

24 - 04 - 23

## 1. IDE

- : 통합 개발 툴
- : Java의 이클립스, Web환경의 Visual Studio Code
- : 제공하는 것
  - ① 소스개발 - 에디터
  - ② 컴파일 - 실행
  - ③ 빌드 - 라이브러리 (클래스모아들것) 관리
  - ④ 배포 - 개발한 것을 실제 서비스화

## 2. 컴파일 언어

: 인간의 언어를 기계어로 바꾸는 것

ex) `int a = 30` → `101100101`

### ① 원시 프로그램

: 인간이 이해하는 프로그램

ex) `int a = 30`

### ② 목적 프로그램

: 기계가 이해하는 프로그램

: 기계어

ex) `101100101`

종류

: Java, C, C#

## 3. Web 환경

: 개발언어는 HTML, CSS, JavaScript 가 있다

: IDE는 Visual Studio Code

## 4. 인터프리터 언어

: 목적 프로그램이 있다.

: 한줄 한줄 번역 후 실행

## 5. 네트워크

ex) 서버-클라이언트, 브라우저, URL, 도메인, http, TCP주소 등으로 이해

## 6. 서버 - 클라이언트 (Web의 구조)

### ① 서버

- : 웹 서비스 제공 (소스파일 제공)
- : 요청처리 → 응답하면 화면전송, 프로그램 - Live Server
- : View 만드는 입장

### ② 클라이언트

- : 웹 서비스 이용, 요청 (URL 입력)
- : 프로그램 - 브라우저 (ex. chrome)
- : View 보는 입장

## 7. Web 실행 절차

브라우저 실행 → 주소창 HTTP://www.naver.com → 서버응답 → 브라우저에 화면이 나온다  
프로토콜      도메인  
                      URL

순서 ① 클라이언트의 요청 = request

순서 ② 서버의 응답 = response

\* 웹은 클라이언트부터 시작.

## 8. View

: 클라이언트 요청에 따라 서버가 보여줄 화면을 전송

① 서버에서 응답할 소스파일

② 클라이언트에서 래더링한 화면

## 9. Ping

: 도메인 살아있는지 test

ex) Ping www.google.com

→ 구글의 IP주소가 뜸다 (여러개 IP를 사용하는 이유: 로드밸런싱)

## 10. 본인 IP주소 확인하는 법

: CMD → ipconfig

\* IP를 막는다: 컴퓨터 사용 못하게 (네트워크차단)

포트번호를 막는다: 프로그램 차단.

## 11. Html

### ① 언어

: Document Object Model - DOM 구조

html문서      객체

② 태그 (객체) (Element)

<html> [ <head> [ <meta>  
<meta> ] ] ] <title> 내용  
<body> : 문서에 보여지는 부분

## Q1 서버 프로그램

- : 클라이언트 요청을 처리하여 응답해주는 프로그램.
- : Live Server

## Q2 클라이언트 프로그램

- : 서버가 응답한 html 문서를 [揽다임] 하는 프로그램. = 컴파일
- : Chrome 브라우저

## Q3 프로토콜

- : 통신규약. (서버와 브라우저 사이 문서를 전송하기 위해 사용되는 약속)
  - : 요청하는 서비스를 의미함
  - : http (hypertext transfer Protocol)
- 추가) ftp 프로토콜

## Q4. URL

- : 요청하는 전체 주소
- : 프로토콜 :// 타겟서비스주소 : 포트번호 / 요청내용

## Q5. 도메인

- : 컴퓨터주소
- : 도메인에 해당하는 IP주소가 매핑되어있음

## Q6. 포트번호

- : 컴퓨터에 설치된 프로그램의 통신하기 위한 번호.
- : IP or 도메인 : 번호

## Q7. 서버가 정의하는 View

- : 클라이언트에게 응답하기 위한 html 문서
- : Visual Studio code 중 01.html 에디터 흐름 ↑  
사용자와 서버와의 연결

## Q8. 클라이언트가 보는 View

- : 서버가 응답한 html 문서를 브라우저에 랩다임 한 환경.

## Q9. 웹서버가 응답한 것은

- : 클라이언트의 요청에 따라 처리 후 응답한 html 소스 파일.

## Q10. 01.html 파일은 누구에게 있는가

- : 서버에 있다

Q11. IP주소

: 서버 컴퓨터의 주소

Q12. 사용하는 IDE

: Visual Studio Code

Q13. html

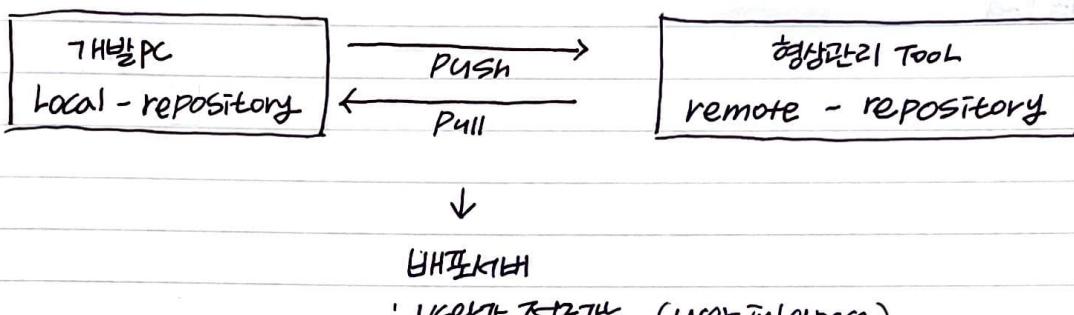
: 인터프리터 언어

: 서버가 클라이언트에게 응답하는 문서

## 1. Github

: 협상관리 Tool

: 개발 과정의 모든 단계 있어 추적 가능하다.



## 2. html

① 웹문서의 기본은 html 형식이다.

html 문서 안에 CSS, Java script가 있다.

② html 문서는 <태그>를 봐야한다.

<태그>는 부모, 자식, 자손이 있다. 부모태그를 제외하면 자녀태그도 설명된다.

부모태그 = parentNode . 자녀태그 = childNode

③ 태그의 특징을 정확히 알고 속성을 설명할 수 있어야 한다.

ex) 속성 - 언어는 영어이다 → <html lang = "en">

④ 절대문서 : html , CSS

한번 만들어지면 회현이 변하지 않는다.

동적문서 : Java script

사이트를 보고 절대문서와 동적문서를 구분할 수 있어야 한다.

• <Style> : CSS를 정의하는 태그

선택자 정의 후 스타일 설정 ☆

• 선택자 : 태그 안 Element를 선택하는 것

방법 ① <태그> 이름

② id 이름 → #

③ class 이름 → .

④ id의 span

• display

① inline : 가로세로 지정불가, 엘리먼트가 차지하는 크기만큼 인식, 줄바꿈X

② inline-block : 가로세로 지정 가능, 줄바꿈X

③ block : 가로세로 지정 가능, 줄바꿈 가능

• id : 속성, "이름"은 중복불가

• class : 속성, "이름" 중복가능

## 1. <table>

; 행과 열로 구성

<th>	<td>	<td>
<tr>		
<tr>		
<tr>		

<table>  
<tr> <td></td><td><td></td></td></td> </tr>  
→ <tr> " "  
<tr> " "  
<tr> "

테이블 험치기 ① 행 험치기 → colspan 사용

ex) <td colspan="3" = "개수" > </td>

② 열 험치기 → rowspan 사용

## 2. <form>

; 범위안에 있는 변수를 서버에 request 보낸다. (DIKTE : get)

; 입력단위는 명역이다.

**변수** DB에 넣겨주기 위해 변수에 값 저장.

: input 등의 name = " " 속성으로 변수 지정.

: 클라이언트가 입력한 정보가 값이 된다.

**종류**

① radio

: 택 1 성격의 Element , 변수명이 같아야 택 1 가능

② checkbox

: 다중선택 . 변수명이 하나라면 서버는 배열의 값을 받는다.

③ submit

: 서버에게 입력된 값 요청

: action에 지정된 값이 요청값이다.

④ reset

: 초기화 (<form> 안의 요소 제거)

input 사용 예시

→ <input type = "종류" value = "이름" >

### 3. CSS 분리.

#### ① 방법

: New Folder → New File → CSS 부분 담기.

→ HTML 문서 CSS 영역에 <link>

→ <link rel = "stylesheet" href = ". /css / 01.css "

#### ② 부트스트랩

<header>

<nav>

<div>

각 영역에 부트스트랩에서 디자인 지정 ↗ HTML 문서 복사, 붙여넣기

### 4. 시멘틱 태그

: 검색엔진에서 검색 시 시멘틱 태그로 분석하기 때문에 노출이 된다.

: <div id = "header" > 와 < header > 은 비슷하다.

#### ① <header>

: 홈페이지의 이름 / 정보

#### ② <nav>

: 홈페이지의 menu

#### ③ <Section>

: 홈페이지의 주요 내용

#### ④ <aside>

: 홈페이지의 사이드 내용

: 광고

#### ⑤ <footer>

: 홈페이지의 개발정보 / 저작권

## 1. JavaScript

- : 정적인 문서 (html, css)를 동적인 문서 (JavaScript)로 구현하자.  
→ 정적인 문서와 동적인 문서를 구분하자.
- : 동적인 문서는 화면의 일부분만 바꿀 수 있다.  
→ 정적인 문서는 화면 전체를 바꿔야 한다. 일부분만 바꾸고 싶어도.

## 2. 문법

### ① 선언문

`var a = 2`

`var b = '1234'`

→ 자료형 구분없이 var 입력. 선언된 값에 따라 자료형이 결정된다.

### ② 출력문

`console.log(a)`

→ 2

### ③ 연산자

= : 대입연산자

== : 자료형 상관없이 값이 같다

==== : 자료형과 값 모두 같다.

### ④ {}

: 블록

### ⑤ html과 JavaScript

: html 내부의 변수는 JavaScript가 사용할 수 없다.

JavaScript에 사용하려면 예를 들어 `document.getElementById('Id이동').innerHTML`을 해야 한다.

### ⑥ []

: 자료형 상관없이 모두 한 배열에 저장이 가능하다.

### ⑦ Array()

: 기본값이 배열. 마찬가지로 자료형 상관없이 저장 가능.

### ⑧ ⑨ 배열 뒤에 값 저장 → 배열명.push(값)

앞에 값 저장 → 배열명.unshift(값)

뒤에 값 삭제 → 배열명.pop()

특정 인덱스번호에 값 저장 → 배열명[번호] = 값

## 1. URI, URL

HTTP:// IP주소 : 포트번호 / 파일자 (요청페이지) ? 파라미터 (전달값)  
① URL 서버에 요청 서버에게 전달  
② URI

## 2. 협상관리

- ① 프로젝트 소스관리 (git에 소스업로드하기)
- ② 프로젝트 버전관리 (다른버전 사용예시: 기능 업그레이드, 블랙프라이데이 이벤트 등)
- ③ 역주석기능, 도록하기 기능
- ④ 팀별 개발기능

## 3. 디버깅

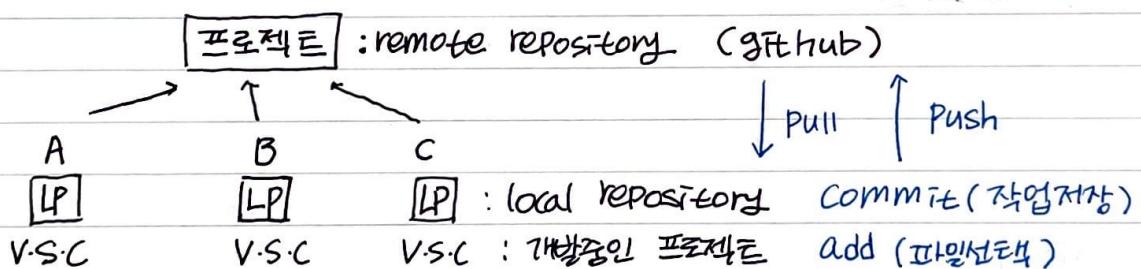
: 고친다

## 4. git

: 협상관리를

## 5. github

: 원격 저장소, 원격 repository



## 6. 스레이징

: add와 commit

## 7. 과정

- ① 적당한 위치에 JSGame 폴더 생성
- ② visual studio code에서 JSGame 폴더 오픈 (이제 JSGame 폴더로 하위에는 아무 파일도 없다.)
- ③ visual studio code 프로그램의 터미널 실행
- ④ git local repository를 만든다 (git init)
- ⑤ git 이름, 이메일 등록 (git config -- global user.name ~)
- ⑥ 원격 repository 등록 (git commit, git push origin master)
- ⑦ local git에서 원격 repository 실행