

Phone call logs

Loading CSV file to cypher edit :

To read with headers

```
LOAD CSV WITH HEADERS FROM 'file:///log_of_calls.csv' AS row
RETURN count(row);
```

The screenshot shows the Neo4j browser interface. In the top bar, there is a code editor with the following query:

```
1 LOAD CSV WITH HEADERS FROM 'file:///log_of_calls.csv' AS row
2 RETURN count(row);
```

Below the code editor, the results are displayed in a table:

count(row)
10000

On the left side of the interface, there is a sidebar with three tabs: "Table", "Text", and "Code". The "Table" tab is currently selected. At the bottom of the browser window, a status message reads: "Started streaming 1 records after 34 ms and completed after 242 ms."

Defining and merging nodes along with their relationship

```
LOAD CSV WITH HEADERS FROM 'file:///log_of_calls.csv' AS line
MERGE (c1:City { name: line.from_city })
MERGE (p1:Person { name: line.from_name, number: line.from_number, gender: line.from_gender })
MERGE (p1)-[:FROM]->(c1)
MERGE (c2:City { name: line.to_city })
MERGE (p2:Person { name: line.to_name, number: line.to_number, gender: line.to_gender })
MERGE (p2)-[:FROM]->(c2)
CREATE (c:Call { from: datetime(line.from_dt),
to: datetime(line.to_dt),
duration: duration.between(datetime(line.from_dt), datetime(line.to_dt)).minutes })
CREATE (p1)-[:OUT]->(c)<-[IN]->(p2);
```

The screenshot shows the Neo4j browser interface. In the top bar, there is a command-line prompt with the following text:

```
neo4j$ LOAD CSV WITH HEADERS FROM 'file:///log_of_calls.csv' AS line MERGE (c1:City { n...'
```

Below the command-line, a status message indicates the operation completed successfully:

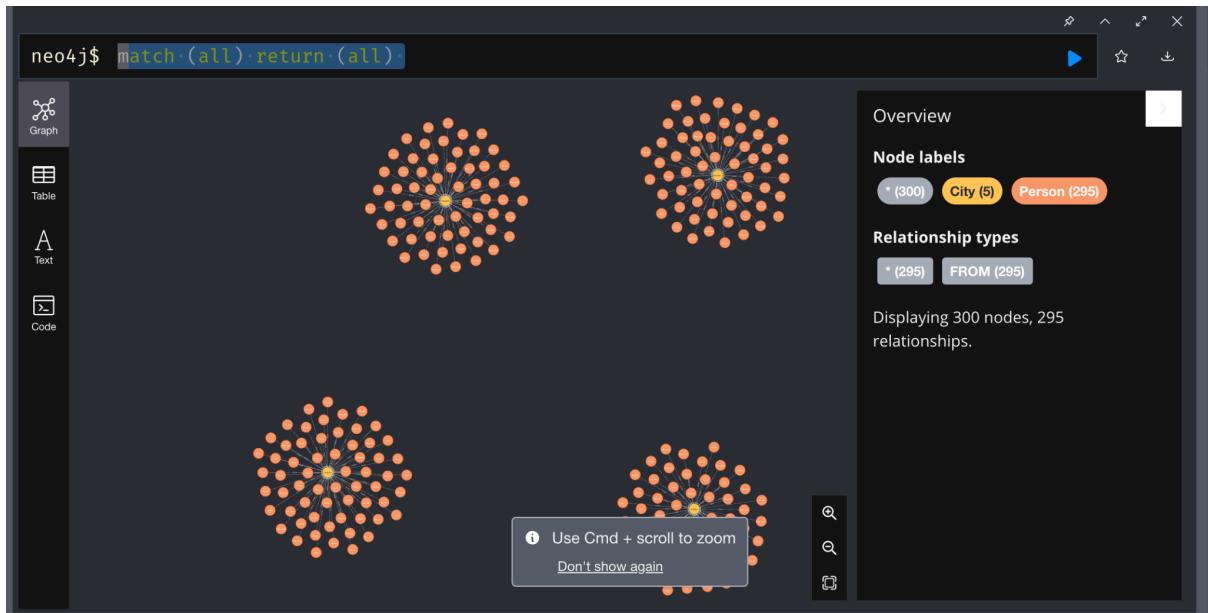
Added 12005 labels, created 12005 nodes, set 36005 properties, created 22000 relationships, completed after 7816 ms.

On the left side of the interface, there is a sidebar with three tabs: "Table", "Warn", and "Code". The "Table" tab is currently selected. At the bottom of the browser window, another status message reiterates the completion details:

Added 12005 labels, created 12005 nodes, set 36005 properties, created 22000 relationships, completed after 7816 ms.

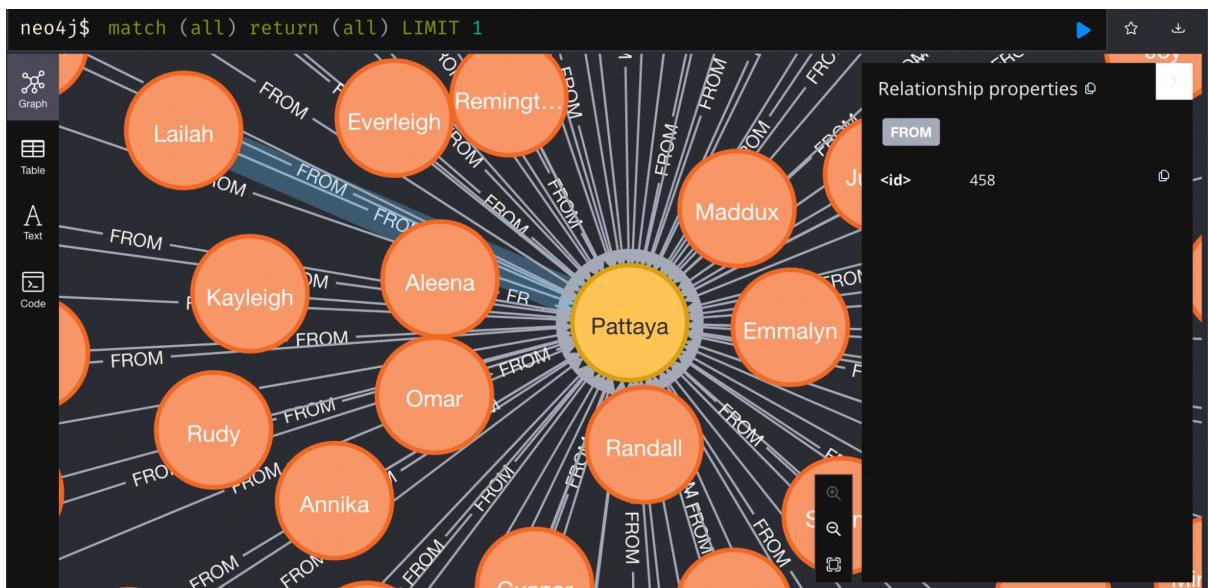
To check the result of complete datasets:

```
match (all) return (all)
```



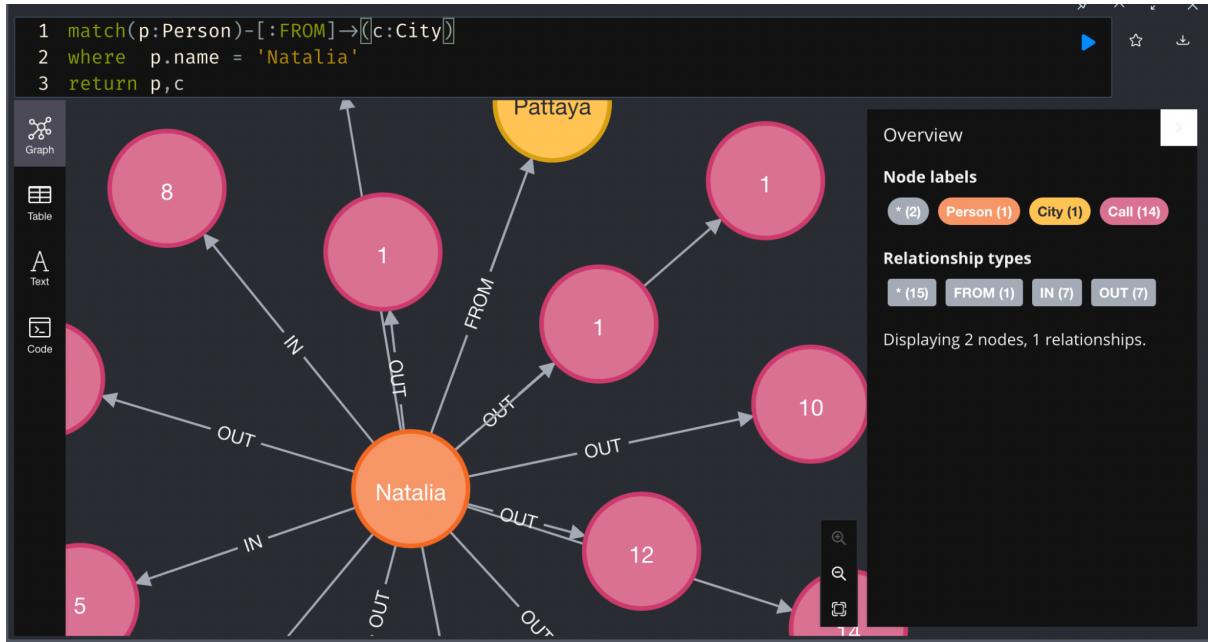
Each parent node represents cities and their child nodes as person:

```
match (all) return (all) LIMIT 1
```



Showing a persons Natalia's calls from Pattaya City details:

```
match(p:Person)-[:FROM]->(c:City)
where p.name = 'Natalia'
return p,c
```



Aggregation:

```
match (p1:Person)- [rel:OUT] ->(c:Call)
where p1.name = "Javier" or p1.name="Robert"
Return p1.name as name, count(c) as total_calls, max(c.duration) as max_duration
```

	name	total_calls	max_duration
1	"Javier"	4	14
2	"Robert"	6	12

Print the name of the person starting with letter “R” and total incoming calls counts made by them arranged in Desc .

```
match (p1:Person) - [rel:IN] -> (c:Call)
where p1.name =~'R.*'
return p1.name, count(c.duration) as Count1
Order by Count1 Desc
limit 5
```

	p1.name	Count1
1	"Rory"	15
2	"Royal"	13
3	"River"	12
4	"Raven"	12
5	"Rowan"	11

Print person name, number whose duration sum of out going call is arranged in descending order

```
match (p1:Person) - [:IN] -> (c:Call) <- [:OUT] - (p2:Person)
return p2.name, p2.number, Sum(c.duration) as Sum1
Order by Sum1 Desc
limit 5
```

	p2.name	p2.number	Sum1
1	"Ruby"	"420-452-5327"	114
2	"Demi"	"226-316-6289"	111
3	"Lina"	"448-273-3382"	100
4	"Clay"	"487-489-6036"	99
5	"Jabari"	"798-158-2190"	97

Print the name total count of calls made in city “KANCHANABURI”

```

MATCH (c:City { name: "Kanchanaburi" })<-[ :FROM ]-(p1:Person)-[:OUT]->(cc:Call)<-[ :IN ]-(p2:Person)
RETURN count(cc) as total_in_out
ORDER BY total_in_out

```

The screenshot shows the Neo4j Browser interface with a query in the top panel:

```

1 MATCH (c:City { name: "Kanchanaburi" })<-[ :FROM ]-(p1:Person)-[:OUT]->(cc:Call)<-[ :IN ]-(p2:Person)
2 RETURN count(cc) as total_in_out
3 ORDER BY total_in_out DESC

```

The results are displayed in a table below:

	total_in_out
1	1865

At the bottom, a status message reads: Started streaming 1 records after 14 ms and completed after 29 ms.

Print person name, number whose duration sum of incoming call is arranged in descending order

```

match (p1:Person) - [:IN] -> (c:Call) <- [:OUT] - (p2:Person)
return p1.name,p1.number, Sum(c.duration) as Sum1
Order by Sum1 Desc
limit 5

```

The screenshot shows the Neo4j Browser interface with a query in the top panel:

```

1 match (p1:Person) - [:IN] -> (c:Call) <- [:OUT] - (p2:Person)
2 return p1.name,p1.number, Sum(c.duration) as Sum1
3 Order by Sum1 Desc
4 limit 5

```

The results are displayed in a table below:

	p1.name	p1.number	Sum1
1	"Ivy"	"299-886-5420"	110
2	"Savannah"	"876-613-2629"	108
3	"Philip"	"735-978-4477"	106
4	"Mercy"	"896-523-1169"	105
5	"Marleigh"	"863-528-9295"	105

At the bottom, a status message reads: Started streaming 5 records after 60 ms and completed after 207 ms.

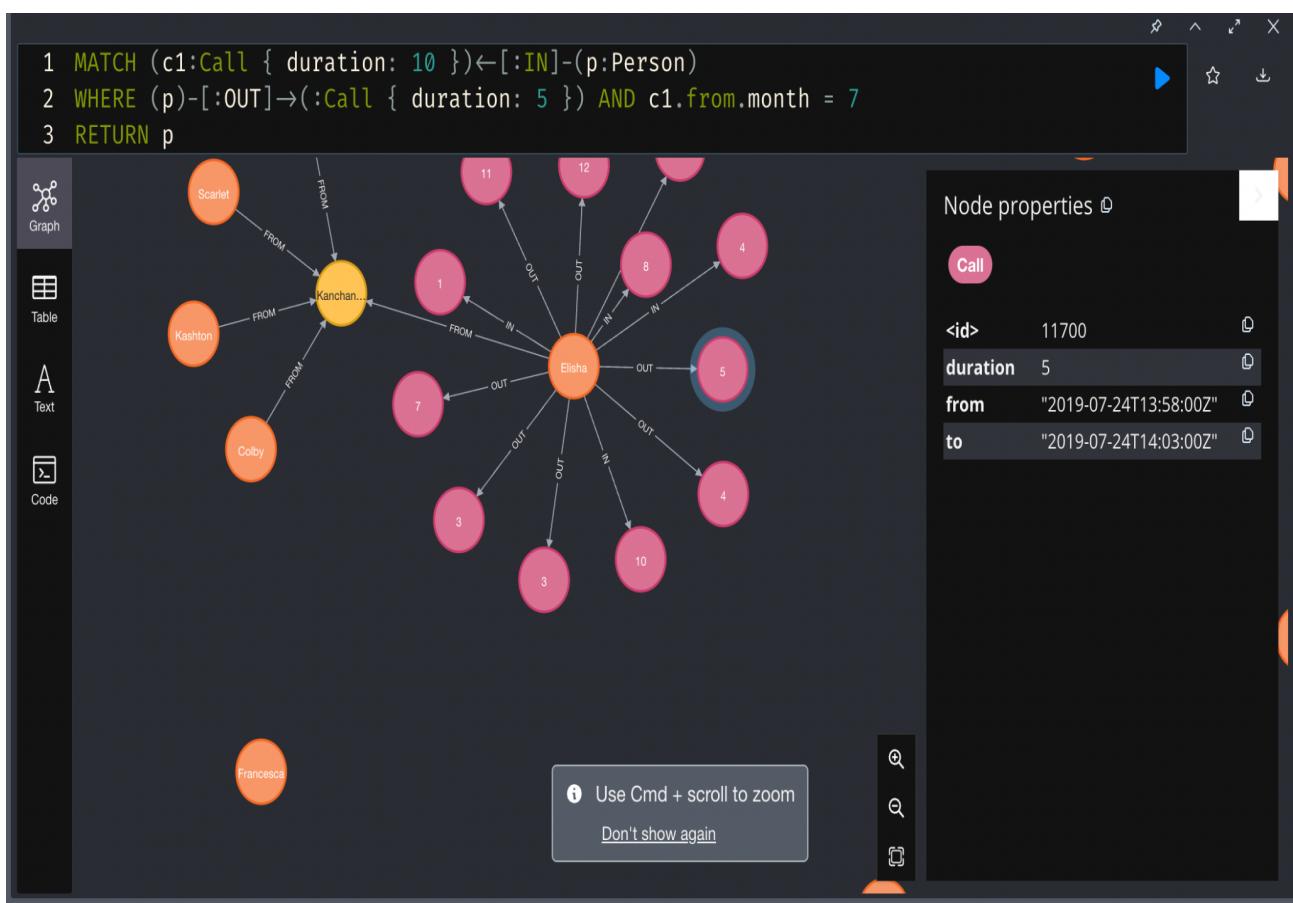
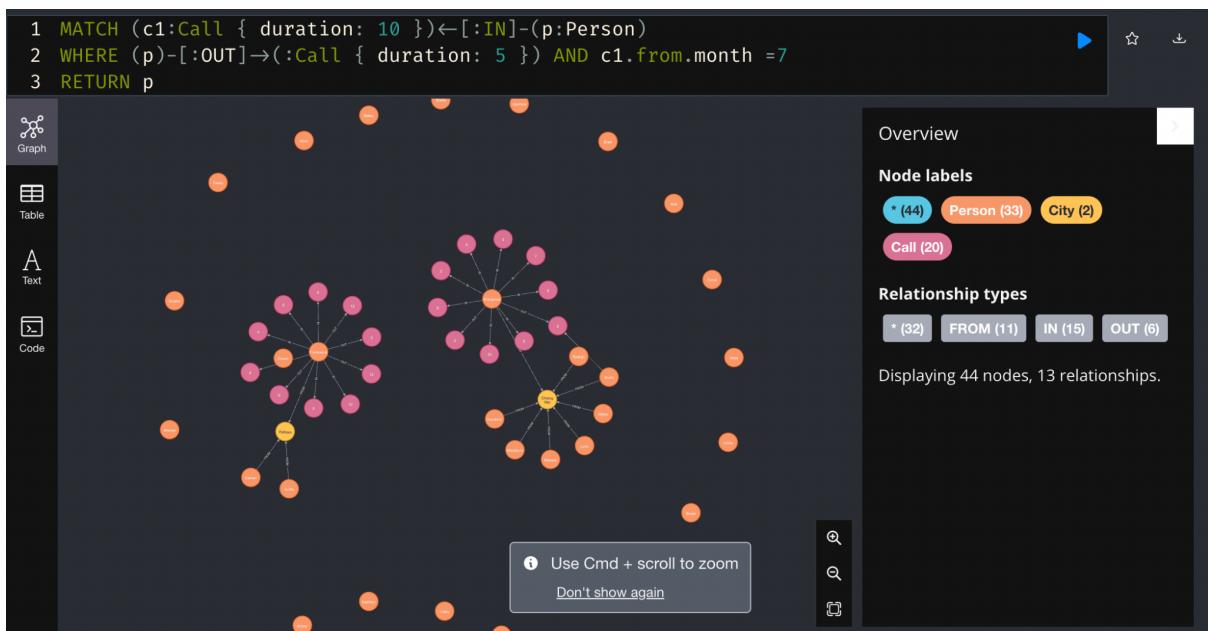
List of Person who made incoming call of duration 10 outgoing duration 5 in month of “July”

```

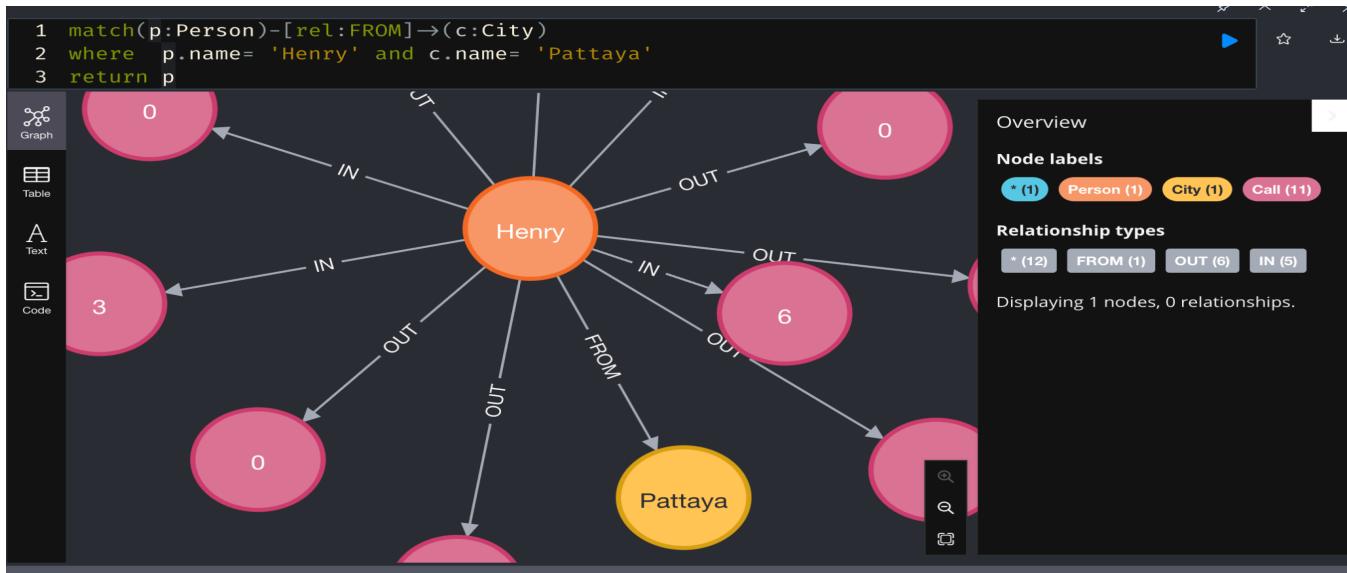
MATCH (c1:Call { duration: 10 })<-[ :IN ]-(p:Person)

```

```
WHERE (p)-[:OUT]->(:Call { duration: 5 }) AND c1.from.month = 8  
RETURN p
```



Data Tranform: Checking node



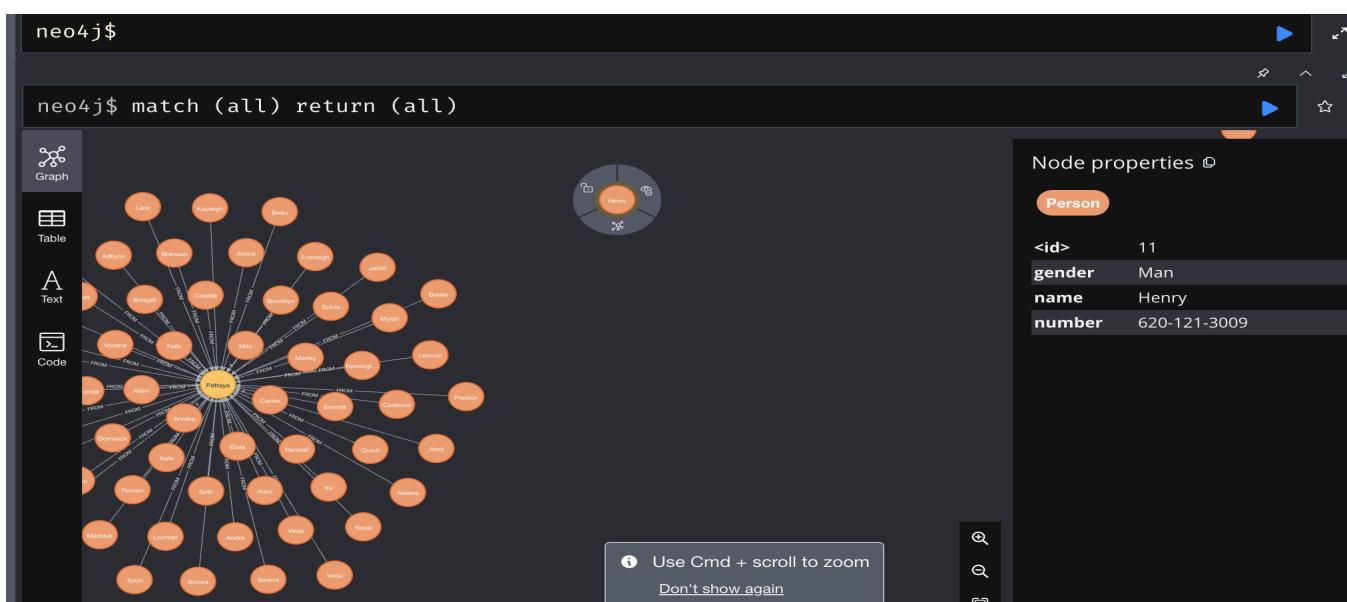
Deleting rel of that person:

```

neo4j$ 

1 match(p:Person)-[rel:FROM]-(c:City)
2 where p.name= 'Henry' and c.name= 'Pattaya'
3 delete rel
  
```

Deleted 1 relationship, completed after 418 ms.



Reference links:

Dataset from Bitbucket data centre. https://vbatush.bitbucket.io/log_of_calls.csv

Studied from: <https://neo4j.com/blog/neo4j-call-detail-records-analytics/>

https://www.youtube.com/watch?v=zNUzkvptw8E&list=PLqfPEK2RTgChcOZ6qHgSfwiBPCz2Bzdjh&ab_channel=SatishCJ