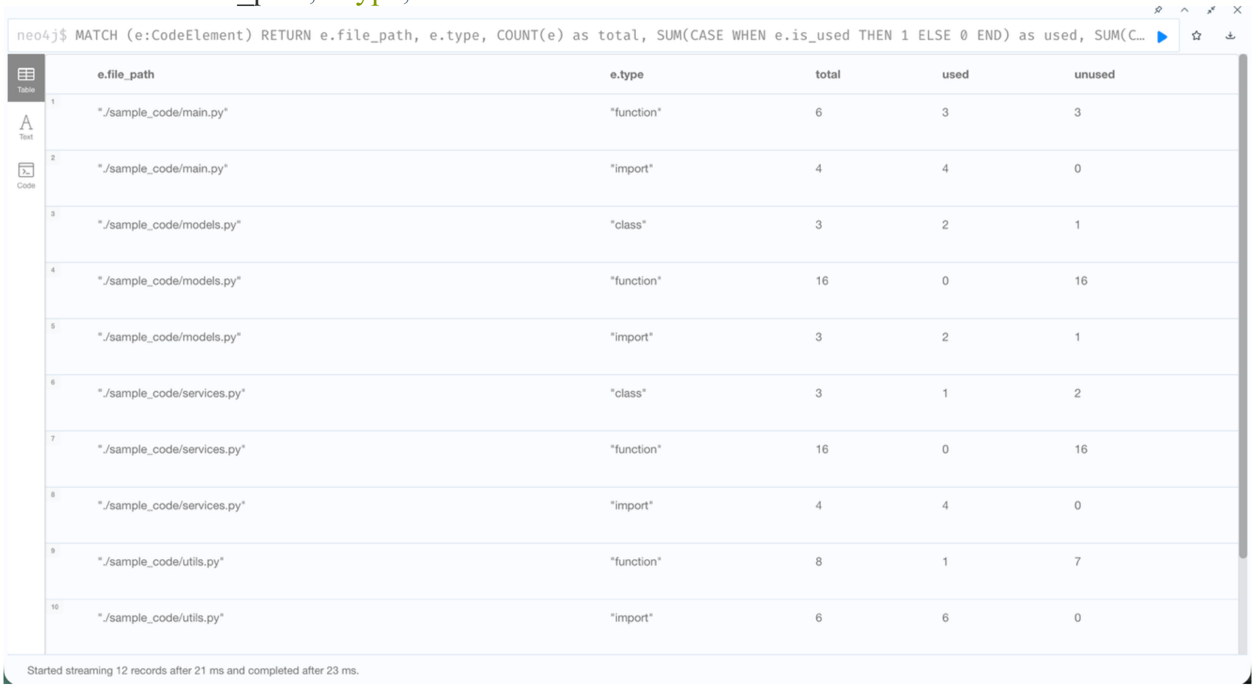
e.name	e.file_path	e.line_number
"typing.Optional"	"./sample_code/models.py"	2
"datetime.timedelta"	"./sample_code/validators.py"	29

Started streaming 2 records after 14 ms and completed after 15 ms.

10. *Get usage statistics by file*

```
MATCH (e:CodeElement)
RETURN e.file_path,
e.type,
COUNT(e) as total,
SUM(CASE WHEN e.is_used THEN 1 ELSE 0 END) as used,
SUM(CASE WHEN e.is_used THEN 0 ELSE 1 END) as unused
```

ORDER BY e.file_path, e.type;



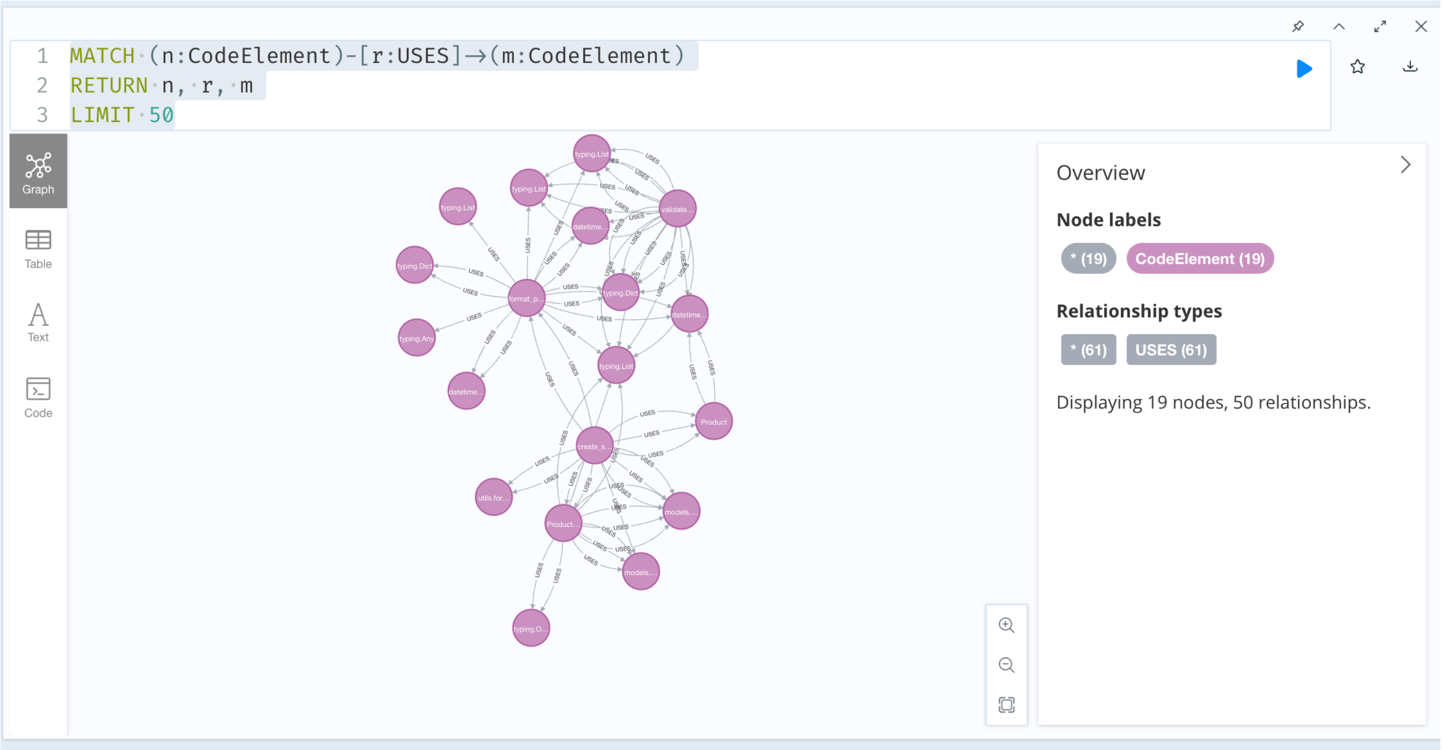
```
neo4j$ MATCH (e:CodeElement) RETURN e.file_path, e.type, COUNT(e) as total, SUM(CASE WHEN e.is_used THEN 1 ELSE 0 END) as used, SUM(C…
```

e.file_path	e.type	total	used	unused
"./sample_code/main.py"	"function"	6	3	3
"./sample_code/main.py"	"import"	4	4	0
"./sample_code/models.py"	"class"	3	2	1
"./sample_code/models.py"	"function"	16	0	16
"./sample_code/models.py"	"import"	3	2	1
"./sample_code/services.py"	"class"	3	1	2
"./sample_code/services.py"	"function"	16	0	16
"./sample_code/services.py"	"import"	4	4	0
"./sample_code/utils.py"	"function"	8	1	7
"./sample_code/utils.py"	"import"	6	6	0

Started streaming 12 records after 21 ms and completed after 23 ms.

11. View Nodes WITH Relationships

MATCH (n:CodeElement)-[r:USES]->(m:CodeElement)
RETURN n, r, m
LIMIT 50



```
MATCH (n:CodeElement)-[r:USES]→(m:CodeElement)
RETURN n, r, m
LIMIT 50
```

Overview

Node labels

* (19) CodeElement (19)

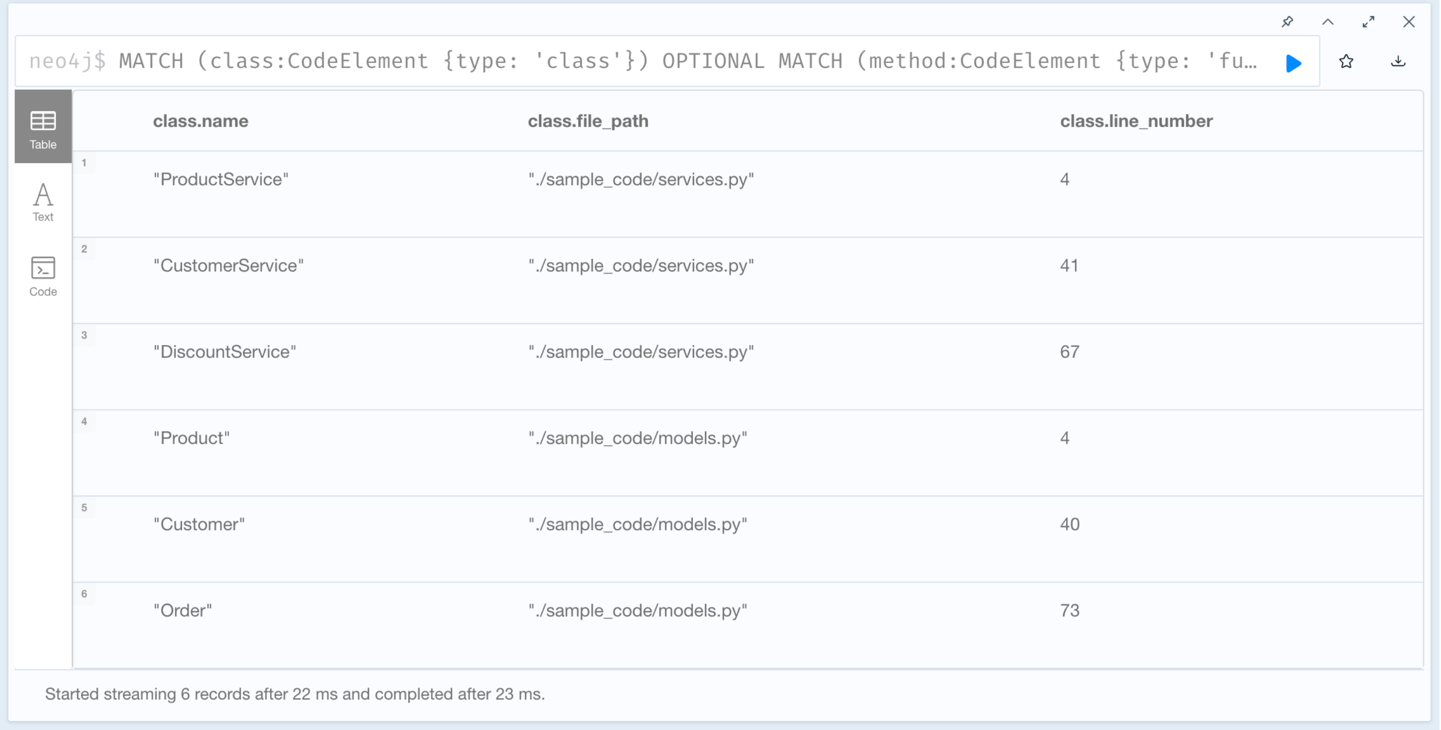
Relationship types

* (61) USES (61)

Displaying 19 nodes, 50 relationships.

12. *Find classes with no methods being used*

```
MATCH (class:CodeElement {type: 'class'})
OPTIONAL MATCH (method:CodeElement {type: 'function'})
WHERE method.name STARTS WITH class.name + '.'
AND method.is_used = true
WITH class, COUNT(method) as used_methods
WHERE used_methods = 0
RETURN class.name, class.file_path, class.line_number;
```



```
neo4j$ MATCH (class:CodeElement {type: 'class'}) OPTIONAL MATCH (method:CodeElement {type: 'fu…
```

	class.name	class.file_path	class.line_number
1	"ProductService"	"./sample_code/services.py"	4
2	"CustomerService"	"./sample_code/services.py"	41
3	"DiscountService"	"./sample_code/services.py"	67
4	"Product"	"./sample_code/models.py"	4
5	"Customer"	"./sample_code/models.py"	40
6	"Order"	"./sample_code/models.py"	73

Started streaming 6 records after 22 ms and completed after 23 ms.