**Node.js 웹 프로그래밍 SCHOOL 1기 사전과제**

제출일 : 170707

작성자 : 김세준

1. **[영타 연습]** 강사님의 코드 속도를 따라오지 못해서 강의가 빠르다는 피드백이 종종 발생합니다. 개강 전까지 충분한 연습을 통해 영타 속도를 개선해 오셔야 합니다. **Words per Minutes가 50개 이상, 정확도 90% 이상**으로 기록된 영타 **스크린샷 총 3장 (HTML 1장, CSS 1장, JavaScript 1장)** 을 찍어주세요.

<http://www.speedcoder.net> **→** 접속 후 로그인 (페이스북, 구글 등) → **HTML, CSS, JavaScript** 선택.

|  |  |
| --- | --- |
| 인증 스크린샷  (예시) | ../../Downloads/html.jpg../../Downloads/jsp.jpg../../Downloads/7702.tmp.jpg |

**2. [프로그램 설치]** 4개월간 학습과정을 거치면서 다양한 문제에 부딪치게 됩니다. 기본적인 Tool사용법부터 코드 에러 등 원인을 알 수 없는 각종 오류와 맞닥뜨리게 됩니다. 중요한 것은 이 때 스스로 구글링을 통해서 이를 해결하는 능력입니다.

따라서 4개월간 사용할 Tool 설치와 환경설정을 사전에 요청드립니다. 아래의 Tool들을 설치하시고 정상 작동화면을 스크린샷 첨부해 주십시오. 또한 이 과정에서 발생하는 문제점과 해결과정을 상세히 기술하여 주시기 바랍니다.

* Visual Studio Code : <https://code.visualstudio.com/>

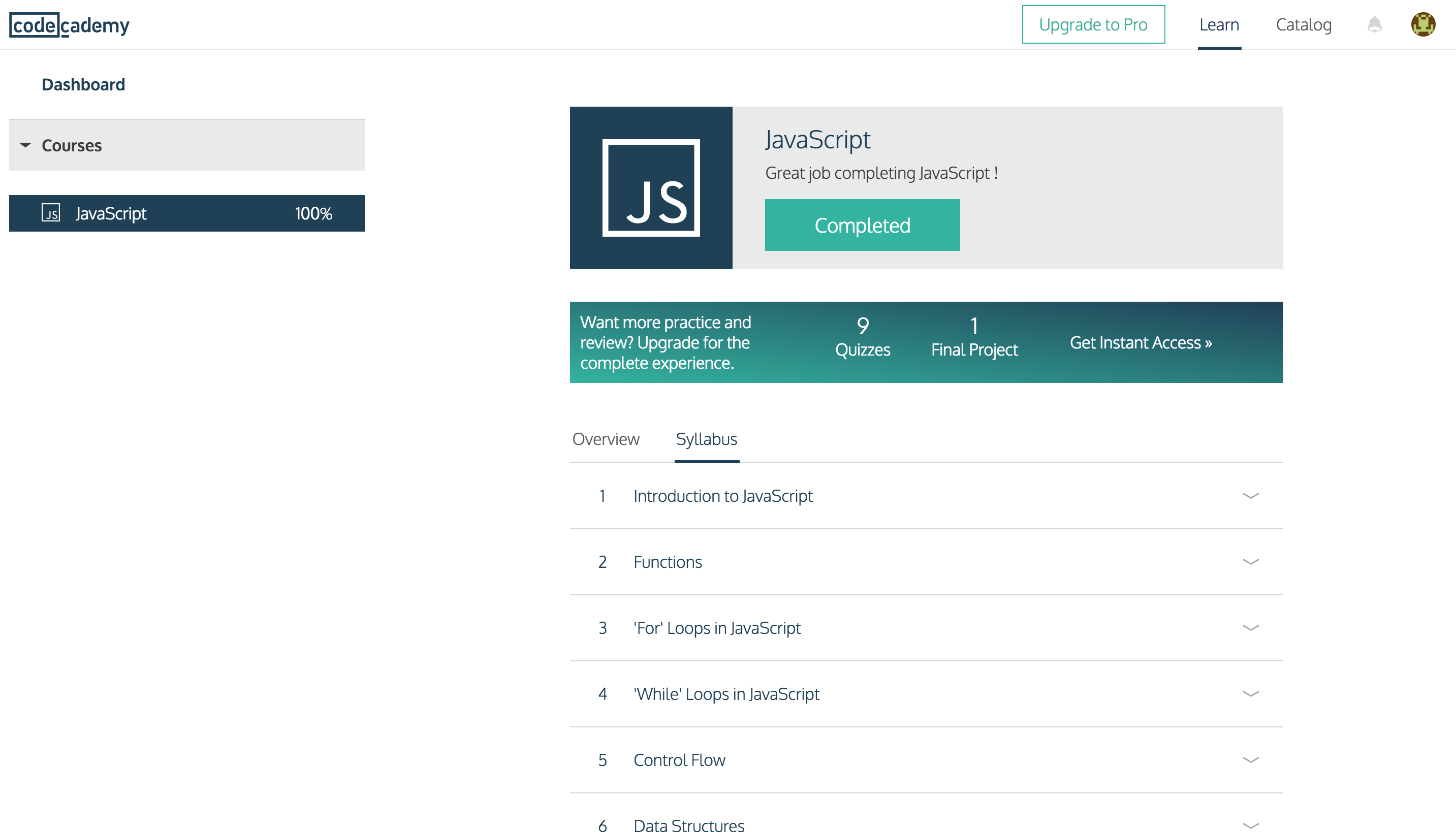
|  |  |
| --- | --- |
| 설치일 | 170707 |
| 설치 소요 시간 | 5분 |
| 문제 상황 | 없음 |
| 해결 방법 |  |
| 정상 작동화면 |  |

**3. [JavaScript 학습]** Node.js 웹 프로그래밍 SCHOOL에서 사용할 언어인 JavaScript의 기초적인 내용을 학습하는 과제입니다 **Codecademy (영문, 실습 위주)** 혹은 **Tryhelloworld (한글, 동영상 강의 위주)** 중 **하나를 택**하셔서, JavaScript 강좌 수강을 완료하신 후 인증해주시면 됩니다.

**[ Option 1: Codecademy 실습 ]**

1) codecademy 홈페이지에 접속하여 우측 상단의 회원가입(Sign up) 혹은 로그인을 해주세요.

- <https://www.codecademy.com/learn/javascript>  
\*codecademy 에는 두 종류의 자바스크립트 과정이 있습니다. Preview 말고, 10시간 짜리 정식 코스를 수행해주시면 됩니다.

2) 로그인 후 start를 누르시고 학습을 시작해주세요. 왼쪽의 설명을 보며 코드를 완성해주세요. 퀴즈를 풀듯이 설명이 잘 되어 있습니다.   
3) 모든 단계를 100% 완료하면 START 에서 COMPLETED 로 변경됩니다. 이를 캡처하여 인증샷을 보내주세요.   


4. **[에세이]** 아래 내용을 조사하셔서, 자신만의 언어로 짧은 글을 써주세요. 복사-붙여넣기식 작성은 지양해주세요. 참고한 페이지의 링크 또한 함께 적어주시기 바랍니다. 아래 키워드는 커리큘럼 상, 저희가 다루게 될 내용입니다. 각 키워드를 구글에 검색해보시고, 정의, 장점, 사용한 예시 등을 작성해주세요. (A4 2장 이내) 잘 모르는 부분이 있다면, 무엇을 모르는지도 함께 적어주시면 좋습니다.

1. 클라이언트와 서버 (Client & Server)

서버 : 일반적인 클라이언트에게 네트웍을 통해 서비스를 제공하는 시스템 웹페이지, 사이트, 앱을 저장하는 컴퓨터. 클라이언트의 장비가 웹페이지에 접근하길 원할 때, 서버로부터 클라이언트의 장치로 사용자의 웹브라우저에서 보여지기 위한 웹페이지의 사본이 다운된다.

클라이언트 : 서비스를 사용하는 사용자 일반적인 웹 사용자의 인터넷이 연결된 장치들과 이런 장치들에서 이용가능한 웹에 접근하는 소프트웨어ex)파이어폭스, 크롬.

1. 크롬 개발자 도구 (Chrome DevTools)

구글에서 만든 웹브라우저인 크롬의 기능중 하나. 개발을 도와주는 다양한 도구가 기본적으로 제공된다. 개발자도구를 이용하면 HTML.CSS.Javascript의 생산성을 극대화 할수 있다. 경험에 의하면 nodejs를 사용할 때 jade로 쉽게 에디트할 수 있었지만, 개발자도구로 더욱 정확하게 현재 코드가 어떻게 진행되는지 볼 수 있었다.

1. 렌더링과 파싱 (Rendering & Parsing)

클라이언트 사이드 렌더링의 경우, 사용자의 행동에 따라 필요한 부분만 다시 읽어드린다. 그래서 서버측에서 렌더링해 전체 페이지를 다시 읽어들이는 것 보다 빠른 인터렉션을 기대한다. Ajax기능을 위해 클라이언트 렌더링 요소가 포함될 수 밖에 없다. 클라이언트 사이드 렌더링은 페이지를 읽어 드리는 시간, 자바스크립트를 읽어드리는 시간, 자바스트크립트가 화면을 그리는 시간까지 모두 마쳐야 사용자에게 보여진다. 단점 : 초기구동속도가 느리다. 보안문제가 추가적으로 발생하고 검색 엔진 최적화의 문제가 존재한다. 서버사이드 렌더링 장점은 반대의 경우이다. 앵귤러2가 등장하고. 웹 컴포넌트에 대한 표준을 위해 여전히 노력하고있다.

1. SPA Single Page Application (React, Vue.js 등)

단일 페이지 어플리케이션은 브라우저에 로드되고 난 뒤에 페이지 전체를 서버에 요청하는것이 아니라 최초한번 페이지전체를 로딩한후 이후부턴 데이터만 변경해서 사용할 수 있는 웹 어플리케이션을 말한다.

SPA는 Angular Js, React, 등이 존재하며 AngularJS 는 SPA 를 만들때 도움을 주는 프레임워크이고 자바스크립트 기반의 MV\* 오픈소스 프레임워크라고도 합니다.

1. Node.js 프레임워크 (Express, LoopBack 등)

<NPM> http, os = nodejs가 제공하는 모듈 자바스크립트가 제공하는 모듈은 따로 있다. 자바스크립트가 동작하는 어떤 시스템에서건 사용할 수 있는 모듈 기본적인 기능 : 업그레이드나, 설치, 삭제, 의존성관리 올려준다. 글로벌 : 이 컴퓨터 전역에 설치 g가 없다면 : 프로젝트안에서 부품으로 사용 uglify-js : 가독성을 위한 줄바꿈이 다 없어진다. 이것들도 데이터들이기 때문에 기계가 처리하기 위한 필수적인 공백만 남기고 나머지는 제거하는 경제적인 프로그램 npm install [이름] -g : 어떤 npm을 독립적으로 전역적으로 설치한다. 부품으로 사용하려 한다면, 옵션 제거후 install npm install [이름] --save :다른 디렉토리에 사용할 때도 의존성이 있는 패키지를 가져오는데 어려움이 없다. npm init : npm을 초기화 npm을 공부를 해두면 도움이 많이 된다. 콜백 함수 cmd창에 – node 입력하면 - cmd 창에서 바로 명령내릴 수 있는 상태로 변환 callback함수 : console.log(a.sort(b));라는 것에 b가 callback함수에 해당 (atom package 중에 – 크로스브라우징 신택스 테스트 패키지찾기) 동기 , 비동기 동기 : synchronous 비동기 : asynchronous 빨래 – 설거지 – 청소 비동기 : 빨래하는 업체와 설거지하는 업체와 순서가 중요하지 않다면, = 뭐가 우선적으로 일이 끝날 수 있는지 알 수 없다. 비동기식 : 속도가 빠르다 <순서대로> 사용자 – router – controller 사용자 – get.(‘/’) - send(‘hello home page’) (사용자가 /홈) 사용자 – get.(‘/login’) - send(‘login please’) (사용자가 /login) <http://a.com/login> -> login : path라고 한다. [http://a.com/topic?id=1](http://a.com/topic?id=1" \t "_blank) -> id가 1 이라는 결과를 출력 ?id=1 같은 것을 query string GET VS POST 웹브라우저가 상호작용하는 방식의 대표적 두 가지 어떠한 정보를 요청해서 가져오는 방식 : GET 사용자의 정보를 전송하는 방식 : POST post의 방식은 url의 데이터가 초과되면 브라우저가 데이터를 버리기 때문에 내용이 많아진다면 post를 사용해서 정보를 사용하면 된다.

**5. [프로젝트 구상]** Node.js 웹 프로그래밍 SCHOOL에서는 3개월간의 학습 이후 1개월간 팀 프로젝트 (혹은 개인 포트폴리오) 를 진행합니다. 팀 프로젝트 시기가 되어 프로젝트를 구상하면 프로젝트 기획에 시간을 들이느라 개발이 늦어지는 경우가 있습니다. 따라서 개강 전에 본인의 평소에 만들고 싶었던 프로젝트에 대해 생각하는 시간을 갖고 있습니다. 구체적으로 만들고 싶은 프로젝트가 없다면, 가장 좋아하는 웹 페이지나 서비스에 대해 서술하시면 됩니다.

|  |  |
| --- | --- |
| 프로젝트명 | 예매시스템 |
| 프로젝트 목적 | 서버와 네트워크의 통신 및 사용자가 편리하게 사용할 수 있도록 한다. |
| 프로젝트 내용 | 예매 시스템을 만든다. |
| 벤치마킹 서비스 | Cgv |
| 관련 이미지 |  |