프론트엔드 프레임워크 수업 소개

**목차**

[1. 수업 계획 2](#_Toc156778633)

[1) 선수과목 2](#_Toc156778634)

[2) 수업 내용 2](#_Toc156778635)

[2. 개발도구 설치 3](#_Toc156778636)

[3. 프론트엔드 관련 기술 4](#_Toc156778637)

[1) Javascript 언어 버전 4](#_Toc156778638)

[2) WebAssembly 5](#_Toc156778639)

[3) 하이브리드 앱 (hybrid app) 6](#_Toc156778640)

[4) 하이브리드 앱 개발 기술의 종류 6](#_Toc156778641)

# 수업 계획

## 선수과목

### Javascript 언어 필수

Java 언어 문법과 Typescript 언어 문법을 알고 있으면 도움이 됨.

### HTML CSS 필수

HTML CSS 등 이미 잘 안다고 전제하고 React를 강의함

### 웹프로그래밍

언어 기술 무관 어떤 웹프로그래밍을 할 줄 알아야 진도를 따라올 수 있다.

## 수업 내용

### Typescript 언어

### React.js 프레임웍

React, TanStack Query, Next.js

# 개발도구 설치

M1 맥북 개발도구 설치.docx

Windows 개발도구 설치.docx

위 문서를 참고하여 아래 SW들을 설치하라.

- node.js

- Visual Studio Code

- git 클라이언트

# 프론트엔드 관련 기술

## Javascript 언어 버전

Javascript 문법 표준을 ECMAScript 라고 부른다. 줄여서 ES.

### ES5 (ECMAScript5)

Javascript 문법 표준의 5번째 버전을, ECMAScript5,줄여서 ES5 라고 부른다.

2009년 발표됨.

가장 많은 웹브라우저가 지원하는 보편적인 표준.

객체지향 문법이 빠져있다.

### ES6 (ECMAScript6)

Javascript 문법 표준의 6번째 버전

문법 표준 발표 연도를 붙여서 ES**2015**, ECMAScript**2015** 라고 부르기도 한다.

객체지향 문법 추가됨.

ES5 문법에 비해서 크게 변함.

프론트엔드 개발자라면 최소한 ES6 문법으로 개발해야 한다.

### 자잘한 버전 업

ES6 문법에서 자잘한 버전 업.

ES7, ES2016

ES8, ES2017

ES9, ES2018

ES10, ES2019

ES11, ES2020

ES12, ES2021

ES6에서 자잘한 버전업들이기 때문에,

위 표준들을 전부 뭉뚱그려서 ES6 라고 부른다.

### Typescript

마이크로소프트가 만든 언어.

Java 언어에는 자료형(type)이 있지만, Javascript에는 없다.

(예: int, double, String, ArrayList)

Java 언어에 있는 자료형과 객체지향 문법과 유사한 것을

Javascipt에 추가한 것이 Typescript 이다.

그래서, Java 개발자에게는 순수 Javascript 문법보다 Typescript 문법이 더 친근하다.

프론트엔드 개발자라면 Typescript 문법으로 개발하는 것이 바람직하다.

## WebAssembly

이제 막 나와서 아직 대세는 아니지만, 미래 대세가 될 가능성이 높음

Java 소스코드를 컴파일한 바이트코드를 Java Virtual Machine (JVM)이 실행하 듯이,

C/C++, Java, C#, Python 등 고급 언어 소스코드를 컴파일한 WebAssembly를 웹브라우저가 실행한다.

(가) C/C++ 소스코드를 기계어코드로 컴파일하고, 그 실행 파일을 CPU가 직접 실행

(나) Java 소스코드를 바이트코드로 컴파일하고, 그 결과를 JVM이 실행

(다) C/C++, Java, C# 등 고급 언어 소스코드를 WebAssembly로 컴파일하고, 그 결과를 웹브라우저가 실행

(가) 방식이 가장 빠르지만, (나) (다) 방식도 꽤나 빨라서 (가)와 큰 차이가 없다.

### WebAssembly 기술의 장점

C/C++, Java, C#, Python 등 고급 언어로 구현된 수 많은 소스코드를 가져와서,

WebAssembly로 컴파일하면 웹브라우저에서 실행된다.

웹브라우저가 그야말로 운영체제처럼 진화할 수 있다.

예를 들어 안드로이드 앱을 WebAssembly로 컴파일하여 웹브라우저에서 실행.

어지간한 Windows 앱도 WebAssembly로 컴파일하여 웹브라우저에서 실행.

지금까지 프론트엔드 개발 언어는 Javascript 언어뿐이었지만,

앞으로는 다양한 언어로 개발할 수 있다.

하지만 당분간은 Javascript 언어가 대세.

## 하이브리드 앱 (hybrid app)

프론트엔드 개발 기술을 사용하여 모바일 앱도 구현할 수 있다.

이런 앱들을 하이브리드 앱이라고 한다.

(예: 네이버 앱, facebook 앱)

### 장점

프론트엔드 개발 기술을 사용하여,

모바일 웹사이트와 안드로이드 앱, 아이폰 앱을 한 번에 구현할 수 있다.

### 단점

하이브리드 앱이 네이티브 앱보다 약간 느리다.

## 하이브리드 앱 개발 기술의 종류

### Phonegap

Adobe가 개발함.

ES5 + jQuery로 개발.

React Native 전에는 대세 기술이었음.

### React Native

facebook이 개발함.

ES6 or Typescript 언어 + React.js로 하이브리드 앱 개발.

현재 가장 널리 사용됨

### Xamarin

Microsoft가 개발함.

C# 언어로 개발.

Unity 3D 게임 엔진과 더불어, 가장 대표적인 모바일 게임 개발 기술.

모바일 게임 분야는 꽉잡고 있지만, 범용 앱 분야까지 대세가 될 확률은 낮다.

### Electron

프론트엔드 개발 기술을 사용하여 PC 데스크탑 앱을 만든다. (Windows, Mac, Linux)

ES6 or Typescript 언어 + React.js or Vue.js

Visual Studio Code가 Electron으로 개발된 하이브리드 앱이다.

### Flutter

Google이 개발함.

Dart 언어로 개발.

React Native 보다 빠르다.

모바일 앱뿐만 아니라, PC 데스트탑 앱도 만들 수 있다. (Windows, Mac, Linux)

React Native의 유력한 대안으로 기대되지만 아직은 아님.