

Unfortunately, due to the use of Neo4j, I couldn't have the queries on a Jupyter Notebook. Hence I just used a text editor for you to see them. Sorry for the inconvenience.

```
CREATE (q1:Quest {name: 'The Shadowed Forest', complexity: 'Medium',
rewards: ['1000 gold', 'Enchanted Bow'], player_level_requirement:
10})
CREATE (q2:Quest {name: 'Goblin Invasion', complexity: 'Hard',
rewards: ['2000 gold', 'Diamond Armor'], player_level_requirement:
15})
CREATE (q3:Quest {name: 'Mystic Caverns', complexity: 'Easy',
rewards: ['500 gold', 'Mystic Amulet'], player_level_requirement: 5})
CREATE (q4:Quest {name: 'Lost City of Atlantis', complexity: 'Very
Hard', rewards: ['5000 gold', 'Trident of Poseidon'],
player_level_requirement: 20})
CREATE (q5:Quest {name: 'The Labyrinth of Time', complexity:
'Extreme', rewards: ['10000 gold', 'Chrono Blade'],
player_level_requirement: 30})

RETURN q1, q2, q3, q4, q5
```

```

CREATE (p1:Player {name: 'Lucas', inventory: ['Sword', 'Potion']}),
      (p2:Player {name: 'Emily', inventory: ['Bow', 'Arrow']}),
      (p3:Player {name: 'Sophia', inventory: ['Staff', 'Magic
Book']})),
      (p4:Player {name: 'Daniel', inventory: ['Dagger',
'Lockpick']})),
      (p5:Player {name: 'Isabella', inventory: ['Spear',
'Helmet']})),
      (p6:Player {name: 'Oliver', inventory: ['Mace', 'Shield']}),
      (p7:Player {name: 'Ava', inventory: ['Axe', 'Rope']}),
      (p8:Player {name: 'Mia', inventory: ['Whip', 'Hat']}),
      (p9:Player {name: 'Jackson', inventory: ['Crossbow',
'Bolt']})),
      (p10:Player {name: 'Amelia', inventory: ['Lance', 'Horse
Armor']})),

```

```

(p1)-[:FRIENDS_WITH {date_became_friends: date()}]->(p2),
(p1)-[:FRIENDS_WITH {date_became_friends: date()}]->(p3),
(p2)-[:FRIENDS_WITH {date_became_friends: date()}]->(p4),
(p2)-[:FRIENDS_WITH {date_became_friends: date()}]->(p5),
(p3)-[:FRIENDS_WITH {date_became_friends: date()}]->(p6),
(p3)-[:FRIENDS_WITH {date_became_friends: date()}]->(p7),
(p4)-[:FRIENDS_WITH {date_became_friends: date()}]->(p8),
(p4)-[:FRIENDS_WITH {date_became_friends: date()}]->(p9),
(p5)-[:FRIENDS_WITH {date_became_friends: date()}]->(p10),
(p5)-[:FRIENDS_WITH {date_became_friends: date()}]->(p1),
(p10)-[:FRIENDS_WITH {date_became_friends: date()}]->(p6)

```

```

RETURN p1, p2, p3, p4, p5, p6, p7, p8, p9, p10

```

```
CREATE (npc1:NPC {name: 'Gandalf', dialogue: ['You shall not pass!',  
'A wizard is never late, nor is he early.']}),  
      (npc2:NPC {name: 'Dumbledore', dialogue: ['Happiness can be  
found in the darkest of times, if one only remembers to turn on the  
light.', 'It does not do to dwell on dreams and forget to live.']}),  
      (npc3:NPC {name: 'Yoda', dialogue: ['Do or do not, there is no  
try.', 'Fear is the path to the dark side.']}),  
      (npc4:NPC {name: 'Voldemort', dialogue: ['There is no good and  
evil, there is only power, and those too weak to seek it.', 'I am  
Lord Voldemort.']}),  
      (npc5:NPC {name: 'Saruman', dialogue: ['Against the power of  
Mordor, there is no victory.', 'We must join with Sauron.']}))  
  
RETURN npc1, npc2, npc3, npc4, npc5
```

Queries asked to fulfill:

```
MATCH (p:Player), (q:Quest)
WHERE id(p) % 5 = id(q) % 5
CREATE (p)-[:PARTICIPATED_IN {
  items_collected: q.rewards,
  points_scored: 1000,
  time_taken: 120
}]->(q)
```

```
MATCH path = shortestPath((p1:Player {name:
'Player1'})-[:FRIENDS_WITH*]-(p2:Player {name: 'Player2'}))
WITH path, relationships(path) as rels
UNWIND rels as rel
MATCH (rel)-[:PARTICIPATED_IN]->(q:Quest)
RETURN path, collect(distinct q) as sharedQuests
```

```
MATCH (p:Player)-[:PARTICIPATED_IN]->(q:Quest {name: 'Goblin
Invasion'})
MATCH (p)-[:ENCOUNTERED]->(npc:NPC)
RETURN npc.name as NPC, npc.dialogue as DialogueOptions
```

```
MATCH (p:Player)-[:DEFEATED]->(e:Enemy {name: 'Goblin'})
RETURN p.name as PlayerName
```

```
MATCH (p1:Player)-[:FRIENDS_WITH]-(p2:Player)
WHERE NOT
exists((p1)-[:PARTICIPATED_IN]->()-[:PARTICIPATED_IN]-(p2))
RETURN p1.name as Player1, p2.name as Player2
```

```
MATCH (p:Player {name: 'PlayerName'})-[:FRIENDS_WITH*0..1]-(friend)
MATCH (friend)-[:PARTICIPATED_IN]->(q:Quest)
RETURN q.name as QuestName
```

```

MATCH (p1:Player {name:
'Player1'})-[:FRIENDS_WITH]-(mf:Player)-[:FRIENDS_WITH]-(p2:Player
{name: 'Player2'})
WHERE exists((mf)-[:PARTICIPATED_IN]->()<-[:PARTICIPATED_IN]-(p1))
AND exists((mf)-[:PARTICIPATED_IN]->()<-[:PARTICIPATED_IN]-(p2))
    OR exists((mf)-[:DEFEATED]->()<-[:DEFEATED]-(p1)) AND
exists((mf)-[:DEFEATED]->()<-[:DEFEATED]-(p2))
RETURN mf.name as MutualFriend

```

```

MATCH (p:Player)-[:ENCOUNTERED]->(npc:NPC)
RETURN npc.name as NPC, count(p) as NumberOfPlayers
ORDER BY NumberOfPlayers DESC
LIMIT 1

```

```

MATCH (q:Quest)-[:PARTICIPATED_IN]-(p:Player)
WHERE 'ItemName' IN q.rewards
RETURN q.name as QuestName, collect(p.name) as PlayersCollectedItem

```