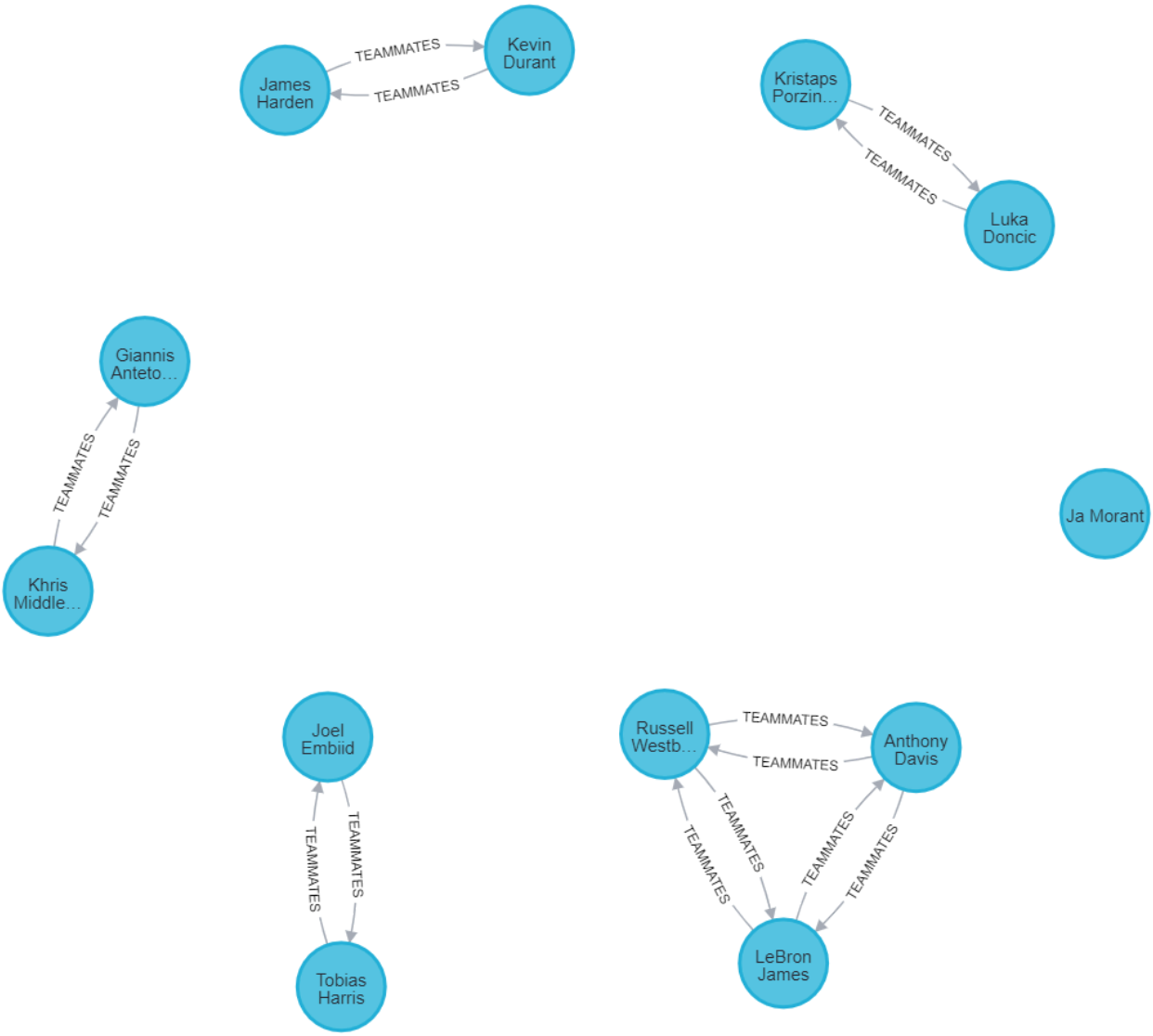
```
MATCH (n : PLAYER) RETURN (n)
```



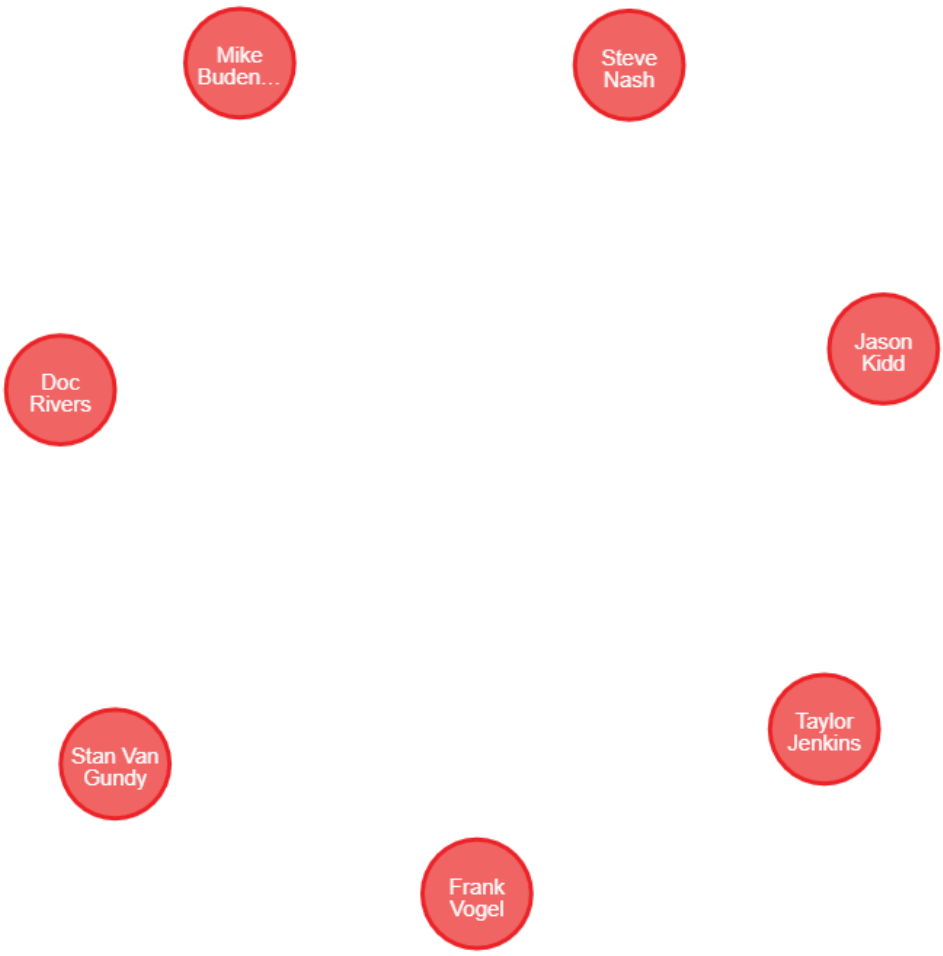
1. `MATCH (n:PLAYER)`

- This tells Neo4j to **look for nodes** that have the **label** `PLAYER`.
- `n` is a **variable** that represents each matching node.
- `PLAYER` is the **label** assigned to nodes (similar to a type or class — e.g., `PLAYER`, `COACH`, `TEAM`).

2. `RETURN n`

- This returns all the `PLAYER` nodes found in the `MATCH` clause.

```
MATCH (n : COACH) RETURN (n)
```



```
MATCH (n:PLAYER) RETURN n.name, n.height
```

	n.name	n.height
7	"Kevin Durant"	2.08
8	"James Harden"	1.96
9	"Giannis Antetokounmpo"	2.11
10	"Khris Middleton"	2.01
11	"Joel Embiid"	2.13
12	"Tobias Harris"	2.03

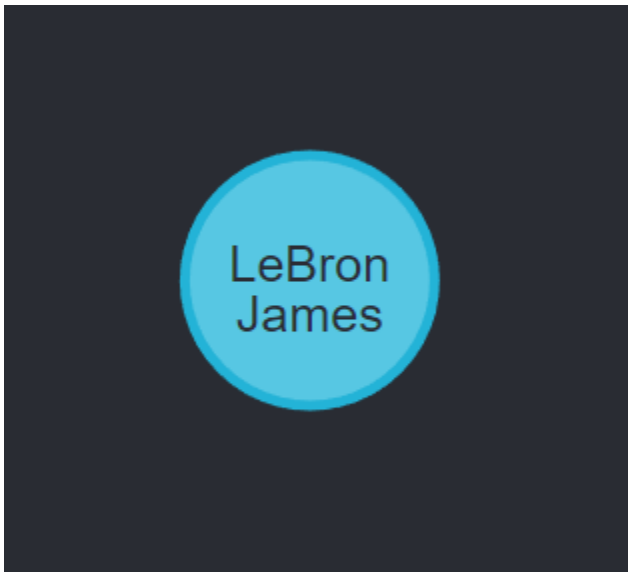
ted streaming 12 records after 5 ms and completed after 7 ms.

```
MATCH (n:PLAYER) RETURN n.name AS player_name, n.age AS age, n.number AS jersey_number
```

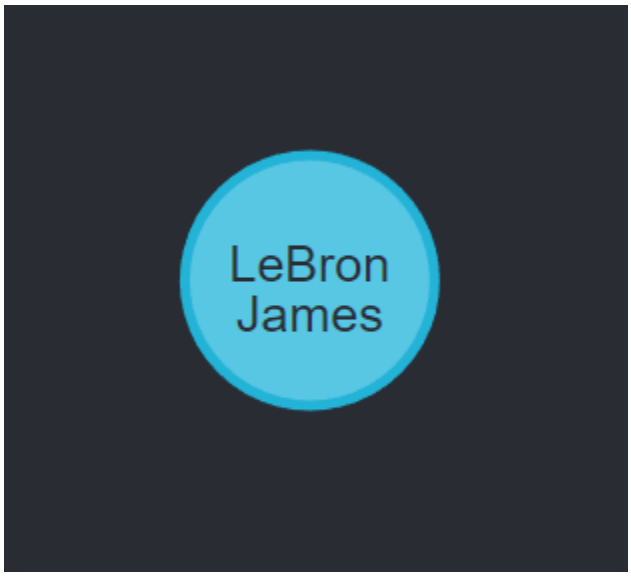
	player_name	age	jersey_number
7	"Kevin Durant"	33	7
8	"James Harden"	32	13
9	"Giannis Antetokounmpo"	26	34
10	"Khris Middleton"	30	22
11	"Joel Embiid"	27	21
12	"Tobias Harris"	29	22

ted streaming 12 records after 5 ms and completed after 6 ms.

```
MATCH (n:PLAYER) WHERE n.name = 'LeBron James' RETURN (n)
```



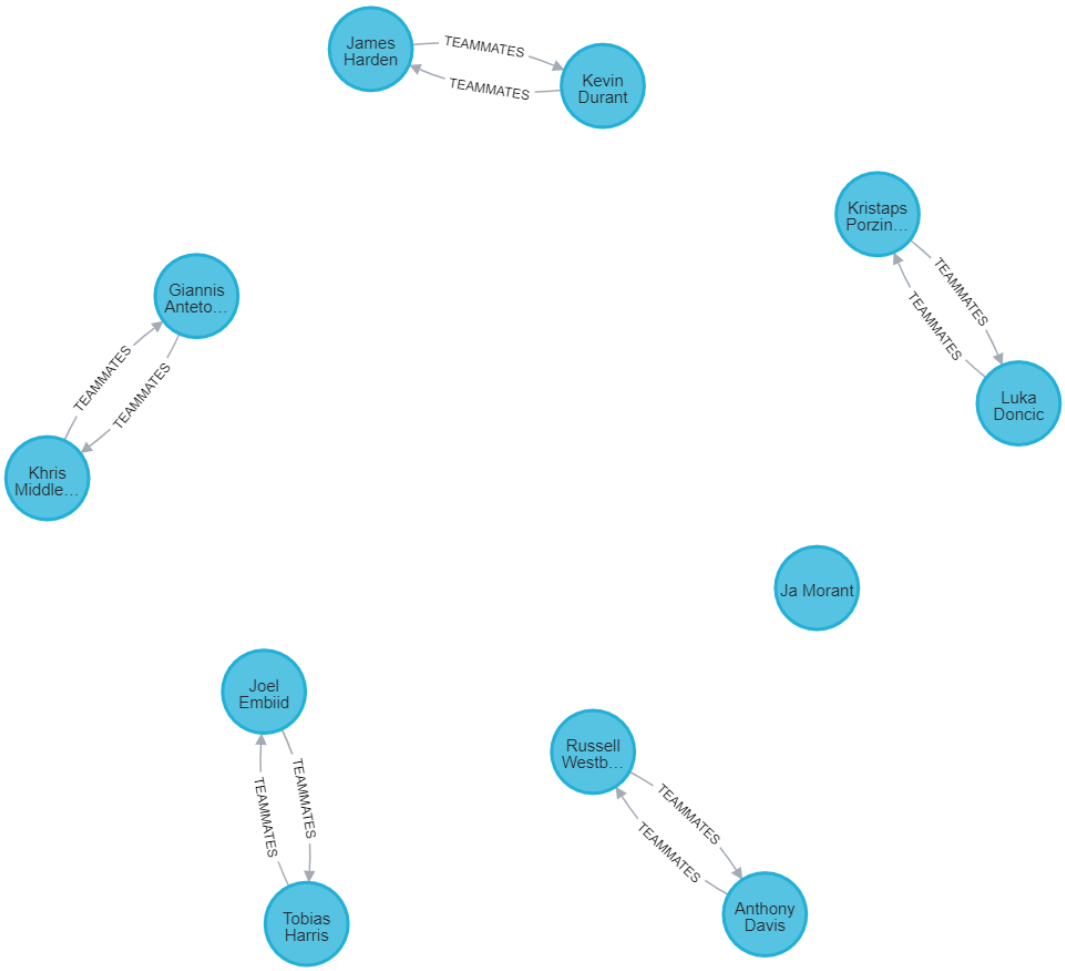

```
MATCH (n:PLAYER {name : "LeBron James" , height : 2.06})  RETURN (n)
```



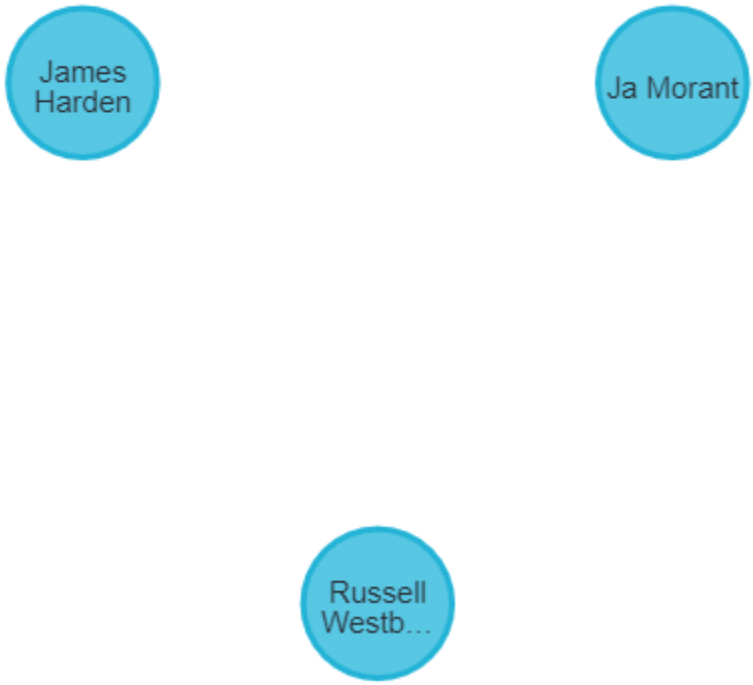
`MATCH (n:PLAYER ...)`: Finds a node with the label `PLAYER` and the exact properties:

- `name` = `"LeBron James"`
- `height` = `2.06`

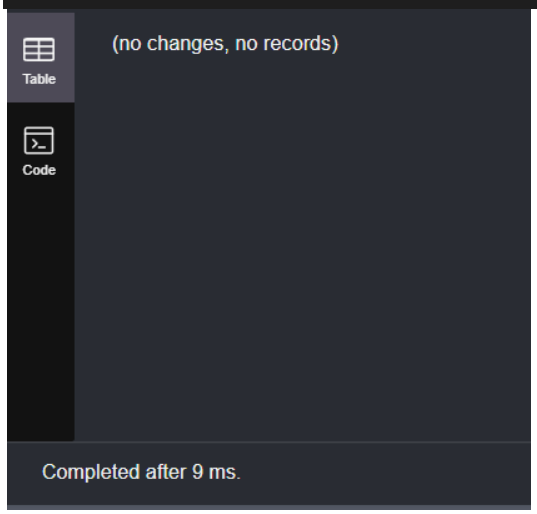
```
MATCH (n:PLAYER) WHERE n.name <> "LeBron James" RETURN (n)
```



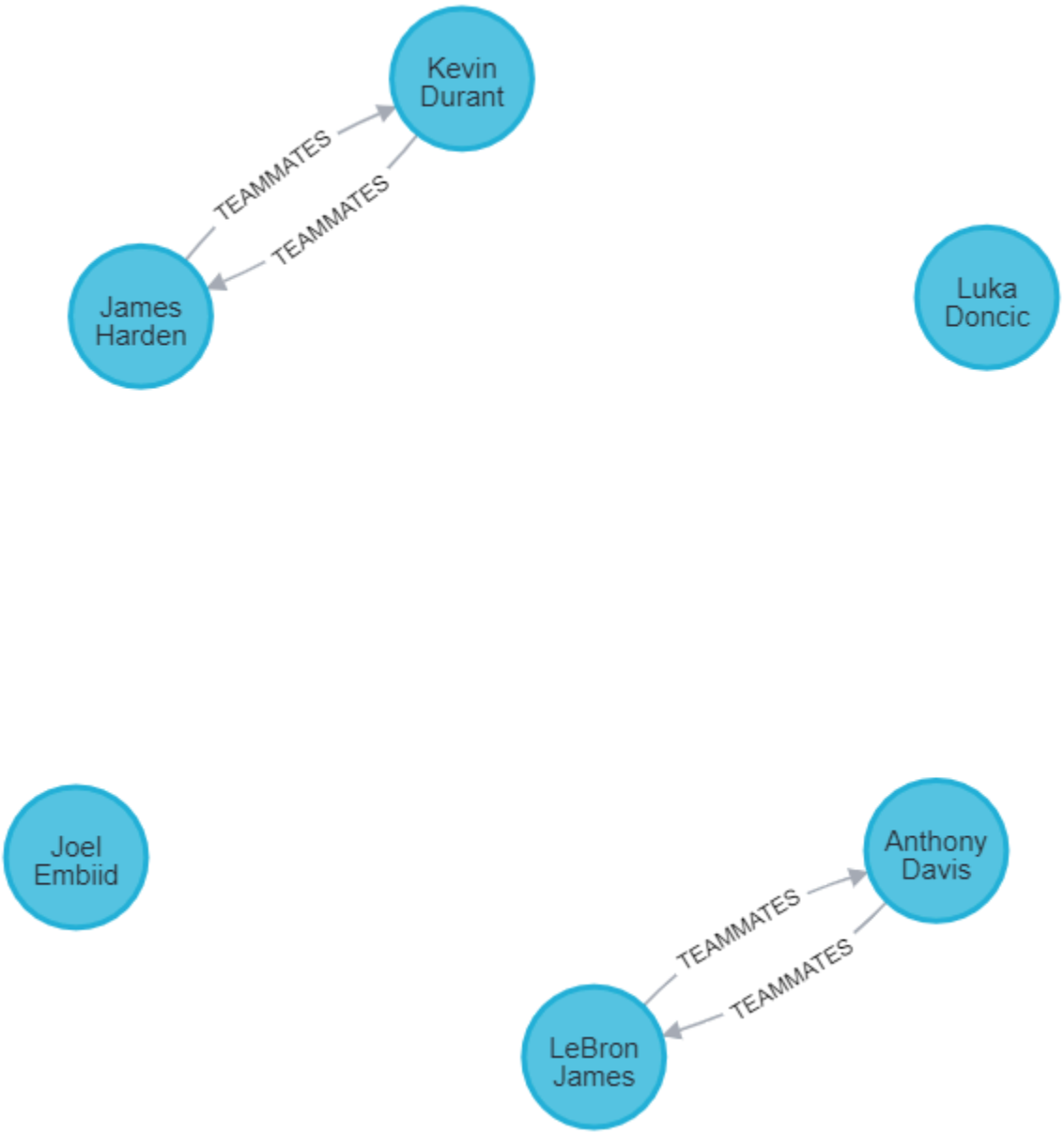
```
MATCH (n:PLAYER) WHERE n.height < 2  RETURN (n)
```



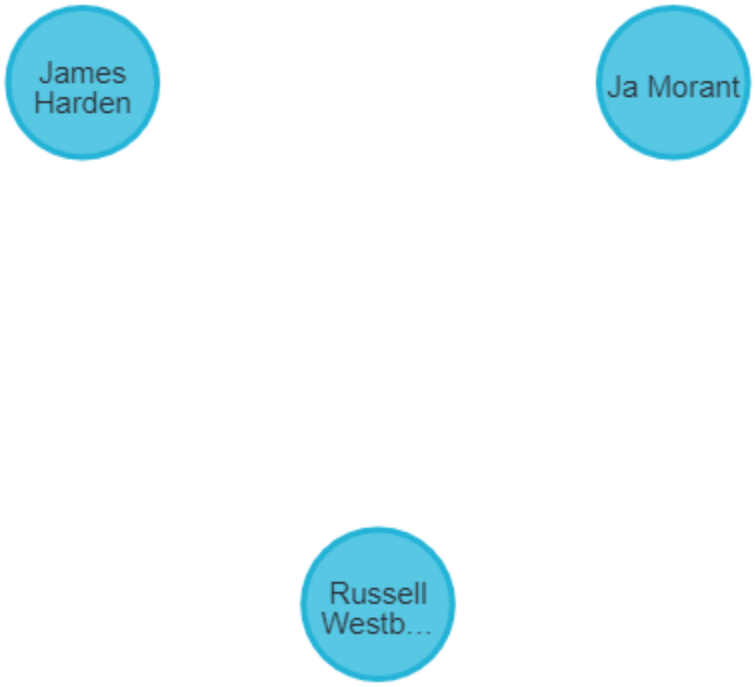

```
MATCH (n:PLAYER) WHERE n.height / ((n.weight) * (n.weight)) > 25 RETURN (n)
```



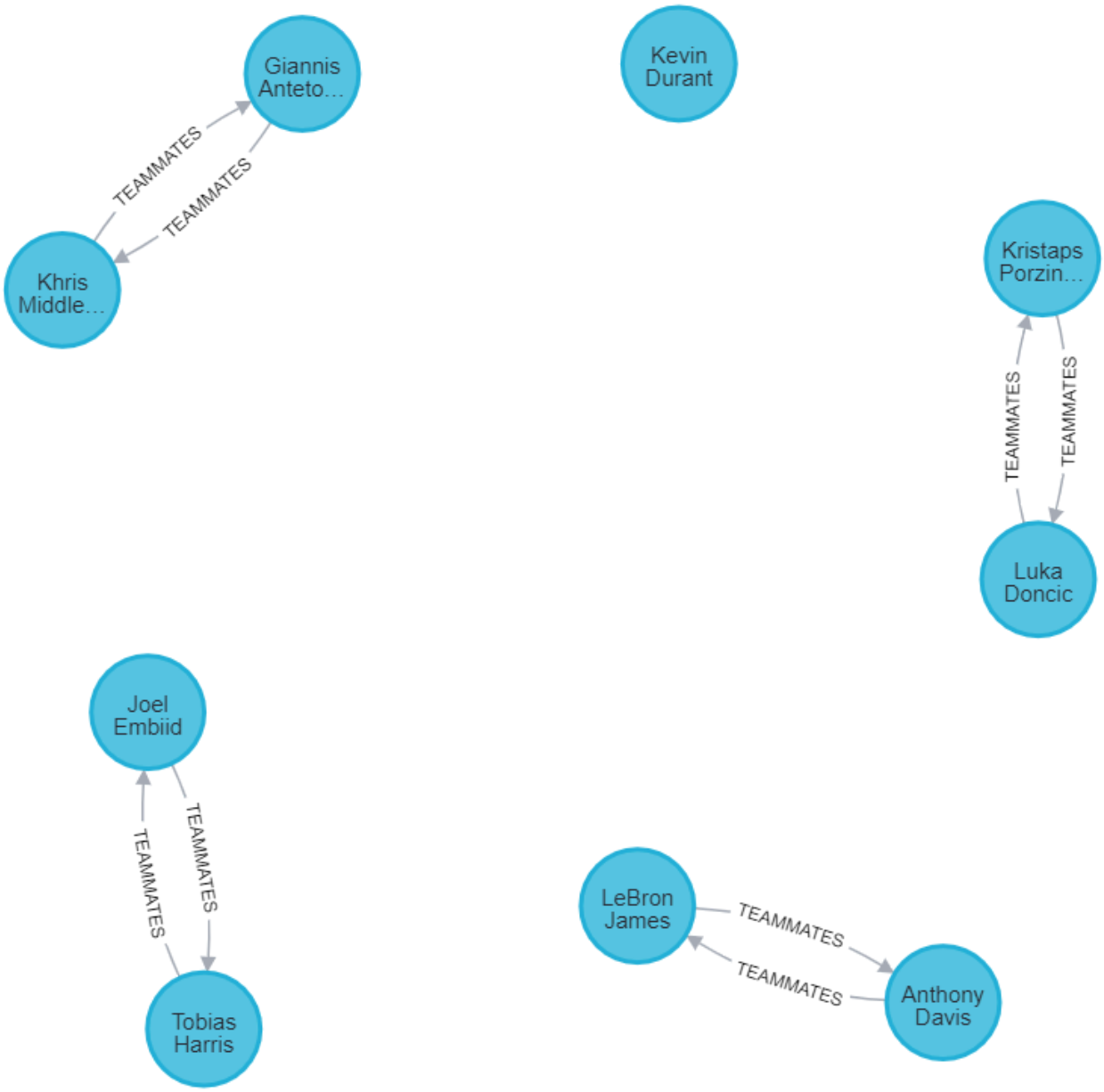
```
MATCH (n:PLAYER) WHERE n.weight / ((n.height) * (n.height)) > 25 RETURN (n)
```



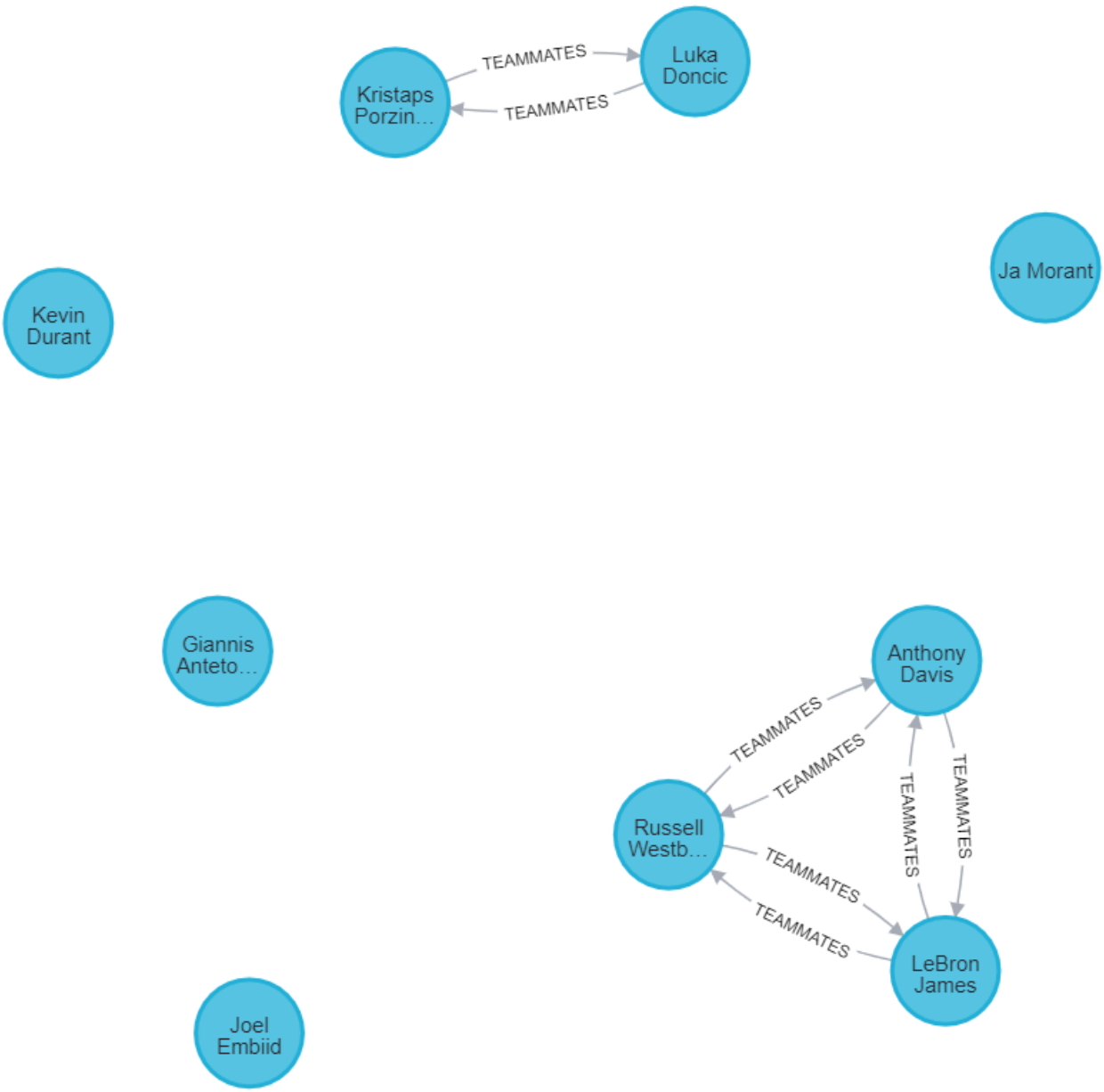
```
MATCH (n:PLAYER) WHERE n.weight > 50 AND n.height < 2 RETURN (n)
```



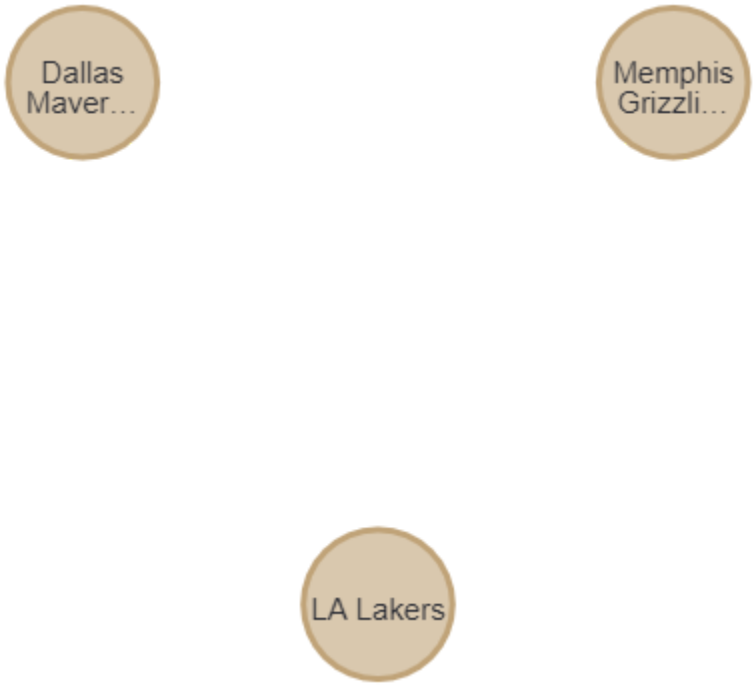
```
MATCH (n:PLAYER) WHERE n.weight > 50 AND n.height >= 2 RETURN (n)
```



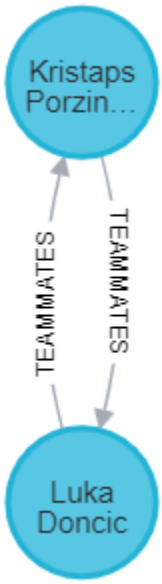
```
MATCH (n:PLAYER) WHERE NOT n.weight = 100 RETURN (n)
```



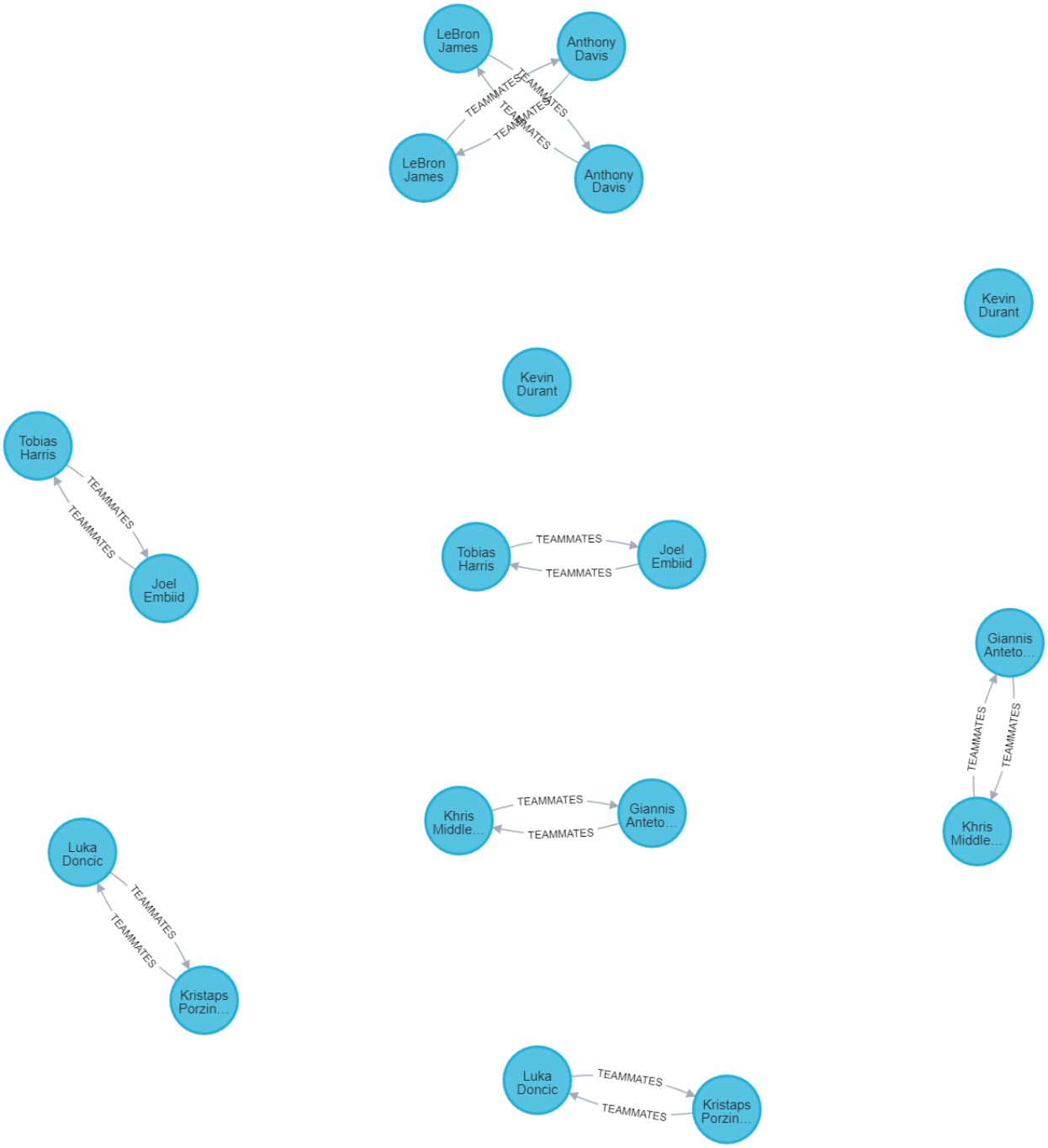
```
MATCH (n:TEAM) RETURN (n) LIMIT 3
```



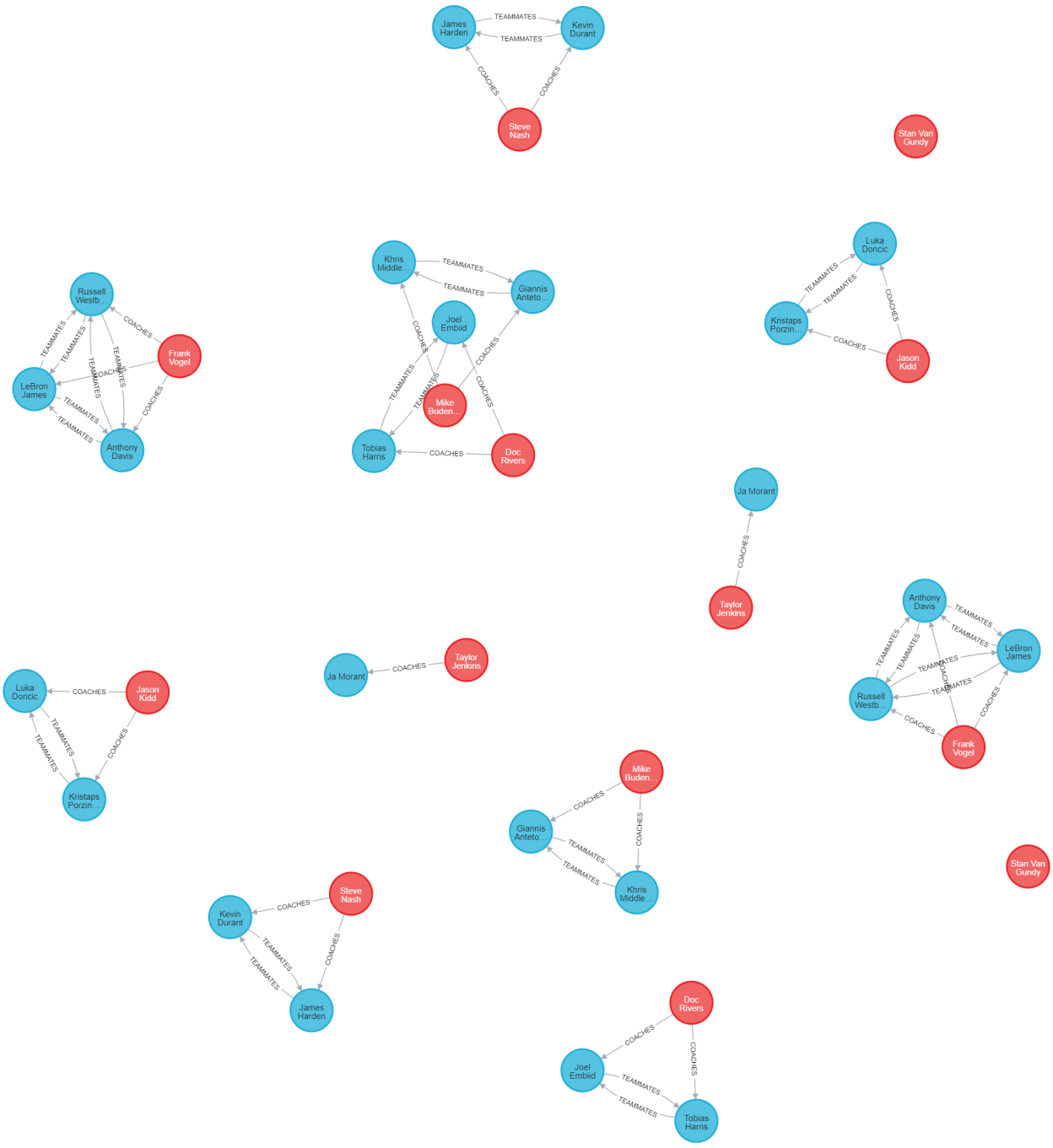

```
MATCH (n:PLAYER) WHERE n.height >= 2 RETURN (n) SKIP 2 limit 2
```



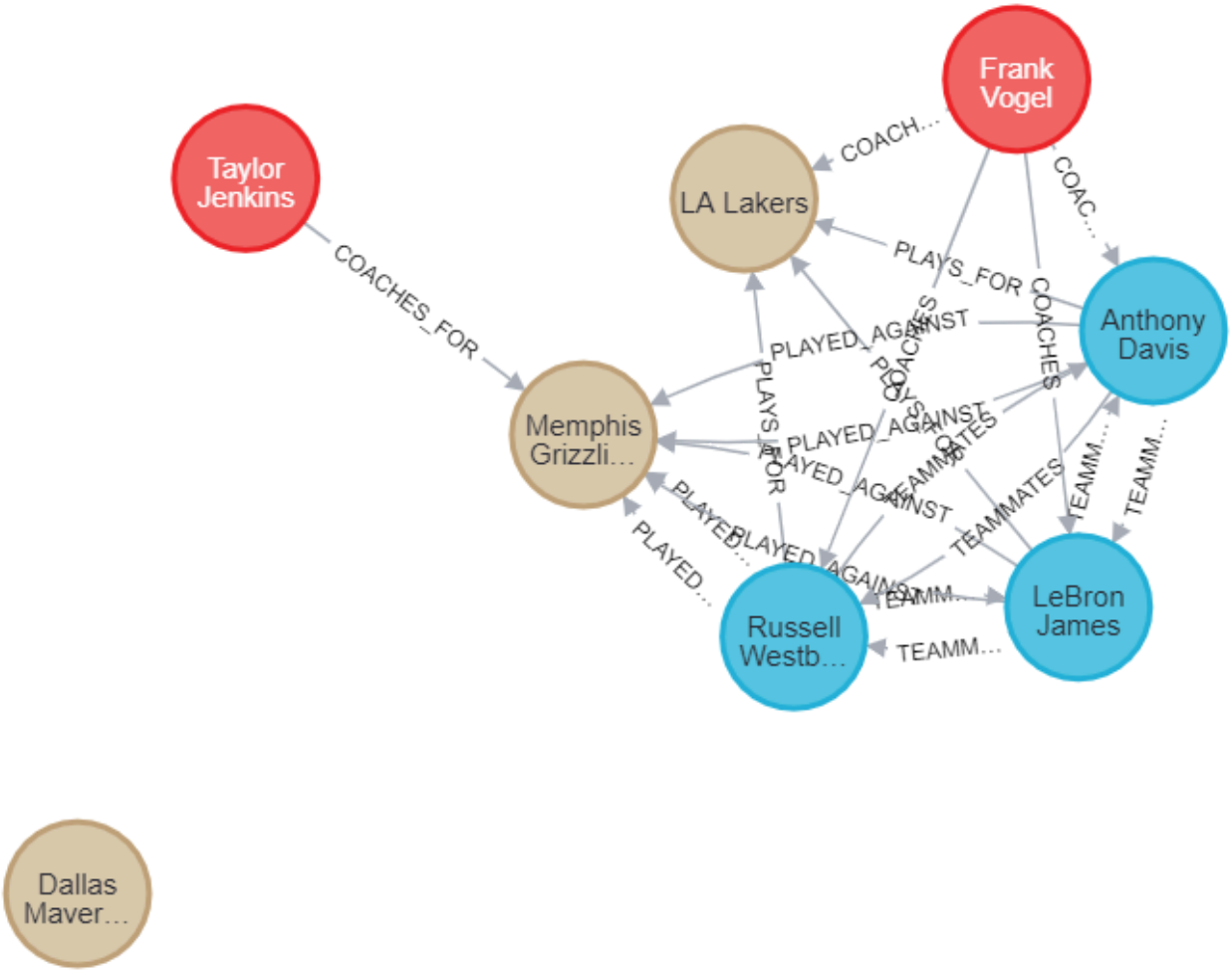
```
MATCH (n:PLAYER) WHERE n.height >= 2 RETURN n ORDER BY n.height DESC
```



```
MATCH (n:PLAYER), (p:COACH) RETURN (n),(p)
```

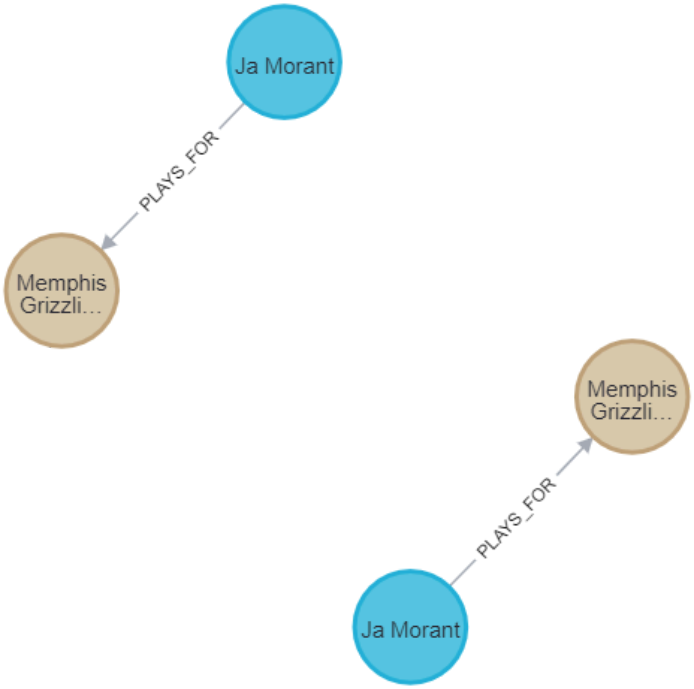


```
MATCH (p:PLAYER)
RETURN p AS entity
LIMIT 3
UNION ALL
MATCH (c:COACH)
RETURN c AS entity
LIMIT 2
UNION ALL
MATCH (t:TEAM)
RETURN t AS entity
LIMIT 3
```

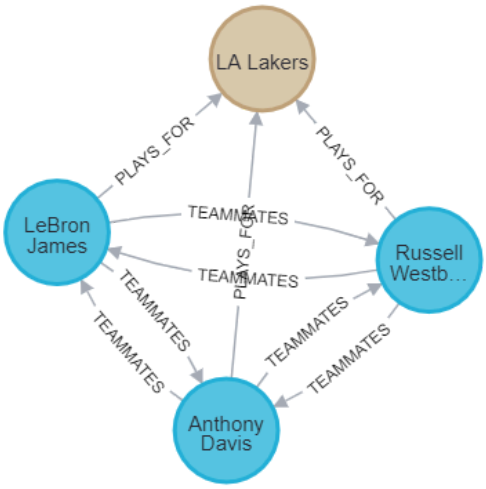


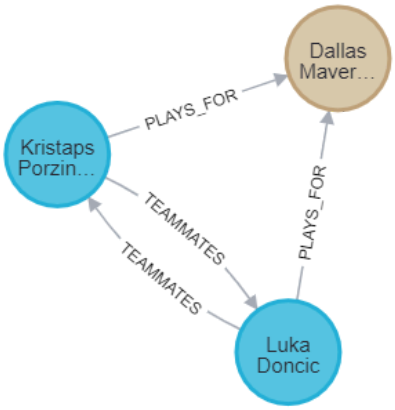
```
MATCH (p:PLAYER) -[:PLAYS_FOR]-> (t:TEAM) WHERE p.name = "Ja Morant" RETURN p
```

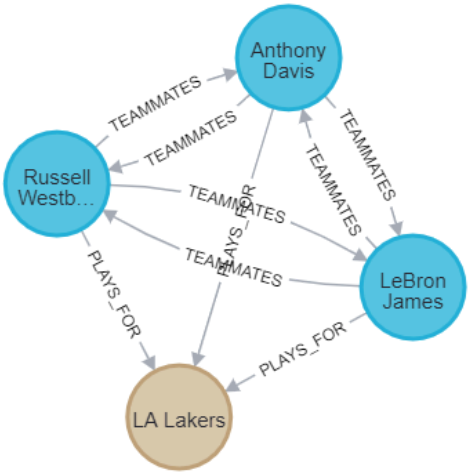
```
MATCH (p:PLAYER) -[:PLAYS_FOR]-> (t:TEAM) WHERE p.name = "Ja Morant" RETURN p,t
```



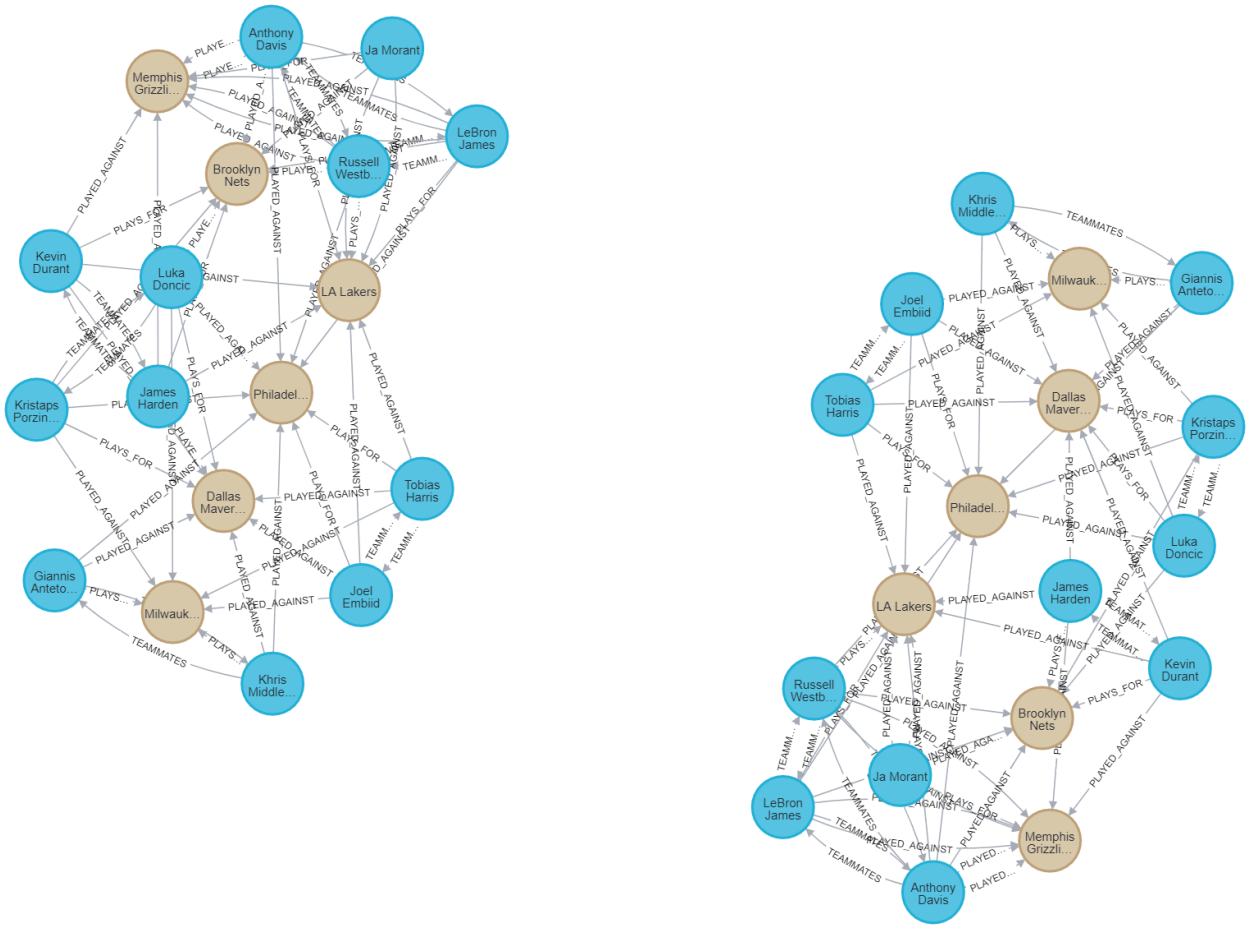
```
MATCH (p:PLAYER) -[:PLAYS_FOR]-> (t:TEAM) WHERE t.name = "LA Lakers" OR t.name = "Dallas Mavericks" RETURN p,t
```



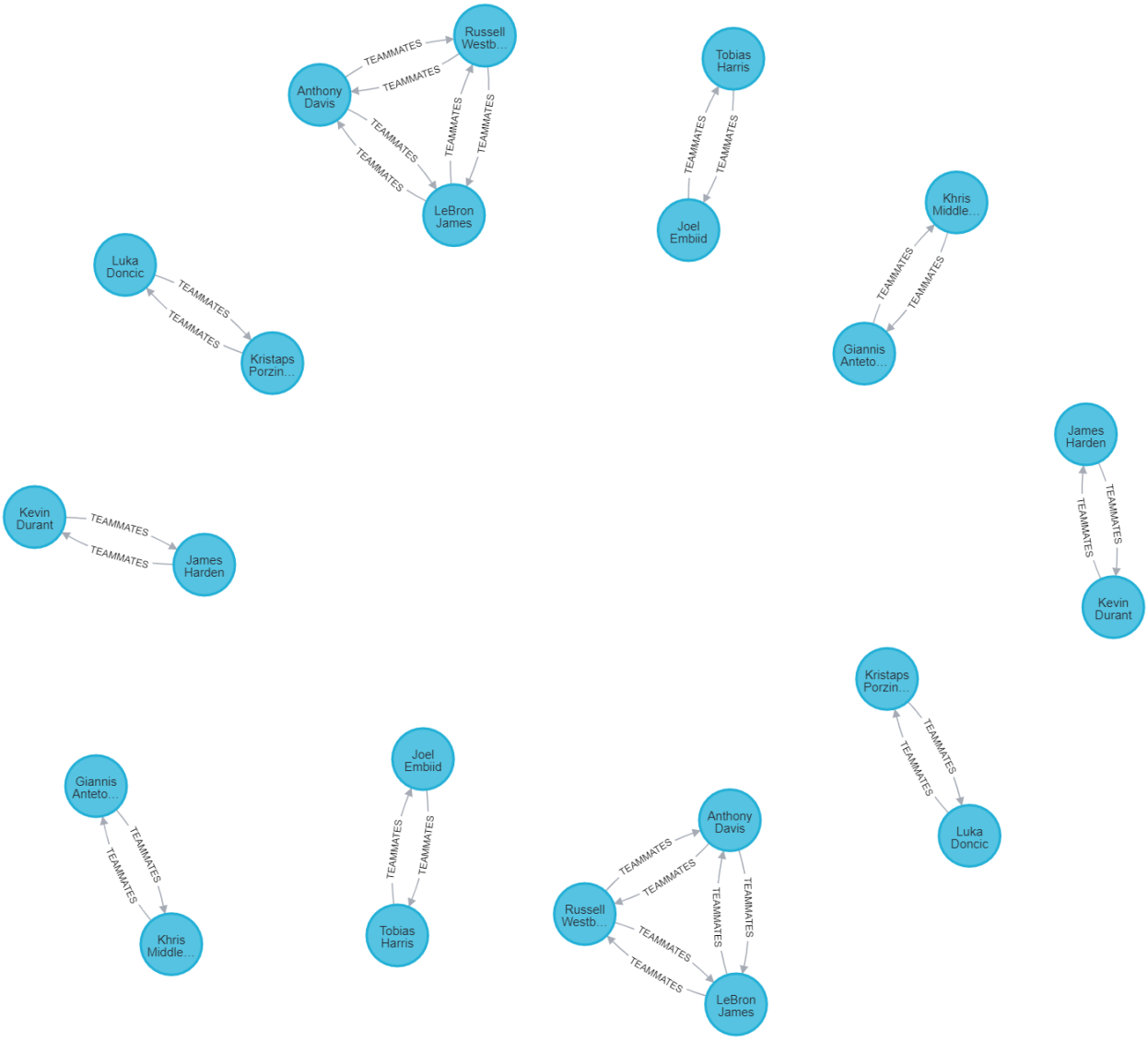




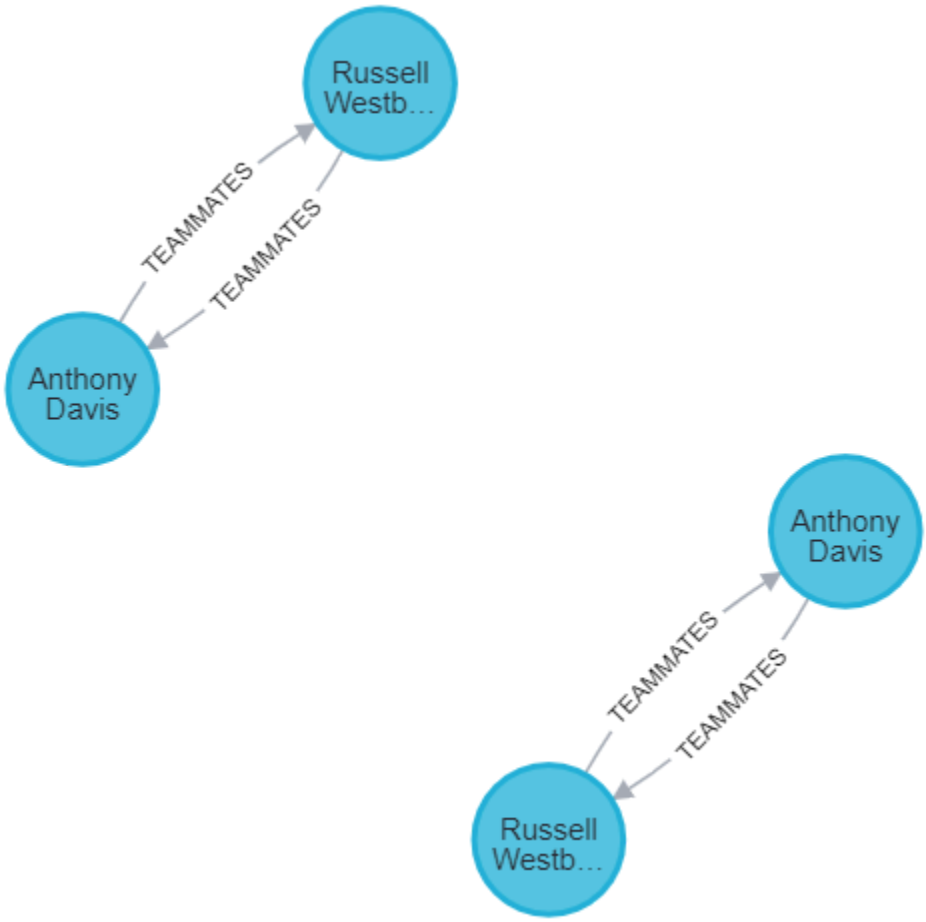
```
MATCH (p:PLAYER) -[contract : PLAYS_FOR]-> (t:TEAM) WHERE contract.salary >= 50000
RETURN p,t
```



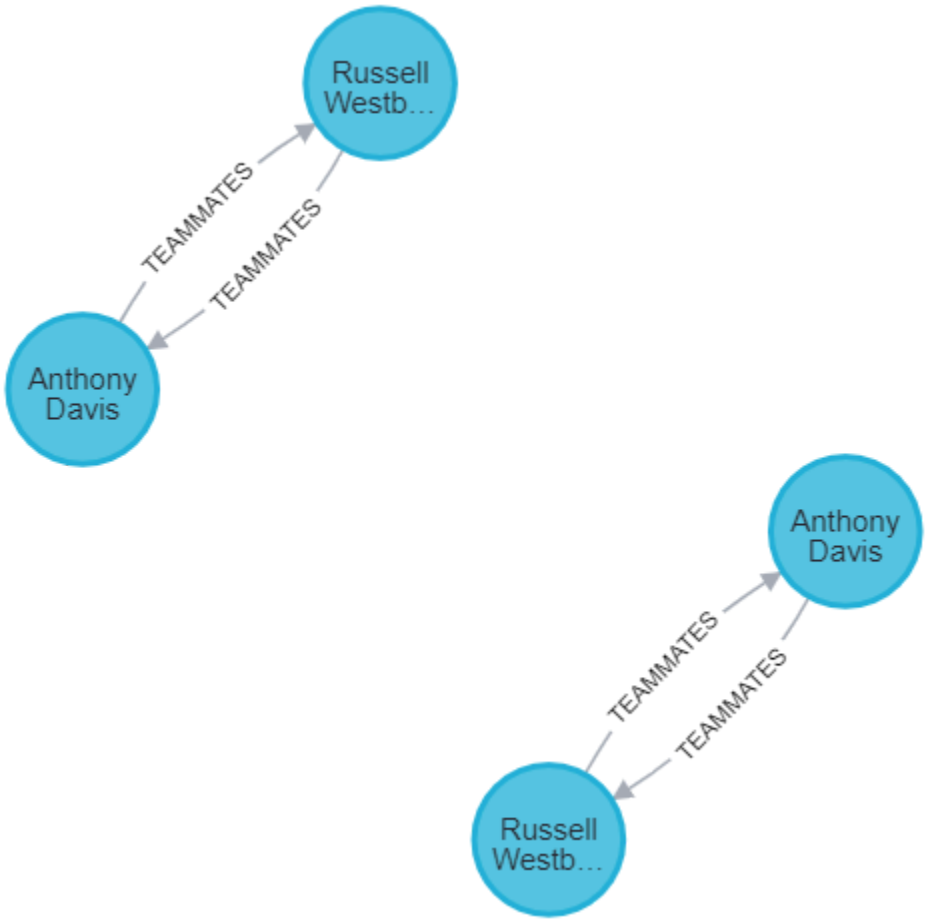
```
MATCH (p:PLAYER) -[:TEAMMATES]-> (p1:PLAYER) RETURN p
```



```
MATCH (p:PLAYER {name : "LeBron James"}) -[:TEAMMATES]-> (p1:PLAYER) RETURN p1
```



```
// GET ALL OF LEBRONS TEAMMATES THAT MAKE MORE THAN 40M //
MATCH (lebron:PLAYER {name: "LeBron James"}) - [:TEAMMATES] -> (teammate:PLAYER)
MATCH (teammate) - [contract:PLAYS_FOR] -> (:TEAM)
WHERE contract.salary >= 4000000
RETURN teammate
```



```
MATCH (p:PLAYER) -[gameplayed:PLAYED_AGAINST]->(t2:TEAM) RETURN
p.name,count(gameplayed)
```

	p.name	count(gameplayed)
1	"Ja Morant"	6
2	"Kevin Durant"	6
3	"James Harden"	6
4	"Joel Embiid"	6
5	"Tobias Harris"	6
6	"Russell Westbrook"	8

```
MATCH (p:PLAYER) -[played_againts:PLAYED_AGAINST]-> (t:TEAM) RETURN p.name ,
SUM(played_againts.points) LIMIT 1
```

	p.name	SUM(played_againts.points)
1	"Ja Morant"	218

```
MATCH (p:PLAYER) -[:PLAYS_FOR]-> (t1:TEAM {name: "LA Lakers"})  MATCH (p)
-[played_againts:PLAYED_AGAINST]-> (t:TEAM) RETURN p.name ,
SUM(played_againts.points) LIMIT 1
```

	p.name	SUM(played_againts.points)
1	"Russell Westbrook"	144

```
MATCH (n:PLAYER {name : "Ja Morant"}) DELETE // bcoz it has relationships to do this use
detach keyword
```

ERROR **Neo.ClientError.Statement.SyntaxError**

```
Invalid input '': expected an expression (line 1, column 45 (offset: 44))
"MATCH (n:PLAYER {name : "Ja Morant"}) DELETE"
                                            ^
```

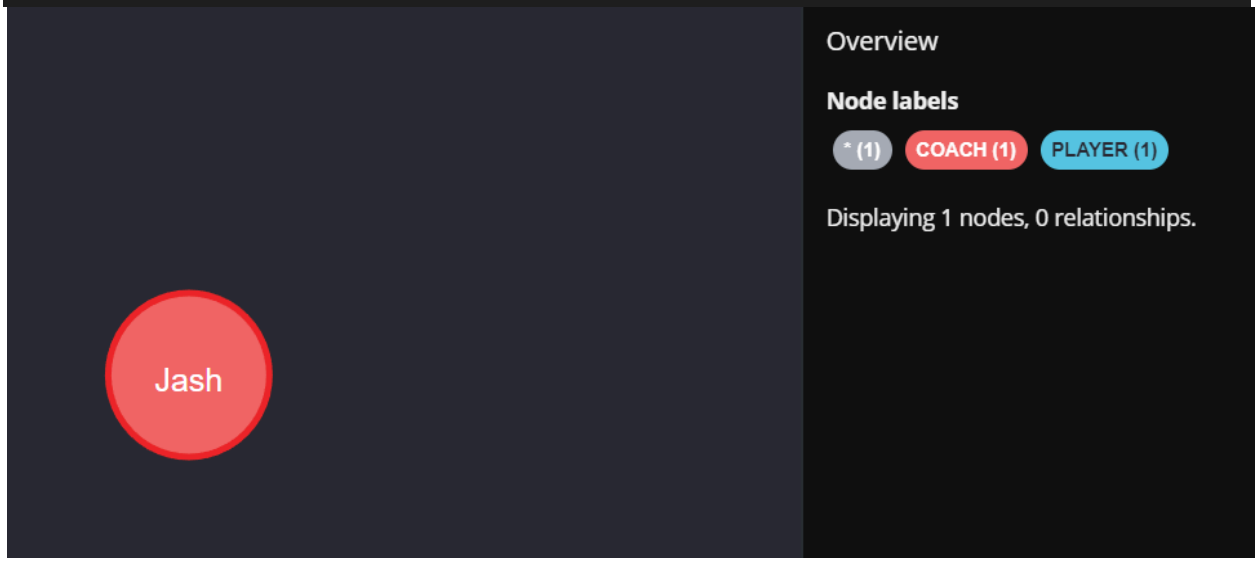
This fails **if the node has any relationships**, because Neo4j enforces relationship integrity.

```
MATCH (n:PLAYER {name : "Ja Morant"}) DETACH DELETE n
```

```
MATCH (n:PLAYER {name : "Joel Embiid"}) -[rel :PLAYS_FOR]-> (:TEAM) DELETE rel
```

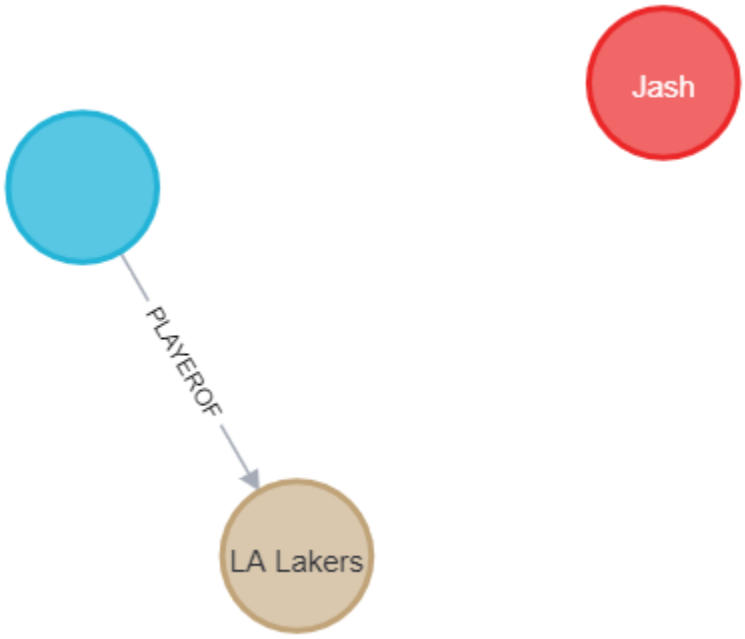
```
MATCH (n) DETACH DELETE n
```

```
CREATE (n:PLAYER:COACH {name : "Jash", height : 23 , weight : 80}) RETURN n
```



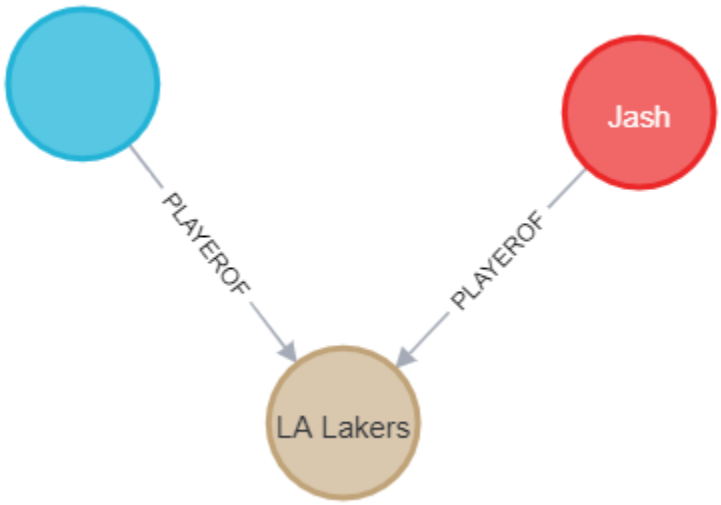

```
CREATE (:PLAYER) -[:PLAYEROF {salary : 50000}]-> (:TEAM {name : "LA Lakers"})
```

```
MATCH (n) RETURN (n)
```

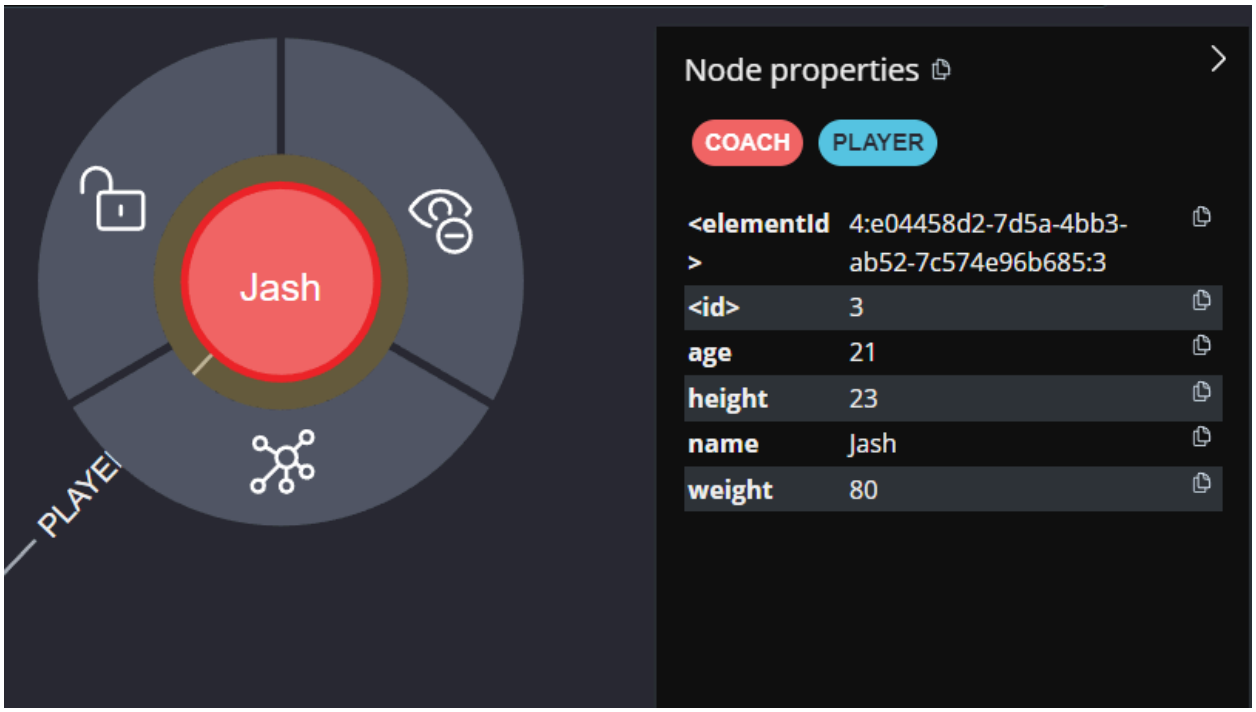



```
MATCH (n:PLAYER {name : "Jash"}), (t:TEAM {name : "LA Lakers"}) CREATE (n)
-[:PLAYEROF]-> (t)
```

```
MATCH (n) RETURN (n)
```



```
MATCH (n:PLAYER {name : "Jash"}) SET n.age = 21
```



you can **add or update properties** of a node **after it's created** using the SET clause.

```
MATCH (p:PLAYER {name: "John"})
SET p.age = 25, p.position = "Midfielder"
RETURN p
```

```
MATCH (p:PLAYER {name: "John"})
SET p += {height: 180, weight: 75}
RETURN p
```