



# package.json v/s package-lock.json

In Detail



Gagan Saini

# package.json

**Metadata:** It's is a JSON file that contains metadata about the Node.js project, such as the project's name, version, description, and dependencies.

**Project Configuration:** It also houses various project-specific configurations, including scripts, author information, license, and more.



# package.json

**Dependency Listing:** The file lists all the direct dependencies required for the project to run correctly but does not specify their exact versions.





# package-lock.json

**Dependency Locking:** It's, introduced in npm v5 and later, is automatically generated to lock the exact versions of the dependencies installed.

**Version Consistency:** It ensures that all developers working on the project and deployment environments use the same dependency versions, thus preventing version conflicts.



# package-lock.json

**Faster Installation:** It enables faster and more deterministic dependency installation, promoting a consistent development environment.



# Relationship

Both files work hand-in-hand to manage dependencies effectively. When installing dependencies with npm, package-lock.json is automatically generated or updated based on the versions resolved in package.json.





# Caret ( ^ )

When a version number has a caret symbol (^) prefix, it indicates that the dependency can be updated to the latest **minor or patch** version within the **same major** version.

example: **^1.2.3** means any version greater than or equal to **1.2.3** but less than **2.0.0**.

By using this it allows you for flexibility when installing updates. It ensures that your project gets compatible updates with bug fixes and minor improvements but **avoids major version updates** that might introduce breaking changes.



# Tilda ( ~ )

When a version number has a tilde symbol (~) prefix, it indicates that the dependency can be updated to the **latest patch version** within the **same minor** version.

example: **~1.2.3** means any version greater than or equal to **1.2.3** but less than **1.3.0**.

Using this allows for more conservative updates, as it only includes **patches** with bug fixes and backward-compatible changes. Major or minor updates that could introduce breaking changes are avoided.





# package.json

## Example file



```
{  
  "name": "Pizza_Party_Planning",  
  "version": "1.0.0",  
  "description": "Plan for the pizza party",  
  "dependencies": {  
    "pepperoni": "^1.0.0",  
    "margherita": "^2.0.0",  
    "vegetarian": "^1.5.0"  
  }  
}
```



# package-lock.json

## Example file

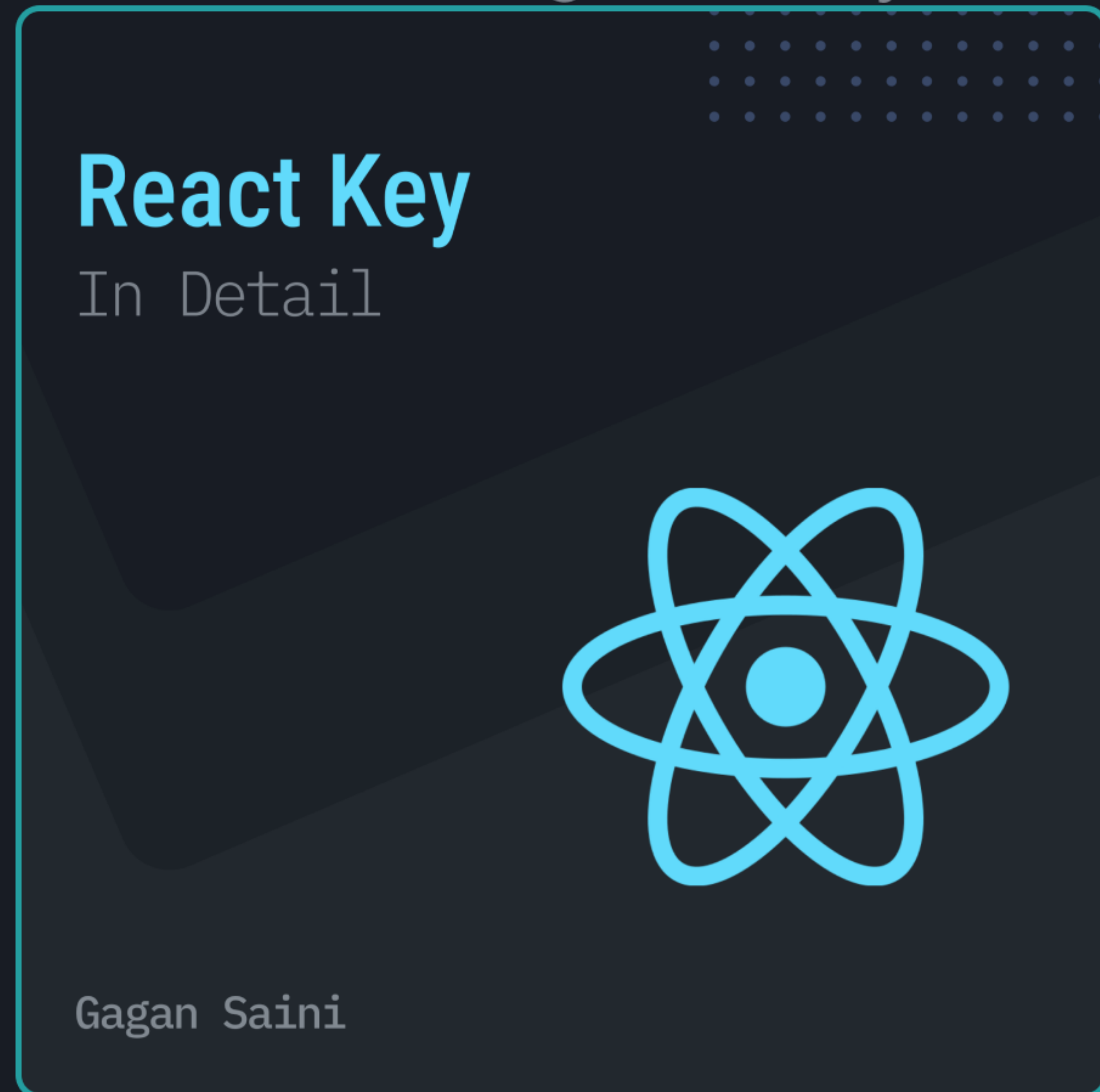


```
{
  "name": "Pizza_Party_Planning",
  "lockfileVersion": 2,
  "requires": true,
  "dependencies": {
    "pepperoni": {
      "version": "1.0.0",
      "integrity": "sha256-Pepperoni_Checksum"
    },
    "margherita": {
      "version": "2.0.0",
      "integrity": "sha256-Margherita_Checksum"
    },
    "vegetarian": {
      "version": "1.5.0",
      "integrity": "sha256-Vegetarian_Checksum"
    }
  }
}
```

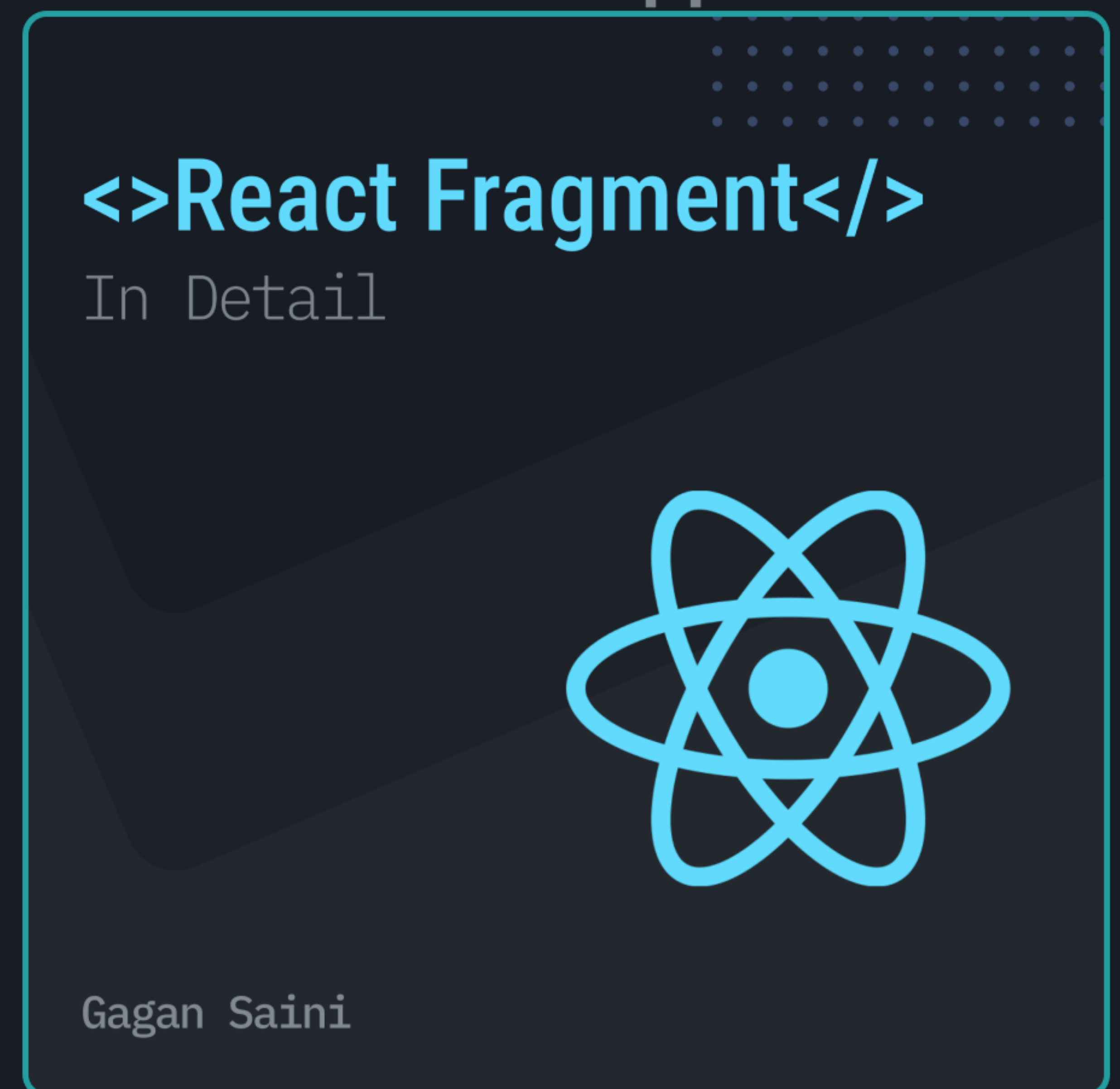


# Checkout others posts!

## React rendering with Keys



## The Invisible Wrapper







@Gagan Saini

---

Web Developer

Connect For More !

