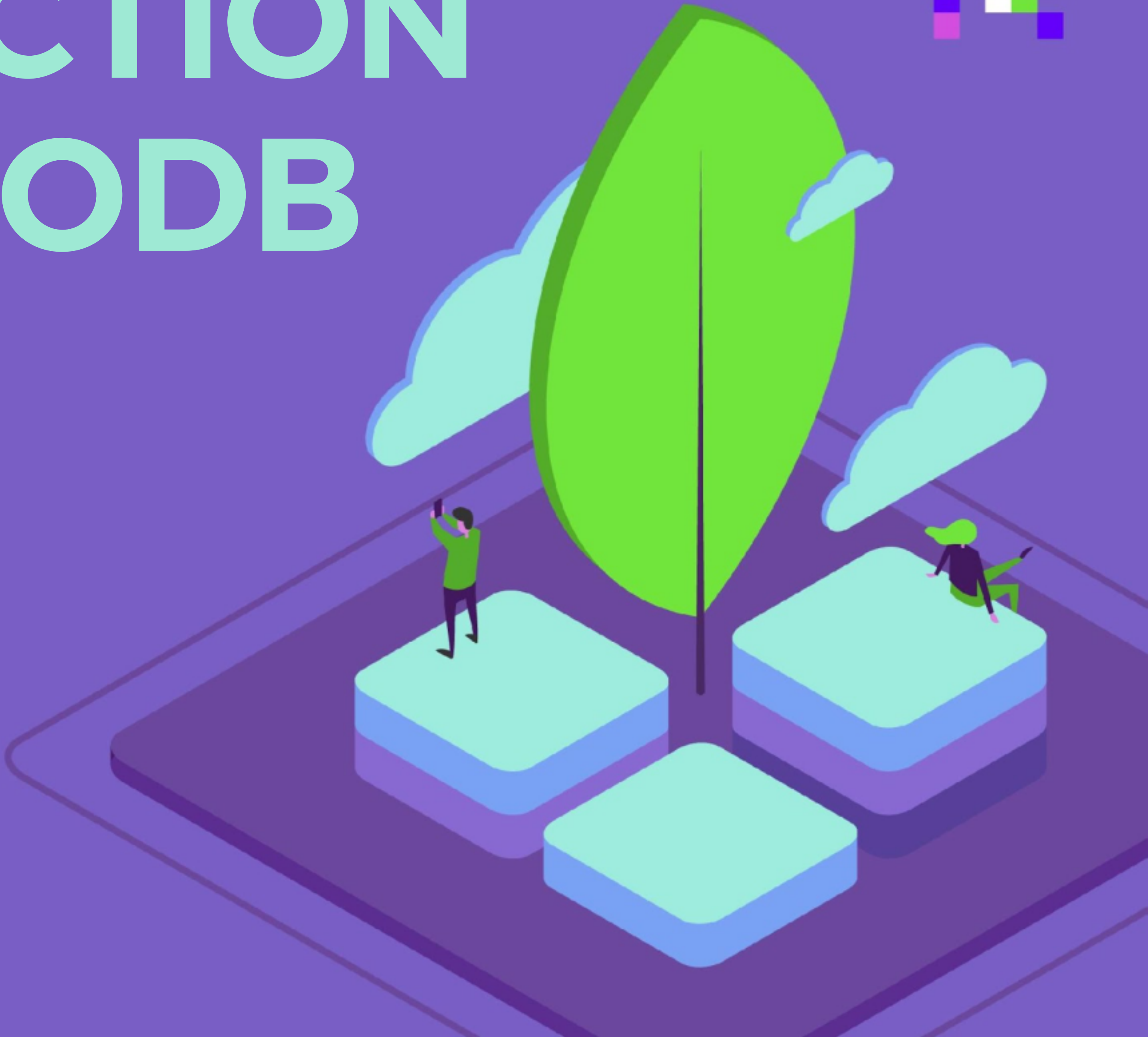
INTRODUCTION TO MONGODB



Intro to Databases / NOSQL INTUITION

NoSQL vs. SQL

SQL	NoSQL

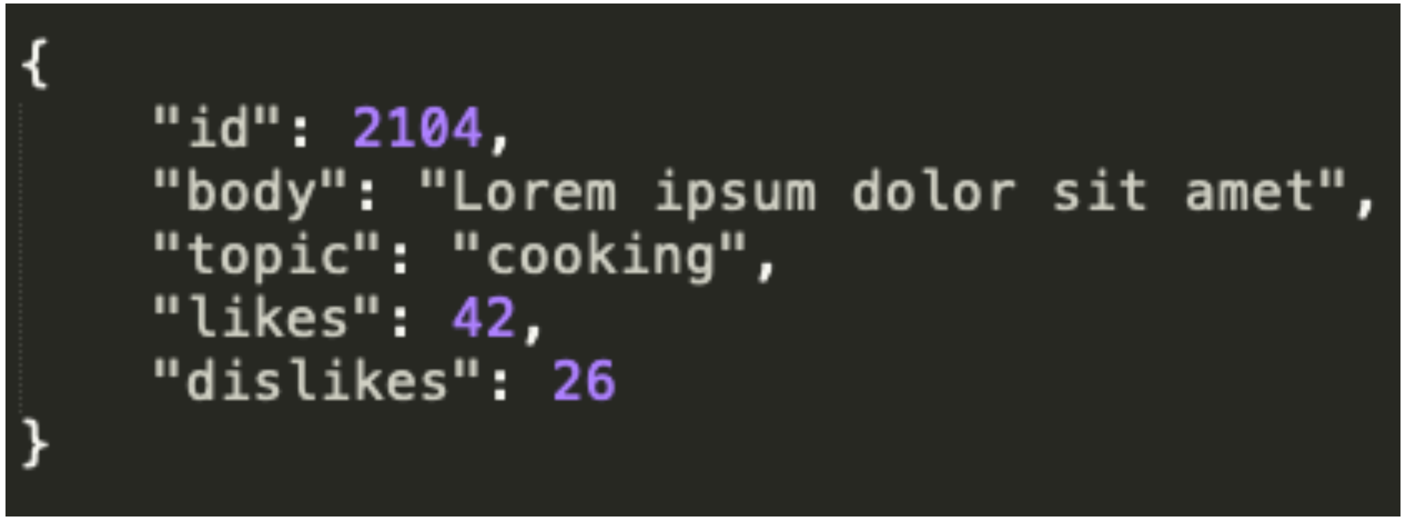
NoSQL vs. SQL

SQL	NoSQL
Table Based	**Documents**, Key-Value pairs, Graph-based, or Wide Column Stores

NoSQL vs. SQL

Example: Blog Post

id	body	topic	likes	dislikes
2104	Lorem ipsum dolor sit amet	cooking	42	26



```
{
    "id": 2104,
    "body": "Lorem ipsum dolor sit amet",
    "topic": "cooking",
    "likes": 42,
    "dislikes": 26
}
```

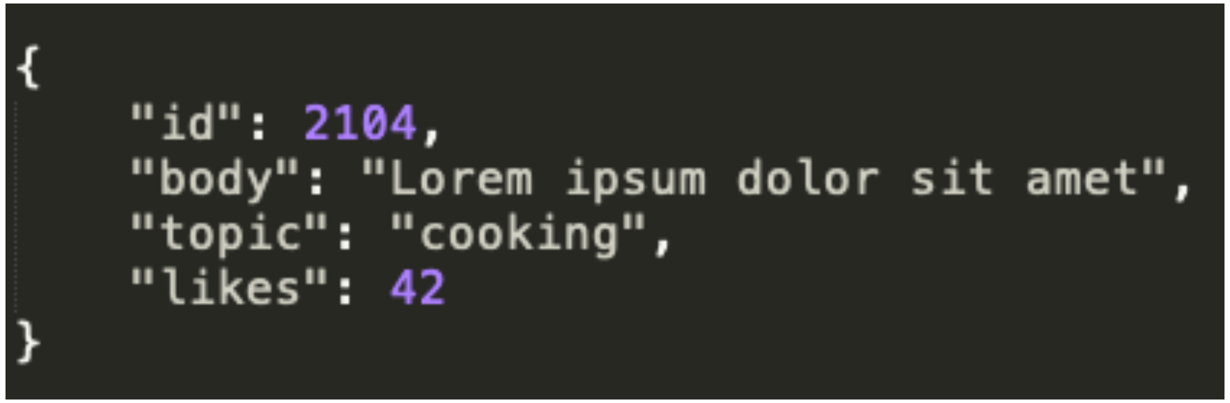
NoSQL vs. SQL

SQL	NoSQL
Table Based	**Documents**, Key-Value pairs, Graph-based, or Wide Column Stores
Defined Schema	Undefined / Flexible Schema

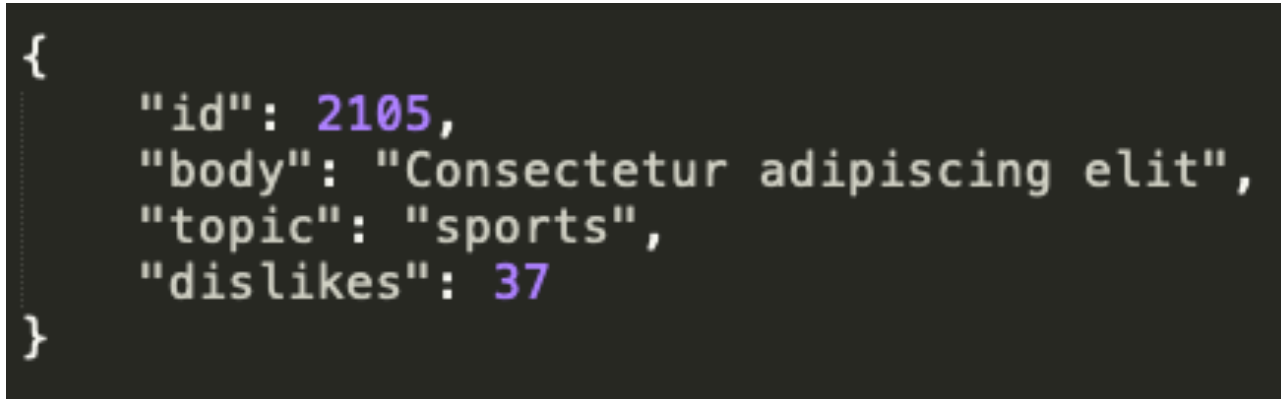
NoSQL vs. SQL

Example: Blog Post

id	body	topic	likes	dislikes
2104	Lorem ipsum dolor sit amet	cooking	42	0
2105	Consectetur adipiscing elit	sports	0	37



```
{
    "id": 2104,
    "body": "Lorem ipsum dolor sit amet",
    "topic": "cooking",
    "likes": 42
}
```



```
{
    "id": 2105,
    "body": "Consectetur adipiscing elit",
    "topic": "sports",
    "dislikes": 37
}
```

NoSQL vs. SQL

SQL	NoSQL
Table Based	**Documents**, Key-Value pairs, Graph-based, or Wide Column Stores
Defined Schema	Undefined / Flexible Schema
Better for Complex Queries	Better for Complex Data Structures

NoSQL vs. SQL

Example: Blog posts with comments (SQL)

Table 1: Posts

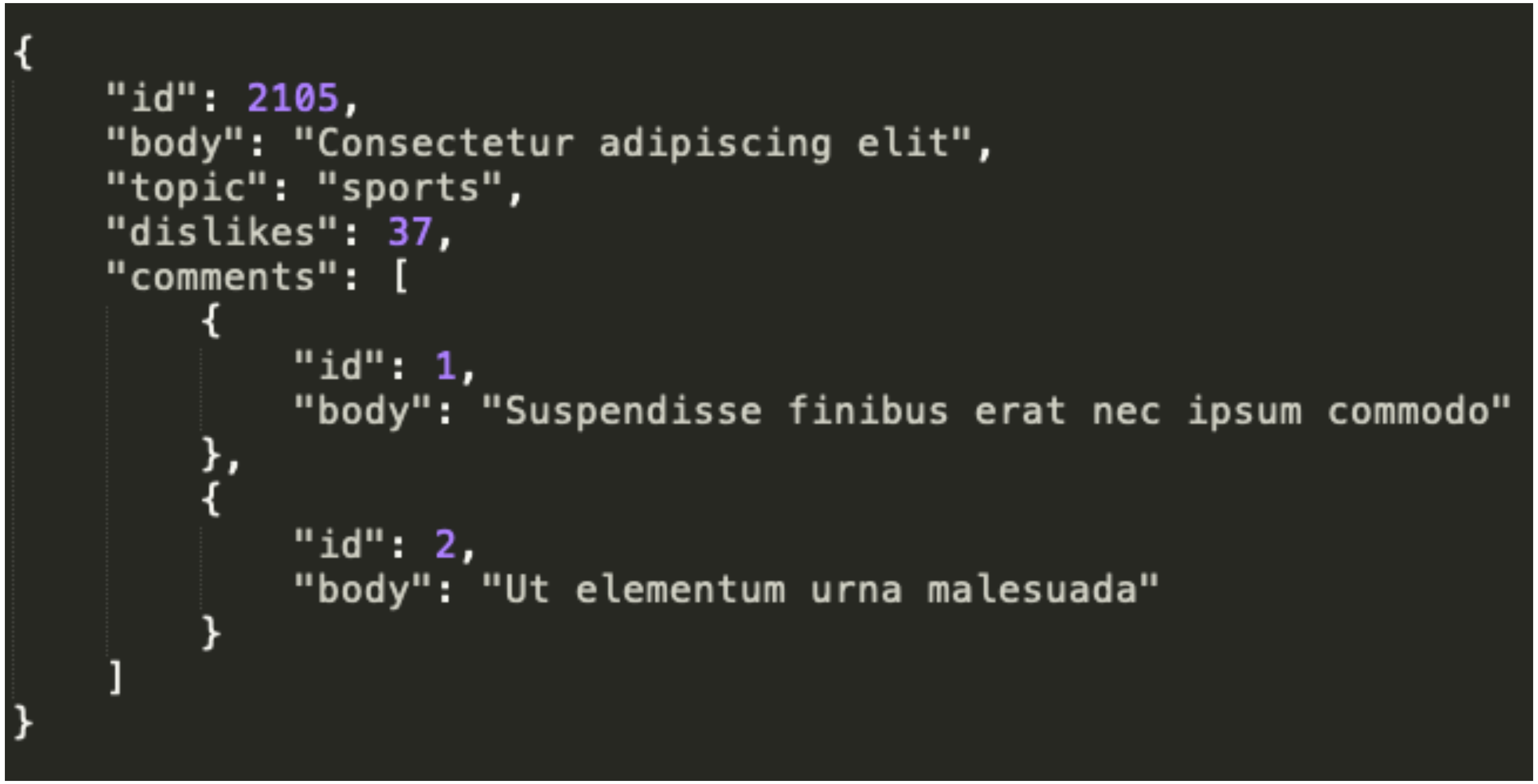
id	body	topic	likes	dislikes
2104	Lorem ipsum dolor sit amet	cooking	42	0
2105	Consectetur adipiscing elit	sports	0	37

Table 2: Comments

id	post_id	body
1	2105	Suspendisse finibus erat nec ipsum commodo
2	2105	Ut elementum urna malesuada

NoSQL vs. SQL

Example: Blog post with comments (NoSQL)



```
{
    "id": 2105,
    "body": "Consectetur adipiscing elit",
    "topic": "sports",
    "dislikes": 37,
    "comments": [
        {
            "id": 1,
            "body": "Suspendisse finibus erat nec ipsum commodo"
        },
        {
            "id": 2,
            "body": "Ut elementum urna malesuada"
        }
    ]
}
```

NoSQL vs. SQL

SQL	NoSQL
Table Based	**Documents**, Key-Value pairs, Graph-based, or Wide Column Stores
Defined Schema	Undefined / Flexible Schema
Better for Complex Queries	Better for Complex Data Structures
Better for Transactional Systems	Better for Horizontal Scaling

NoSQL vs. SQL

Example: Accounting System (SQL)

Table 1: Accounts

id	first_name	account_balance
104	Robert	35105.32
105	Marie	48206.53

Table 2: Transactions

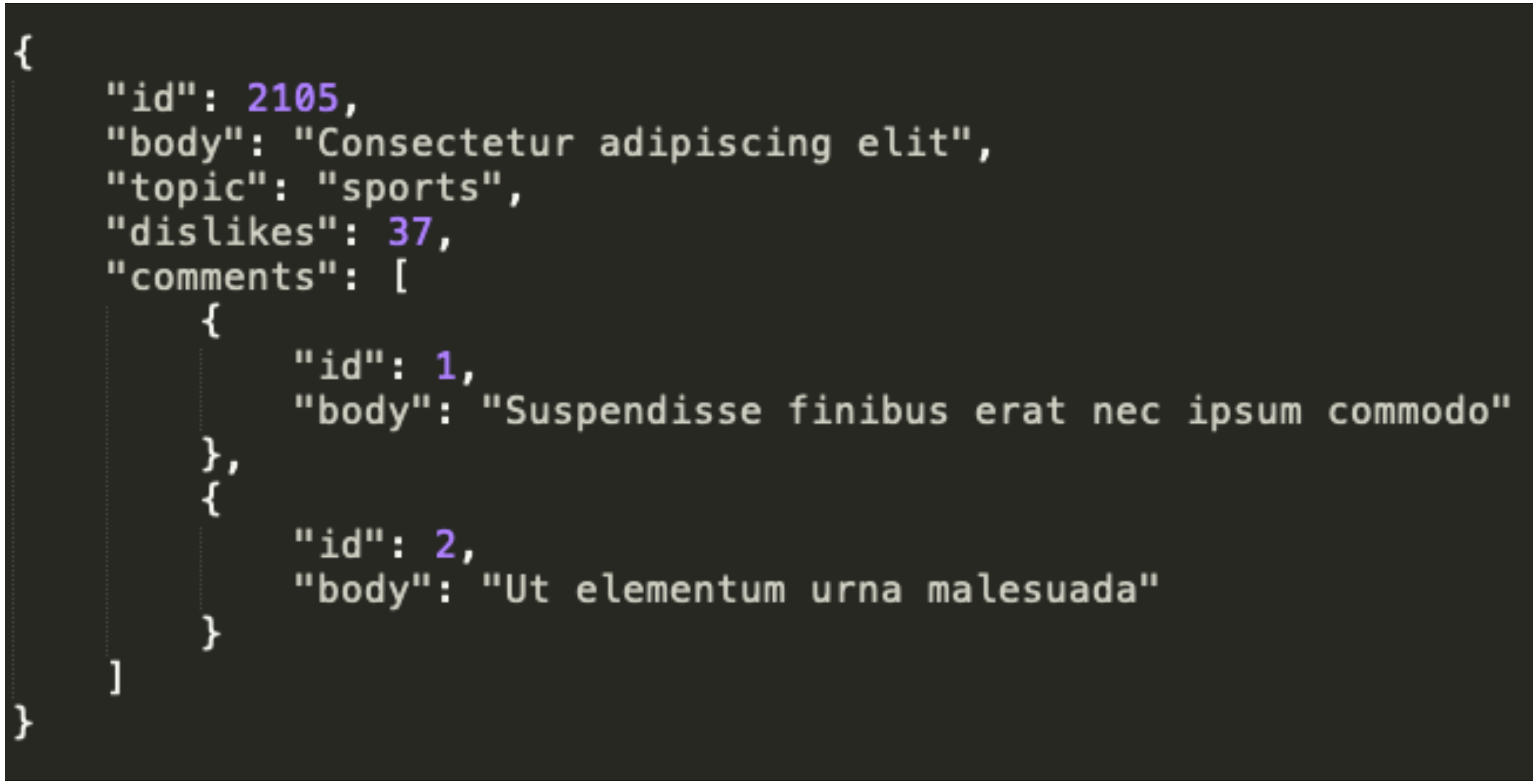
id	sender_id	receiver_id	amount
1	105	104	2000.00

NoSQL vs. SQL

SQL	NoSQL
Table Based	**Documents**, Key-Value pairs, Graph-based, or Wide Column Stores
Defined Schema	Undefined / Flexible Schema
Better for Complex Queries	Better for Complex Data Structures
Better for Transactional Systems	Better for Horizontal Scaling

NoSQL vs. SQL

Example: Blog post (NoSQL)



```
{
    "id": 2105,
    "body": "Consectetur adipiscing elit",
    "topic": "sports",
    "dislikes": 37,
    "comments": [
        {
            "id": 1,
            "body": "Suspendisse finibus erat nec ipsum commodo"
        },
        {
            "id": 2,
            "body": "Ut elementum urna malesuada"
        }
    ]
}
```

NoSQL vs. SQL

Example: Blog posts with comments (SQL)

Table 1: Posts

id	body	topic	likes	dislikes
2104	Lorem ipsum dolor sit amet	cooking	42	0
2105	Consectetur adipiscing elit	sports	0	37

Table 2: Comments

id	post_id	body
1	2105	Suspendisse finibus erat nec ipsum commodo
2	2105	Ut elementum urna malesuada

NoSQL vs. SQL

SQL	NoSQL
Table Based	**Documents**, Key-Value pairs, Graph-based, or Wide Column Stores
Defined Schema	Undefined / Flexible Schema
Better for Complex Queries	Better for Complex Data Structures
Better for Transactional Systems	Better for Horizontal Scaling
Examples: MySQL, Postgres, Oracle, SQLite	Examples: MongoDB, Cassandra, HBase, Redis, Neo4j

CRUD Operations

Create **R**ead **U**pdate **D**elete

CRUD Operations

- Create

```
db.collection.insert({“name”: “patrick”})
```

- Read

```
db.collection.find({“age”: 42})
```

- Update

```
db.collection.update({“country”: “US”}, {“country”: “USA”})
```

- Delete

```
db.collection.remove({“user_id”: 4106})
```