// Create User nodes

LOAD CSV WITH HEADERS FROM 'file:///users.csv' AS row

CREATE (:User {user\_id: toInteger(row.user\_id), name: row.name, age: toInteger(row.age), city: row.city})

// Create Friendship relationships

LOAD CSV WITH HEADERS FROM 'file:///friendships.csv' AS row

MATCH (u1:User {user\_id: toInteger(row.user\_id)})

MATCH (u2:User {user\_id: toInteger(row.friend\_id)})

CREATE (u1)-[:FRIEND]->(u2)

// Create Post nodes

LOAD CSV WITH HEADERS FROM 'file:///posts.csv' AS row

CREATE (:Post {post\_id: toInteger(row.post\_id), content: row.content})

WITH row

// Connect User nodes to their respective Post nodes

MATCH (u:User {user\_id: toInteger(row.user\_id)})

MATCH (p:Post {post\_id: toInteger(row.post\_id)})

CREATE (u)-[:POSTED]->(p)

// Create Comment nodes

LOAD CSV WITH HEADERS FROM 'file:///comments.csv' AS row

CREATE (:Comment {comment\_id: toInteger(row.comment\_id), content: row.content})

WITH row

// Connect User nodes and Post nodes to their respective Comment nodes

MATCH (u:User {user\_id: toInteger(row.user\_id)})

MATCH (p:Post {post\_id: toInteger(row.post\_id)})

MATCH (c:Comment {comment\_id: toInteger(row.comment\_id)})

CREATE (u)-[:COMMENTED]->(c)

CREATE (c)-[:ON]->(p)

// Create Like relationships

LOAD CSV WITH HEADERS FROM 'file:///likes.csv' AS row

MATCH (u:User {user\_id: toInteger(row.user\_id)})

MATCH (p:Post {post\_id: toInteger(row.post\_id)})

CREATE (u)-[:LIKED]->(p)