TypeScript:

- TypeScript is a superset of JavaScript. TypeScript offers optional static typing, classes and interfaces.
- Unlike JavaScript, TypeScript has a certain standardized way of writing code which you must adhere. So, it gives lots of robustness and maintainability over dynamically typed JavaScript in a complex project.

Is it possible to create classes in JavaScripts?

- JavaScript is a class-less language. So, you cannot create classes (interfaces) like in TypeScript. However there are ways that you can do this. Such as using a function to manipulate class.
- Traditional JavaScript uses functions and prototype-based inheritance to build up reusable components, but this may feel a bit awkward to programmers more comfortable with an object-oriented approach, where classes inherit functionality and objects are built from these classes.

When to use Class (Introduced officially by ES6 into JS ecosystem):

- Creating multiple instances.
- Using inheritance
- Singleton objects

Interface:

- TS does nothing with the interfaces.
- It just sees if you respect the contracts “at compile time”.

Playground:

- You write typescript on the left and it shows you the JS code on the right.

TS to JS:

- Classes are changes into functions.
- Instance methods are set into the prototype
- Static methods are assigned to the constructor function.
    - This is why we can access static methods anytime, but they are not accessible from an instance.

Transpilers:

- A source-to-source compilers.
- tools that read source code written in one programming language, and produce the equivalent code in another language.

Advantages:

Types:

- First of all it provides types! This powerful thing allow you to write a code and check it without even looking to output code. It helps you with writing better code, with less common bad practices like changing type of variable after initialization.

Transpiling:

- Now a days programmers are starting to use the latest JS version when programming in JS.
- However, browsers are not yet fully compatible with such JavaScript level.
- To circumvent incompatibilities, code is written in ES7 and transpiled[2] to ES5 by using Babeljs[3] during development.
- In a way, TypeScript[4] does the same thing as Babeljs, it transpiles JavaScript to a more compatible JavaScript version.
- The main advantage of writing code in TypeScript before transpiling, as opposed to writing ES7 JavaScript and transpiling it with Babeljs, is that TypeScript provides static-type-checking[5].
- You can create classes in TypeScript as much as you can create classes in ES7.
- Both languages are prototype-based. Therefore, classes are merely syntactic-sugar[7] for a prototype-based architecture. In other words, classes are converted to objects, and methods are automatically placed in the prototype chain.

Optional Static Typing:

- Most likely, the first thing that comes to mind with TypeScript is the optional static type system that it provides.
- Types can be added to variables, functions, properties, etc.
- This will help the compiler and show warnings about any potential errors in code, before an app is ever run.
- Types also help when using libraries and frameworks, as they let developers know exactly what type of data APIs expect.
- The key thing to remember about the type system is that it is optional. TypeScript does not force developers to add types they don’t want to add.

Intellisense:

- One of the biggest advantages of TypeScript is its code completion and IntelliSense.
- Intellisense provides active hints as code is added.
- All the best IDE’s available today have support for code completion, including Atom, Sublime text.

Cross-Browser Compatibility:

- How can we make sure that our website give smooth and seamless user experience across all browsers that your target audience may have access to, be it Firefox, chrome, or even internet explorer.

How to identify and prevent these issues:

- Find crossbrowser compatibility issues with different browsers.
- Use a validator may help you figure out some mistakes that seem so simple: For HTML,
  - it validates the tags are closed and nested,
  - a DOCTYPE is used,
  - and tags are used correctly
  - while for CSS it checks that the property names and values are spelled correctly,
  - no curly braces are missed, and other CSS properties.
- Use a linter in code editor.
- Check for the Doctype:
  - Doctype basically helps your browser to recognize in which language is your website's code written.
  - If you don't specify that, some of the smart browsers will understand it themselves but some dumb browser will not be able to figure out what happened, and they will render some element of your website in a way that you would not like.
- CSS Reset: When you apply CSS reset, you tell every browser to remove the styling to default CSS that causes cross browser incompatibility. make sure that you add your reset stylesheet before your main stylesheet.
- Use separate stylesheets for different browsers.
- Prefer cross browser friendly frameworks like JQuery, AngularJS, ReactJS.
- Don't miss cross browser compatibility testing before promoting to production.

Polyfills:

- If the feature exists in the browser, the polyfill lets the browser do its thing, if not, the polyfill steps in to plug the missing functionality.

Polyfills vs Transpiling:

- It takes your shiny new syntax and spits out plain old-fashioned ES5. If you want to utilize arrow functions, async/await, rest and spread parameters, classes, et al in your code, you need to transpile your ES6 code into ES5 with a tool such as Babel.

- Ideally, we should only polyfill features we've actually used and only send those polyfills that a particular browser actually needs. Polyfill.io can achieve both those needs. Rather than delivering bloat to modern browsers, the service reads the User-Agent HTTP header so that it can deliver only what is necessary. New browsers will receive an almost empty file, old versions of IE will receive a lot of code.

CSS Inheritance:

- The CSS properties will be inherited from the parent element to the child element.
- The default values for many properties is already inherit.

When an element using multiple classes for the same CSS property, the tag that comes at the end takes effect.

Pseudo-elements:

CSS pseudo-element is used to style specified parts of an element.

- Selector::first-line
- Selector::first-letter
- Selector::before
- Selector::after
- selector::selection

How to improve website performance:

- Clean up the HTML Document: Write in a manner that it is concise and effective.
- When it comes to referencing other resources within your HTML document there are a few best practices you should follow.
  - Proper CSS Placement: it is recommended to put CSS at the top of your HTML document's header in order to ensure progressive rendering.
  - Proper JavaScript Placement:On the other hand, if you place JavaScript attributes within the head tag or near the top of the HTML document, you will block the loading process of HTML and CSS elements. This mistake can cause visitors wait on a blank page, and therefore may

CORS:

- A request for a resource (like an image or a font) outside of the origin is known as a cross-origin request. CORS (cross-origin resource sharing) manages cross-origin requests.
- CORS allows servers to specify who (i.e., which origins) can access the assets on the server, among many other things.
- The CORS standard manages cross-origin requests by adding new HTTP headers to the standard list of headers. The following are the new HTTP headers added by the CORS standard:
  - Access-Control-Allow-Origin
  - Access-Control-Allow-Credentials
  - Access-Control-Allow-Headers
  - Access-Control-Allow-Methods

impatiently abandon your site. You can avoid this issue by placing JavaScript attributes at the bottom of your HTML.
        - Prefer Async script loading.
        - Defer JS loading.
        - Optimize CSS Performance: Many CSS options require HTTP requests (unless using inline CSS), so you should make an effort to minimize bloated CSS files without eliminating vital features.
    - Reduce External HTTP Requests:
        - A large portion of a website's load time comes from external HTTP requests. The speed at which an external resource loads can vary depending on the hosting provider's server infrastructure, location, etc. Your first goal when reducing external HTTP requests should be to examine your website with a minimalist outlook.
        - Eliminate:
            - Unnecessary images
            - Unnecessary JavaScript
            - Excessive CSS
            - Unnecessary plugins
    - Minify CSS, JS.
    - Increase Speed with CDN and Caching.
    - Optimize Images.

NOSQL databases:

- NoSQL encompasses a wide variety of different database technologies that were developed in response to the demands presented in building modern applications.

NoSQL Database Types

- Document databases pair each key with a complex data structure known as a document. Documents can contain many different key-value pairs, or key-array pairs, or even nested documents.

- Graph stores are used to store information about networks of data, such as social connections. Graph stores include Neo4J and Giraph.

- Key-value stores are the simplest NoSQL databases. Every single item in the database is stored as an attribute name (or 'key'), together with its value. Examples of key-value stores are Riak and Berkeley DB. Some key-value stores, such as Redis, allow each value to have a type, such as 'integer', which adds functionality.

- Wide-column stores such as Cassandra and HBase are optimized for queries over large datasets, and store

    - Access-Control-Expose-Headers
    - Access-Control-Max-Age
    - Access-Control-Request-Headers
    - Access-Control-Request-Method
    - Origin
- There is a cors middleware for express.

GitHub Flow:

- Create a Git repository for every new project
- Create a Branch for every new feature
- Add a commit
- Open a Pull Request to merge to master
- Discuss and review your code
- Deploy
- Merge to master

AGILE:

Scrum Model Lifecycle:

- Product backlog:
    - is a list that consists of features that should be implemented during the development process.
    - It's ordered by priority and its every item is called a User story.
    - The description of every user story should include the following required fields:
        - Importance of a user story. It's acceptable to use any number you want
        - Initial estimate describes the overall capacity of work. It's measured in story points
        - How to demo. Describes the way of how the working product will be demonstrated

- Sprint Planning and Sprint Backlog Creation:
    - Firstly, you should determine what your sprint's duration will be.
    - A short sprint allows you to release the working version of a product more frequently.
    - As a rule, a sprint lasts about 2 weeks.
    - The product owner determines the importance of a proper user story, while the scrum team defines the appropriate labor costs.
    - After that, the scrum team can select the most important user stories from the product backlog.

- Working on the Sprint. Scrum Meetings:
    - the website development process begins.
    - To track the current working process, a task board is commonly used. There are usually big cards with the names of particular user stories and a bundle of little sticky notes with a description of single tasks which are needed for

columns of data together, instead of rows.

The Benefits of NoSQL

- When compared to relational databases, NoSQL databases are more scalable and provide superior performance, and their data model addresses several issues that the relational model is not designed to address:
- Large volumes of rapidly changing structured, semi-structured, and unstructured data
- Agile sprints, quick schema iteration, and frequent code pushes
- Object-oriented programming that is easy to use and flexible
- Geographically distributed scale-out architecture instead of expensive, monolithic architecture

What is Spring?

- Simply put, the Spring framework provides comprehensive infrastructure support for developing Java applications.
- It's packed with some nice features like Dependency Injection and out of the box modules like:
  - Spring JDBC
  - Spring MVC
  - Spring Security
  - Spring AOP
  - Spring ORM
  - Spring Test

implementation of this or that story.

- Testing and Product Demonstration:
  - Since the ideal result of every sprint is a working product, the full life-cycle testing process is very important.
  - The result of every sprint is product demonstration. The Scrum team creates a review and demonstrates the results of their work.
- Retrospective and Next Sprint Planning
  - Retrospective's main aim is to discuss the results and determine the ways how to improve development process on the next step. The team should conclude what went well during the working process and what can be done better during the future iteration.

Spring Framework 5:

- JDK baseline update
- Core framework revision
- Core container updates
- Functional programming with Kotlin
- Reactive Programming Model
- Testing improvements
- Library support
- Discontinued support

What is SpringBoot?

- Spring Boot is basically an extension of the Spring framework which eliminated the boilerplate configurations required for setting up a Spring application.
- It takes an opinionated view of the Spring platform which paved the way for a faster and more efficient development eco-system.
- Here are just a few of the features in Spring Boot:
  - Opinionated 'starter' dependencies to simplify build and application configuration
  - Embedded server to avoid complexity in application deployment
  - Metrics, Helth check, and externalized configuration
  - Automatic config for Spring functionality – whenever possible

Note: Information gathered in this document has been collected from various sources on the Internet.