

```
== Movie Graph 2.0
:author: Koji Annoura, Neo4j
:tags: Movie Graph
:neo4j-version: 4.3 modified
```

=== 始めよう: まずは基本の5ステップから

MovieでCypherを学ぼう

\* Current Version

== 0-1: ノード

=== ノード

まずは何もないノードを作成しましょう

```
[source,cypher]
```

```
----
```

```
CREATE ()
```

```
----
```

登録したノードを全て表示します

```
[source,cypher]
```

```
----
```

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

```
----
```

ラベル付きのノード

```
[source,cypher]
```

```
----
```

```
CREATE (n:Movie) RETURN n
```

```
----
```

== 0-2: リレーションシップ

=== MatrixとKeanu Reevesを登録

```
[source,cypher]
```

```
----
```

```
CREATE (TheMatrix:Movie {title:'The Matrix'}),
      (Keanu:Person {name:'Keanu Reeves'})
```

```
CREATE (Keanu)-[:ACTED_IN {roles:['Neo']}]>(TheMatrix)
```

```
RETURN *
```

```
----
```

== 0-3: プロパティの追加

=== Keanu Reevesは1964年生まれ

```
[source,cypher]
```

```
----
```

```
MATCH (Keanu:Person {name:'Keanu Reeves'})
```

```
SET Keanu.born = 1964
```

```
RETURN Keanu
```

```
----
```

```
== 0-4: ラベルの追加
```

```
=== 空のノードにラベルを追加します
```

```
[source,cypher]
```

```
----
```

```
MATCH (n)
WHERE id(n) = 0
SET n:Movie
RETURN n
```

```
----
```

```
== 0-5: データを全て削除
```

```
=== Movieの演習前にデータベースを空にします
```

```
[source,cypher]
```

```
----
```

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

```
----
```

```
== 始めよう: Movieで遊ぼう
```

Movie Graphの登録

Try Neo4j with live dataを開いてMovie Graphを作成します。

```
== 1: Matrixを検索
```

```
=== MovieのタイトルがMatrixを含んでいるものは?
```

下のエリアをクリックしてみましょう。

```
[source,cypher]
```

```
----
```

```
MATCH (n:Movie)
WHERE n.title CONTAINS 'Matrix'
RETURN n;
```

```
----
```

```
=== 1999年にリリースされた映画は?
```

```
[source,cypher]
```

```
----
```

```
MATCH (n:Movie)
WHERE n.released = 1999
RETURN n;
```

```
----
```

```
=== 1999年の映画に出てきた俳優は?
```

```
[source,cypher]
----
MATCH (n:Movie)-[:ACTED_IN]-(m:Person)
WHERE n.released = 1999
RETURN n,m
----
```

== 2: 最短経路  
=== トムハンクスとキアヌリーブスの最短経路

```
[source,cypher]
----
MATCH p=shortestPath(
  (:Person {name:"Tom Hanks"})-[*]-(:Person {name:"Keanu Reeves"}))
RETURN p
----
```

=== トムハンクスとキアヌリーブスの全ての最短経路

```
[source,cypher]
----
MATCH p=allShortestPaths(
  (:Person {name:"Tom Hanks"})-[*]-(:Person {name:"Keanu Reeves"}))
RETURN p
----
```

== 3: パターンマッチ  
=== 映画に出演していない人

```
[source,cypher]
----
MATCH (p:Person)
WHERE NOT (p)--(:Movie)
RETURN p
----
```

=== 出演と監督の両方をしている人

```
[source,cypher]
----
MATCH (p:Person)-[:ACTED_IN]->(m:Movie)<-[:DIRECTED]-(p)
RETURN p
----
```

=== 出演と監督の両方をしている人と映画

```
[source,cypher]
----
MATCH (p:Person)-[:ACTED_IN]->(m:Movie)<-[:DIRECTED]-(p)
RETURN p,m
```

-----

== 4: パターンマッチの応用  
=== 2作品以上で共演もしくは出会った人

```
[source,cypher]
-----
MATCH (p:Person)-->(m1:Movie)<--(p2:Person),
      (p)-->(m2:Movie)<--(p2)
RETURN *
LIMIT 25
-----
```

=== リストで出力

```
[source,cypher]
-----
MATCH (p:Person)-->(m1:Movie)<--(p2:Person),
      (p)-->(m2:Movie)<--(p2)
WHERE p <> p2
RETURN p.name, collect(distinct(p2.name))
LIMIT 25
-----
```

== 5: 俳優になろう  
=== 自分を俳優として登録

```
[source,cypher]
-----
CREATE (:Person {name: {myname}});
-----
```

=== 登録を確認

```
[source,cypher]
-----
MATCH (n:Person {name: {myname}})
RETURN n;
-----
```

== 6: 項目(Property)を登録  
=== 生まれた年は?

```
[source,cypher]
-----
MATCH (n:Person {name: {myname}})
SET n.born = 1970
RETURN n
-----
```

=== 住んでいるところ?

```
[source,cypher]
----
MATCH (n:Person {name: {myname}})
SET n.city = '福岡'
RETURN n
----
```

== 7:映画に出演しよう

```
[source,cypher]
----
MATCH (n:Movie)
WHERE n.title CONTAINS 'Apollo'
RETURN n;
----
```

=== アポロ13に出演

```
[source,cypher]
----
MATCH (n:Movie),(p:Person {name: {myname}})
WHERE n.title CONTAINS 'Apollo'
CREATE (n)-[:ACTED_IN]-(p)
RETURN n,p
----
```

== 8:ラベルの追加

=== 自分にMeと言うラベルを追加

```
[source,cypher]
----
MATCH (n:Person {name: {myname}})
SET n:Me
RETURN n
----
```

== 9:映画に出演しよう

=== マトリクスの作品を探す

```
[source,cypher]
----
MATCH (n:Movie)
WHERE n.title CONTAINS 'Matrix'
RETURN n;
----
```

=== マトリクス3部作に出演

```
[source,cypher]
----
```

```
MATCH (n:Movie),(p:Person {name: {myname}})
WHERE n.title CONTAINS 'Matrix'
CREATE (n)-[:ACTED_IN]-(p)
RETURN n,p
-----
```

== 10:最後のページ: リンク

- \* <http://neo4j.com/developer/cypher>[Learn more about Cypher]
- \* [pass:a\[<a help-topic='keys'>Help Keys</a>\]](#)
- \* [pass:a\[<a play-topic='http://guides.neo4j.com/'>Another Guide</a>\]](#)