School of Computer Science & IT JAIN (DEEMED TO BE UNIVERSITY) Knowledge Campus, Bangalore

Department of Master of Computer Applications – Academic Year 2020 – 2021

**Department of Master of Computer Science**

**C# and .net Technologies**

# MySQL vs MS SQL and Angular vs React

**Activity-2**

**Submitted By:**

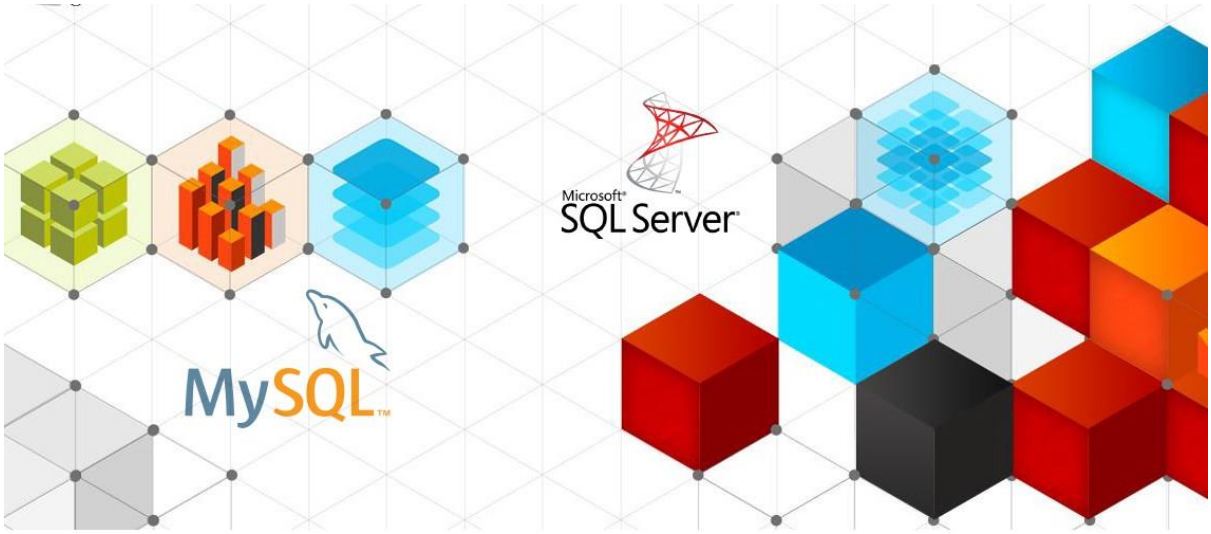
Yerraballi Suresh Kumar Reddy

19MCAR0080

Semester: 3rd MCA

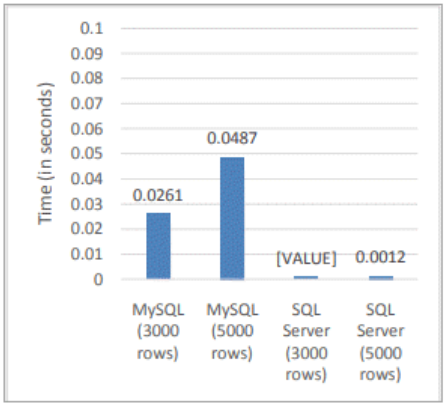
**Submitted To:**

Prof. Kathiresan V

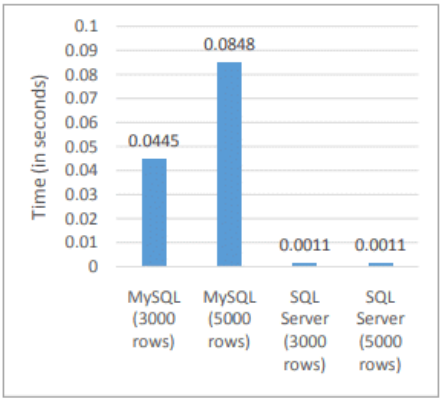



| Factors | MS SQL Server | MySQL |
|---|---|---|
| Length function | SELECT LEN(data_string) FROM TableName | SELECT CHARACTER_LENGTH(data_string) FROM TableName |
| Concatenation function | SELECT ('SQL' + 'SERVER') | SELECT CONCAT ('My', 'SQL') |
| Select top n records from a table | SELECT TOP 10 * FROM TableName WHERE id = 2 | SELECT * FROM TableName WHERE id = 2 LIMIT 10 |
| Generate GUID (Global Unique Identifier) | SELECT NEWID() | SELECT UUID() |
| Get current date and time | SELECT GETDATE() | SELECT NOW() |
| Case Sensitive Collation | In SQL Server, if the database is defined with case sensitive collation then the table names and column names become case sensitive. | In MySQL, there is no case sensitiveness in identifier names. |

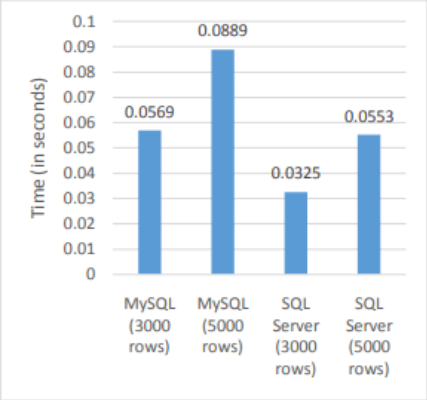
**Averages for a non-conditional SELECT query**



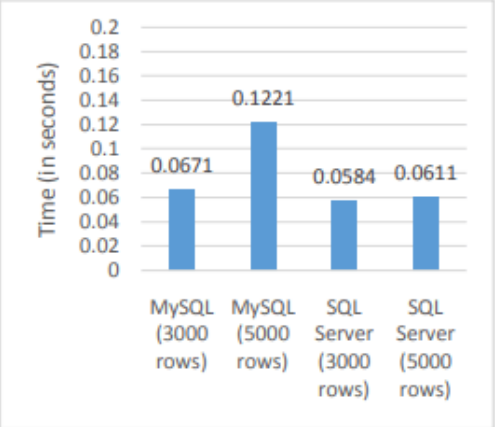

**Averages for SELECT query with a JOIN**



### Averages for SELECT query having an ORDER clause on a non-indexed field

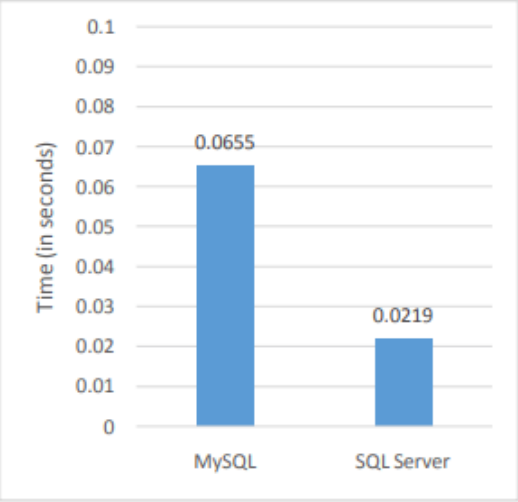



### Averages for SELECT query having a JOIN and an ORDER clause on a non-indexed field

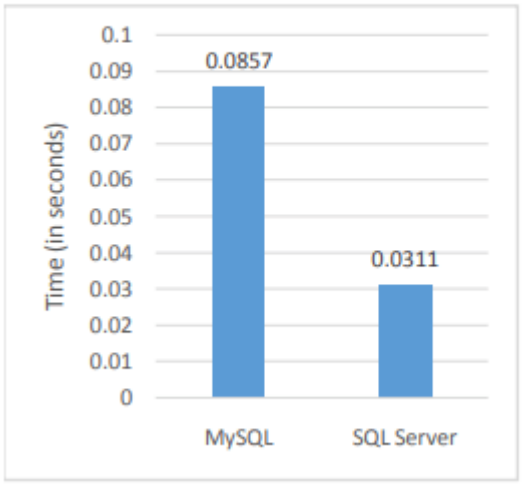



### Averages for 100 INSERT queries

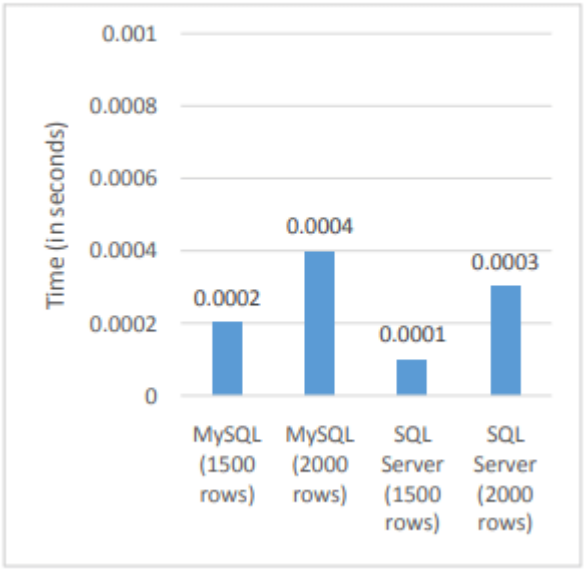

### Averages for conditional DELETE query



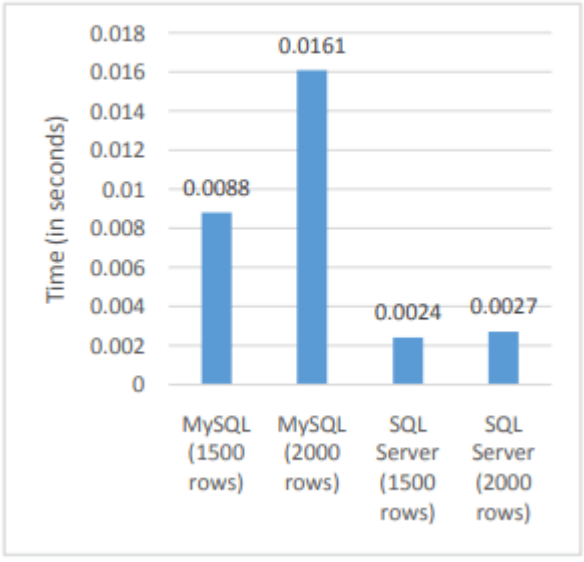
## Averages for non-conditional DELETE query




## Averages for conditional UPDATE query




## Averages for non-conditional UPDATE query



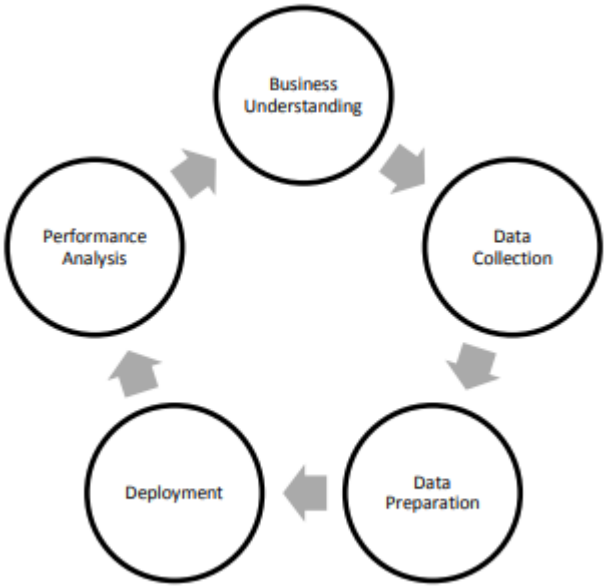


Fig 1: The test method

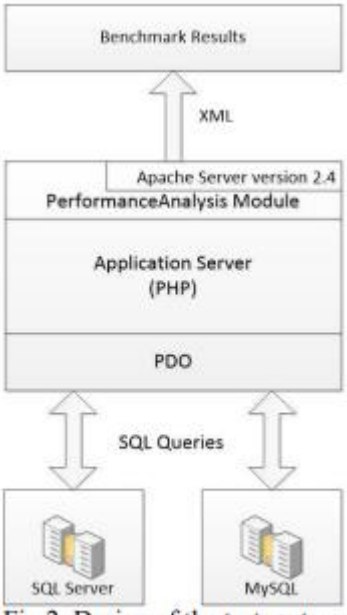


Fig 2: Design of the test system

While developing software applications, programmers use relational database management system (RDBMS) to create, read, update and delete back-end data. They even manipulate the RDBMS through custom structured query language (SQL) statements. The developers have option to choose from several RDBMS according to specific requirements of each project.

But the choice of database differs from one programmer to another. Many enterprises prefer open source database systems to commercial database systems to save money. But many large enterprises opt for commercial RDBMS to avail a number of advanced features along with latest security mechanism and encryption technology.

Both MySQL and MS SQL Server are widely used enterprise database systems. MySQL is an open source RDBMS, whereas SQL Server is a Microsoft product. Microsoft allows enterprises to choose from several editions of SQL Server according to their needs and budget. But the smart programmers always keep in mind the major differences between MySQL and MS SQL Server to pick the right RDBMS for their project.

## Understanding Major Differences between MySQL and MS SQL Server

### Supported Platforms

SQL Server was originally developed by Microsoft for Windows operating system exclusively. Microsoft recently announced its decision to make the RDBMS available on both Linux, and Mac OS X (via Docker). Hence, the enterprises now have option to run the database system on three distinct platforms. But they will lack the option to avail certain features while running SQL Server on Linux or Mac OS X. The enterprises can run MySQL smoothly on several popular operating systems including Windows, Linux and Mac OS X.

### Supported Programming Languages

Both MySQL and SQL Server support multiple programming languages. Both RDBMS support Java, PHP, C++, Python, Ruby, Visual Basic, Delphi, Go and R. But MySQL additionally supports programming languages like Perl, Scheme, Tcl, Haskel and Eiffel. The support for many programming languages makes MySQL popular among varying developer communities.

### Storage Engine

MySQL supports a number of storage engines. While using MySQL, the programmers even have option to use a plug-in storage engine. But the earlier versions of the RDBMS supported only non-transactional storage engine. Hence, the programmers working with older versions of the database system need to upgrade the storage engine. At the same time, the developers have to use a single storage engine while working with SQL Server. But they have to switch to the most recent versions of the RDBMS to avail improved storage engine. The multiple storage engine support makes MySQL more flexible than MS SQL Server.

**Filtering**

MySQL allow users to filter out tables, rows, and users in a number of ways. But it requires users to filter out the tables, rows, or users by individual databases. While filtering the data, the developers have to filter database tables individually by running multiple queries. On the other hand, SQL Server enables developers to take advantage of row-based filtering. The row-based filtering option filters data on a database by database way. Also, the filtered data is stored in a separate distribution database. Hence, it becomes easier for programmers to filter multiple rows without considering the number of databases.

**Backup**

While using MySQL, developers have to backup data by extracting all data as SQL statements. The tool provided by the RDBMS further blocks the database while backing up data. The feature reduces chances of data corruption while switching from one version or edition of MySQL to another. But the feature makes the data restoration process time-consuming due to execution of multiple SQL statements. Unlike MySQL, SQL Server does not block the database while backing up data. The feature enables users to backup and restore huge amount of data without putting extra time and effort.

**Option to Stop Query Execution**

MySQL does not allow users to kill or cancel a query when it is running. The users have to kill the entire process to stop SQL query execution. But SQL Server programmers can truncate a database query during execution without killing the entire process. Also, it uses a transactional engine to keep the state consistent. The feature makes SQL Server score over MySQL.

**Security**

Both enterprise database systems are designed as binary collections. MySQL enables developers to manipulate database files through binaries while running. It even allows the database files to be accessed and manipulated by other processes at runtime. But SQL Server does not allow any process to access or manipulate its database files or binaries. It requires users to perform specific functions or manipulate files by running an instance. Hence, the hackers lack the option to access or manipulate data directly. The design rule makes MS SQL Server more secure than MySQL.

**Editions**

The users can choose from two distinct versions of MySQL. They can use either MySQL Community Sever or MySQL Enterprise Server. The community edition of MySQL is open source and free, whereas the enterprise edition comes with a number of proprietary extensions. On the other hand, MS SQL Server is available in several mainstream and specialized editions. The enterprises can choose from enterprise, standard, web, workgroup, or express edition of SQL Server. Likewise, they can also opt for specialized editions of RDBMS including azure, compact, developer, embedded, evaluation, fast track, and localDB.

**As a Software Stack Component**

The enterprises can choose from several editions of MS SQL Server according to the number of concurrent users and requirements of the project. They can even integrate the RDBMS with a variety of proprietary and open source technologies. But MySQL is designed with features complement the needs of modern web applications. Many web application developers use MySQL as a component of LAMP Stack along with Linux operating system, Apache Web Server, and PHP programming language. However, the components of LAMP stack are interchangeable, and users have option to work with multiple programming languages.

On the whole, the enterprises have option to choose from several editions of MySQL and MS SQL Server. The features of the RDBMS differ from one edition to another. Likewise, each of the two popular enterprise database systems has its own pros and cons. Hence, it is always important for users to pick the right edition of MySQL or MS SQL server according to the nature and requirements of individual software development projects.


## 1. Brief Overview

### React

React is a JavaScript library for UI development. It is managed by Facebook and an open-source community of developers.

The framework was introduced in May 2013.

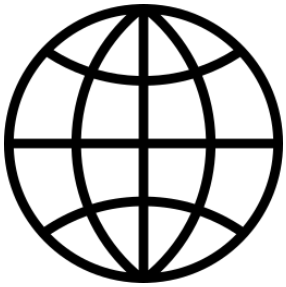
The latest updates were released on August 8th, 2019 – just over a month ago.

### Angular

Angular is an open-sourced JavaScript framework for web and mobile development. It is Type Script-based and managed by Google's Angular Team and the Angular developer community.

Launched in September 2016, Angular (also known as Angular 2.0) is a complete rewrite of AngularJS (Angular 1.0), which was introduced in 2010.

There have been six versions of Angular already, and the latest release took place on August 28th, 2019 – almost three weeks ago.



## 2. Universality

### React

React is a framework used in both web and mobile development. However, for mobile development, it needs to be incorporated with Cordova. Moreover, for mobile development, there is an additional framework – React Native.

React can be used to build both single-page and multiple-page web applications.

### Angular

Angular is suitable for both web and mobile development. In mobile development, however, a great share of work is done by Ionic. Furthermore, similarly to React, Angular has an additional mobile development framework. The counterpart of React Native is NativeScript.

Angular can also be used for both single- and multiple-page web apps.

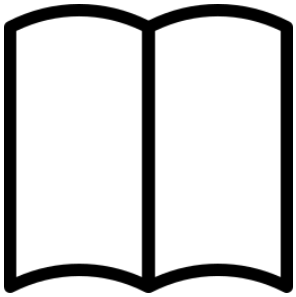
## 3. Self-Sufficiency

### React

React is a framework for UI development, so apps written with React need additional libraries to be used. For instance, Redux, React Router, or Helmet optimize the processes of state management, routing, and interaction with the API. Such functions as data binding, component-based routing, project generation, form validation, or dependency injection require additional modules or libraries to be installed.

### Angular

Angular is a full-fledged framework for software development, which usually does not require additional libraries. All the above-mentioned functions – data binding, component-based routing, project generation, form validation, and dependency injection – can be implemented with the means of Angular package.



**4. Learning Curve**

**React**

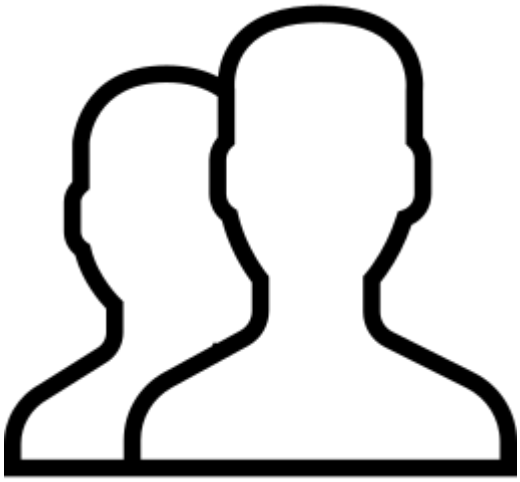
React is minimalistic: no dependency injection, no classic templates, no overly complicated features. The framework will be quite simple to understand if you already know JavaScript well.

However, it takes quite some time to learn how to set up a project because there is no predefined project structure. You also need to learn the Redux library, which is used in more than half of React applications for state management. Constant framework updates also require additional effort from the developer. Furthermore, there are quite a lot of best practices in React, which you will need to learn to do things right.

**Angular**

Angular itself is a huge library, and learning all the concepts associated with it will take much more time than in the case of React. Angular is more complex to understand, there is a lot of unnecessary syntax, and component management is intricate. Some complicated features are embedded into the framework core, which means that the developer cannot avoid learning and using them. Moreover, there are a lot of ways of solving a single issue.

Although TypeScript closely resembles JavaScript, it also takes some time to learn. Since the framework is constantly updated, the developer needs to put some extra learning effort.



**5. Community**

**React**

React framework is one of the most popular JS frameworks worldwide, and the community supporting and developing it is huge.

Working with React, you have to be a continuous learner since the framework is often updated. While the community tries to go forward with the latest documentation as swiftly as possible, keeping up with all the changes is not that easy. Sometimes, there may be a lack of documentation, but the issue is often solved by the community support on thematic forums.

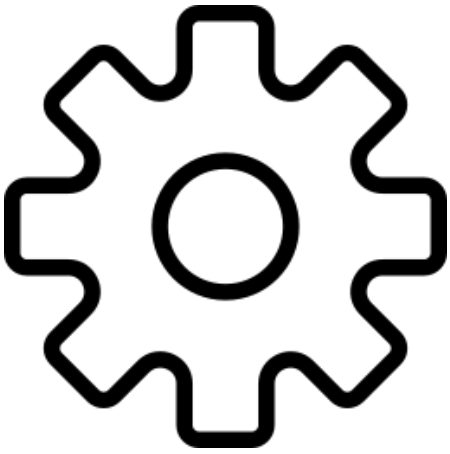
React is actively used by such companies as Facebook, Twitter, Netflix, Airbnb, PayPal, The New York Times, Yahoo, Walmart, Uber, and Microsoft.

**Angular**

Angular is less admired than React and faces a lot of skepticism, partially because of the unpopularity of Angular 1.0. Developers used to dismiss the framework as an overly complicated one as it required a lot of time to be spent learning. However, this framework has been developed by Google, which works in favor of Angular's credibility.

Google provides the long-term support of the framework and constantly improves it. However, the updates are so fast that the documentation often falls behind.

Angular is used by such companies as McDonald's, AT&T, HBO, Apple, Forbes, Adobe, Nike, and Microsoft as well.



**6. Performance**

**React**

React's performance is greatly improved with the introduction of the virtual DOM. Since all virtual DOM trees are lightweight and built on server, the load on browser is reduced. Furthermore, since the data-binding process is unidirectional, bindings are not assigned watchers as in the case of Angular. Respectively, no additional workload is created.

**Angular**

Angular performs worse, especially in the case of complex and dynamic web apps.

The performance of Angular apps is negatively affected by bidirectional data-binding. Each binding is assigned a watcher to track changes, and each loop continues until all the watchers and associated values are checked. Because of this, the more bindings you have, the more watchers are created, and the more cumbersome the process becomes.

However, the most recent update of Angular has greatly improved its performance, and it does not lose to React anymore. Moreover, the size of an Angular application is slightly smaller than the size of a React app.

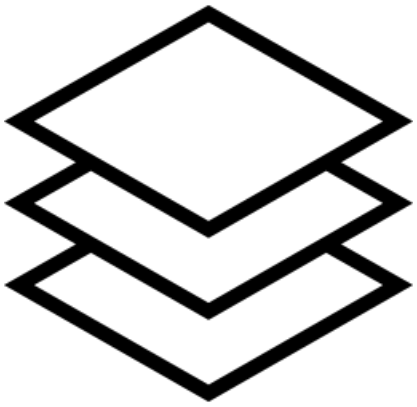
**7. Language**

**React**

React is based on JavaScript ES6+ combined with JSX script. JSX is an extension for syntax, which makes a JavaScript code resemble that written in HTML. This makes the code easier to understand, and typos are easier to spot. For the JSX code to be compiled in a browser, React is augmented with Babel – a code translation tool.

**Angular**

Angular can use JavaScript or TypeScript, which is a superset of JS developed specifically for larger projects. TypeScript is more compact than JavaScript, the code is easier to navigate, and typos are easily spotted. Code refactoring process also becomes simpler and faster.



**8. App Structure**

**React**

The structure of React provides developers with the freedom to choose. There is no “the only right structure” for a React app. However, the necessity to design the app structure at the beginning of each project makes it more difficult and longer to start.

Besides, React offers only View layer, while Model and Controller are added with the usage of other libraries.

The architecture of a React app is component-based. The code is made of React components, which are rendered with React DOM library and directed in two ways: functional (with a function that returns JSX)...

```
function Hello(props){
return <div>Hello {props.name}</div>
}
```

...and class-based (with ES6 classes).

```
class Hello extends React.Component {
render() {
return <h1>Hello, {this.props.name}</h1>;
}
```

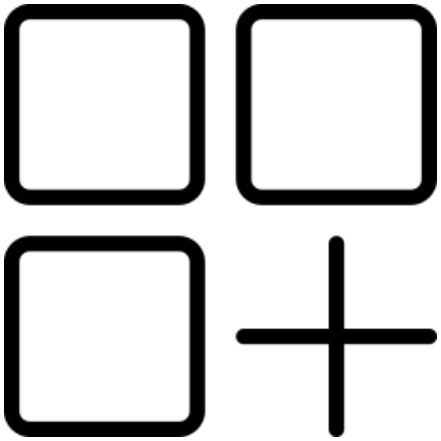
**Angular**

The structure of Angular is fixed and complex, suitable for experienced developers.

Angular is based on three layers – Model, Controller, and View. An object responsible for the Model is initialized by the Controller and displayed with the View.

The application code consists of different Angular components, each being written in four separate files: a TypeScript to implement the component, an HTML file to define the view, a CSS file to define the stylistic features, and a special file for testing purposes. Links to these files are written in the app directive, which displays the structural logic of the app. Respectively, Angular components are also reusable.

```
import { Component } from '@angular/core';
@Component({
selector: 'my-app',
templateUrl: './app.component.html',
styleUrls: ['./app.component.css']
})
export class AppComponent { }
```



**9. UI Components**

**React**

UI tools for React are developed by the community. There are a lot of free and paid UI components on the React portal. To use material design components in React, you would have to install an additional library – Material-UI Library & Dependencies.

**Angular**

Angular has a built-in Material toolset, and it offers a variety of pre-built material design components. There are various buttons, layouts, indicators, pop-ups, and form controls. Because of this, UI configuration becomes simpler and faster.

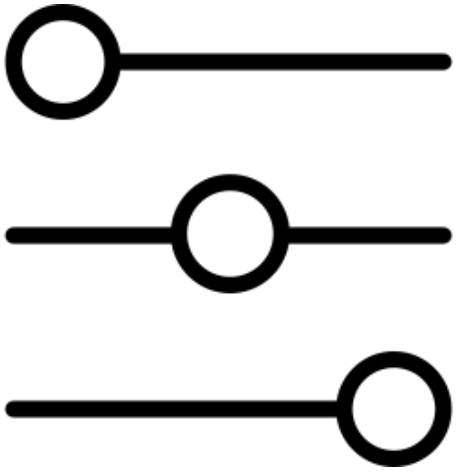
**10. Directives**

**React**

In React, templates and logic are explained in one place – at the end of the component. It allows the reader to quickly grasp the meaning of the code even if they do not know the syntax.

**Angular**

In Angular, each template is returned with an attribute – a directive of how the object has to be set. The syntax of Angular directives is complex and sophisticated, which makes it incomprehensible for a reader without any experience in working with this technology.



**11. State Management**

**React**

In React, each component has its own state. A React developer can create special components for holding the state of the entire application or a particular part of it. The major disadvantage here consists in the fact that the global state needs to be stored in multiple different parts of the app with data being manually passed around different component tree levels.

```
class Clock extends React.Component {
  constructor(props) {
    super(props);
   this.state = {date: new Date()};
 }
  render() {
    return (
      <div>
        <h1>Hello world!</h1>
        <h2>Now is {this.state.date.toLocaleTimeString()}.</h2>
      </div>
    );
  }
}
```

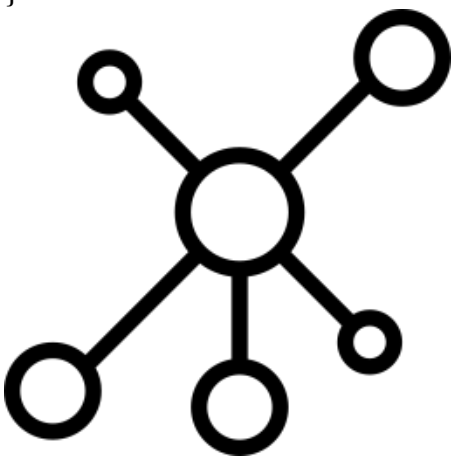
To solve this problem, there is a special state management library – Redux. The idea of it is that the global state is represented as a single stateful object, which is altered in different parts of the app with the help of reducers – special Redux functions.

Another solution is offered by the state management library MobX. Unlike Redux with the global state stored in a single immutable stateful object, MobX offers storing only the minimal required state, while the rest of it can be derived.

**Angular**

In Angular, component data is stored in component properties. Parent components pass data through to children ones. State changes in some parts can be identified and recalculated, but in a large app, it can cause a multi-directional tree series of updates, which will be difficult to track. The features can be improved with the help of state management libraries, such as NgRx or RxJS , which would make sure that the data flow is unidirectional.

```
export class HeroListComponent implements OnInit {
  heroes: Hero[];
  selectedHero: Hero;
  constructor(private service: HeroService) { }
  ngOnInit() {
    this.heroes = this.service.getHeroes();
  }
  selectHero(hero: Hero) { this.selectedHero = hero; }
}
```



**12. Dependency Injection**

**React**

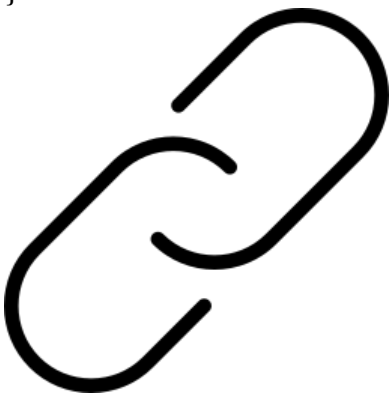
React does not fully support dependency injection as it does not fully comply with the idea of functional programming and data immutability. Instead, it has a global state for all components.

**Angular**

The greatest advantage of Angular rests in the fact that, unlike React, it supports dependency injection. Therefore, Angular allows having different lifecycles for different stores.

```
import { Injectable } from '@angular/core';
import { HEROES } from './mock-heroes';
@Injectable({
  // we declare that this service should be created
  // by the root application injector.
  providedIn: 'root',
})
```

```
export class HeroService {
  getHeroes() { return HEROES; }
}
```



**13. Data Binding**

**React**

Data binding stands for the data synchronization process between Model and View. React should be augmented with Redux, which allows you to work with immutable data and makes data flow unidirectional. Unidirectional binding is predictable, which facilitates the debugging process.

**Angular**

Angular works with bidirectional data-binding and mutable data. While the advantages of mutable and immutable data are a matter of a heated discussion, it is definitely easier to work with bidirectional data-binding rather than with the unidirectional approach. At the same time, bidirectional data-binding negatively affects the performance since Angular automatically develops a watcher for each binding.

The ways of data-binding in Angular:

{{expression}} Interpolation
[target]="expression" Property
bind-target="expression" Attribute
(target)="statement" Event
on-target="statement" Event
[(target)]="expression" Two-way
bindon-target="expression" Two-way

## 14. Change Rendering

### React

React uses a virtual Document Object Model (DOM), which enables easily implementing minor data changes in one element without updating the structure of the entire tree. The framework creates an in-memory cache of data structure, computes the changes, and efficiently updates the DOM displayed in the browser. This way, the entire page seems to be rendered on each change, whereas actually, the libraries render changed subcomponents only.

The React team is constantly improving Fiber – a mechanism aimed at boosting the productivity of change rendering.

### Angular

Angular uses a real DOM, which updates the entire tree structure even when the changes have taken place in a single element. The real DOM is considered to be slower and less effective than the virtual DOM.

To compensate for this disadvantage, Angular uses change detection to identify components that need to be altered. Therefore, the real DOM on Angular performs as effectively as the virtual DOM on React.

## 15. Tools

### React

React is supported by multiple code editors. For instance, the code in React can be edited with Sublime Text, Visual Studio, and Atom. To bootstrap a project, you can use the Create

React App (CLI) tool. In turn, server-side rendering is completed with the use of Next.js framework.
To test the entire app written in React, you would need multiple tools. For instance, Enzyme for component testing, Jest for testing JS code, React-unit for unit testing and so on. To debug the app in the development mode, you can use a browser extension React Dev Tools.
Another interesting tool is React 360, which is a library used for creating AR and VR applications.

**Angular**

Similarly to React, Angular is supported by a variety of code editing tools. For example, you may work with such code editors as Aptana, Sublime Text, and Visual Studio. A project can be promptly set up with Angular CLI. Server-side rendering is completed with the help of Angular Universal.
Unlike React, Angular can be fully tested with a single tool. For the end-to-end testing in Angular, the platforms are Jasmine, Protractor, and Karma. Another tool that debugs the app in the development mode is a browser extension Augury.

**To Wrap Up**

**Angular** is a full-fledged mobile and web development framework. **React** is a framework only for UI development, which can be turned into a full-fledged solution with the help of additional libraries.
React seems simpler at first sight, and it takes less time to start working on a React project. However, that simplicity as the main advantage of React is neutralized because you have to learn to work with additional JavaScript frameworks and tools.

Angular itself is more complex and takes quite some time to master. Yet, it is a powerful tool that offers a holistic web development experience, and once you learn how to work with it, you reap the fruits.

**There is no better framework. Both are updated continuously to keep up with the competitor**. For instance, while React was believed to win because of its virtual DOM, Angular equaled the score by implementing change detection. While Angular was considered to be winning because of being developed by such an authoritative company as Google, the immense devoted React community fully compensated for Google's reputation and made React similar to Angular.
**In the end, React vs Angular is all a matter of personal preference, a matter of skills and habits**. As a beginner in programming, you would probably benefit more from starting with React. As an experienced developer, you just keep working with what you know better.
Do not forget to challenge yourself and start learning a new framework, React or Angular. As a Project Manager or a business owner outsourcing developers, you should talk to your web development team and together choose the framework that suits all of you best, whether it be Angular or React.

**References:**

https://www.ijarcce.com/upload/2015/march-15/IJARCCE%2039.pdf

https://www.hostinger.in/tutorials/difference-between-mysql-and-sql-server

https://www.freecodecamp.org/news/angular-vs-react-what-to-choose-for-your-app-2/