WEB STUDY #2 과제



|  |  |
| --- | --- |
| 학과 | ICT 공학계열 |
| 학번 | 202221130 |
| 이름 | 이신영 |
| 제출일 | 2022. 04. 10 |

Index

• 실습 #1

• 실습 #2

• 실습 #3

• 실습 #4

• css를 사용하여, 회원가입 폼을 퀄리티있게 최대한 꾸며오기

• Javascript 내장 함수 10가지 이상 조사 및 실습

• Server-Side와 Client-Side 조사

• Javascript를 이용한 계산기 만들기

• Javascript를 이용한 디지털 시계 만들기

• 실습 #1

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

|  |
| --- |
| <html>  <head>  <title> 실습 #1 </title>    ***<!-- style 태그 방식 -->***  <style>  table {  border-radius: 16px;  border: 2px solid #0000FF;  }  td {  border-radius: 16px;  border: 5px solid #81F7F3;  }  </style>  </head>    <body>  ***<!-- inline 방식 -->***  <table **border**=**"1"** **width**=**"500px"** **height**=**"100px"** **align**=**"center"** **name**=**"mytable"**>  <tr>  <td **style**=**"color:white; font-weight:bold"colspan**=**"3"** **align**=**"center"** **bgcolor**=**"#FA58F4"**>  2022 CAT-Security Study mentor </td>  </tr>  <tr>  <td **align**=**"center"** **bgcolor**=**"#F7D358"**> 이름 </td>  <td **align**=**"center"** **bgcolor**=**"#F7D358"**> 학과 </td>  <td **align**=**"center"** **bgcolor**=**"#F7D358"**> 성별 </td>  </tr>  <tr>  <td **align**=**"center"**> 정지용 </td>  <td **align**=**"center"**> 컴퓨터정보공학부 </td>  <td **align**=**"center"**> 남 </td>  </tr>  <tr>  <td **align**=**"center"**> 류형호 </td>  <td **align**=**"center"**> 컴퓨터정보공학부 </td>  <td **align**=**"center"**> 남 </td>  </tr>  <tr>  <td **align**=**"center"**> 박혜윤 </td>  <td **align**=**"center"**> 컴퓨터정보공학부 </td>  <td **align**=**"center"**> 여 </td>  </tr>  </table>  </body>  </html> |

* 결과테이블이(가) 표시된 사진

  자동 생성된 설명

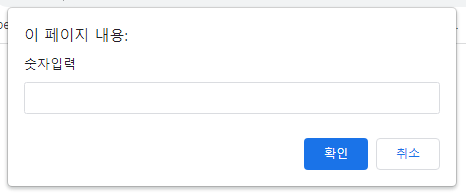
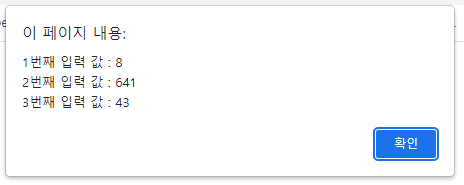
• 실습 #2

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

|  |
| --- |
| <html>  <head>  <title> 실습 #2 </title>  </head>  <body>  <p **align**=**"center"**>  <input **type**=**"button"** **onclick**=**"jstest()"** **value**=**"클릭시 실행"**/> </p>  </body>  </html>  <script>  ***function*** jstest**(){**  ***var*** a **=** prompt**(**"숫자입력"**,** ""**);**  ***var*** b **=** prompt**(**"숫자입력"**,** ""**);**  ***var*** c **=** prompt**(**"숫자입력"**,** ""**);**  alert**(**"1번째 입력 값 : " **+** a **+** "\n2번째 입력 값 : " **+** b **+** "\n3번째 입력 값 : " **+** c **);**  ***var*** a1 **=** Number**(**a**);**  ***var*** b1 **=** Number**(**b**);**  ***var*** c1 **=** Number**(**c**);**  document.write**(**"입력한 수 : " **+** a **+** " " **+** b **+** " " **+** c**);**  ***var*** d **=** a1**;**  ***if*** **(**b1 **>** d**)** d **=** b1**;**  ***if*** **(**c1 **>** d**)** d **=** c1**;**  document.write**(**"<br>가장 큰 수 : " **+** d**);**  **}**  </script> |

* 결과

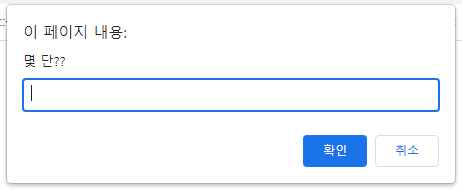


• 실습 #3

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

|  |
| --- |
| <html>  <head>  <title> 실습 #3 </title>  </head>  <body>  <p **align**=**"center"**>  <input **type**=**"button"** **onclick**=**"jstest()"** **value**=**"클릭시 실행"**/> </p>  </body>  </html>  <script>  ***function*** jstest**(){**  ***var*** a **=** prompt**(**"몇 단??"**,** ""**);**  document.write**(**'<table border="1" align="center" name="mytable" style="border: 5px solid blue">'**);**  document.write**(**'<tr>'**);**  ***for*** **(*var*** i **=** 1**;** i **<** 10**;** i**++){**  document.write**(**'<td style="border: 2px dashed red">' **+** a **+** 'X' **+** i **+** '=' **+** **(**a **\*** i**)** **+** '</td>'**);**  **}**  document.write**(**'</tr>'**);**  document.write**(**'</table>'**);**  **}**  </script> |

* 결과

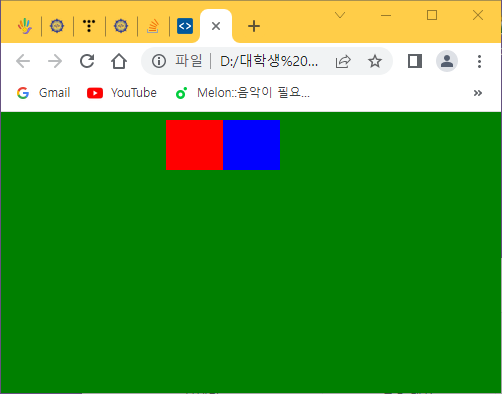
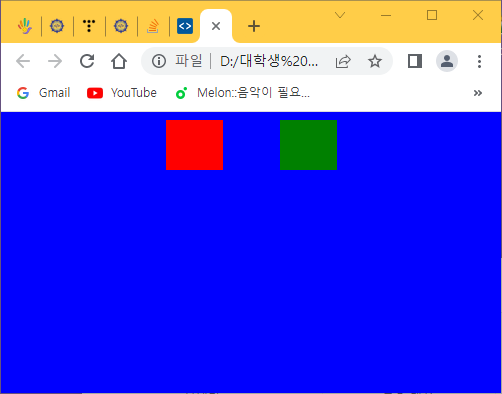
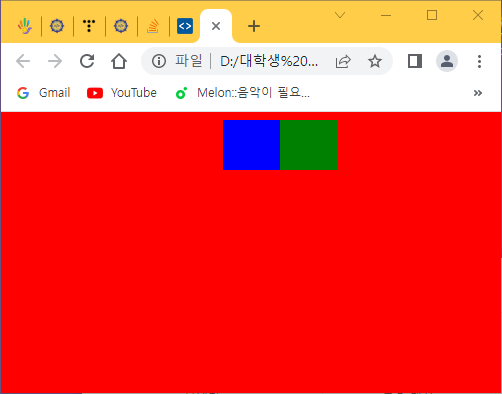
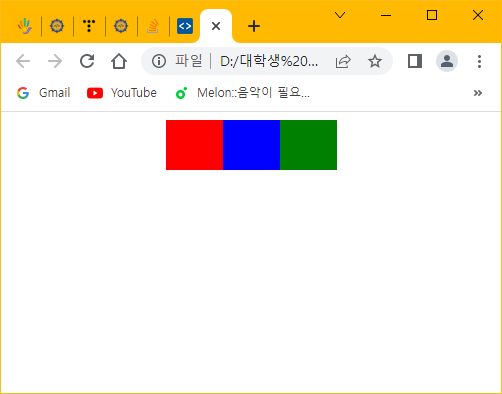
• 실습 #4

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

|  |
| --- |
| <html>  <head>  <title> 실습 #4 </title>  <style>  table{  border-collapse:collapse;  border:none;  }  td {  border:none;  width:55px;  height:50px;  }  </style>  </head>  <body **id**=**"mybgcolor"**>    <table **border**=**"1"** **align**=**"center"**>  <tr><td **bgcolor**=**"red"** **onmouseover**=**"change1()"**></td>  <td **bgcolor**=**"blue"** **onmouseover**=**"change2()"**></td>  <td **bgcolor**=**"green"** **onmouseover**=**"change3()"**></td>  </tr> </table>    <script>  ***function*** change1**()** **{**  document.getElementById**(**"mybgcolor"**).**style.backgroundColor **=** "red"**;**  **}**  ***function*** change2**()** **{**  document.getElementById**(**"mybgcolor"**).**style.backgroundColor **=** "blue"**;**  **}**  ***function*** change3**()** **{**  document.getElementById**(**"mybgcolor"**).**style.backgroundColor **=** "green"**;**  **}**  </script>  </body>  </html> |

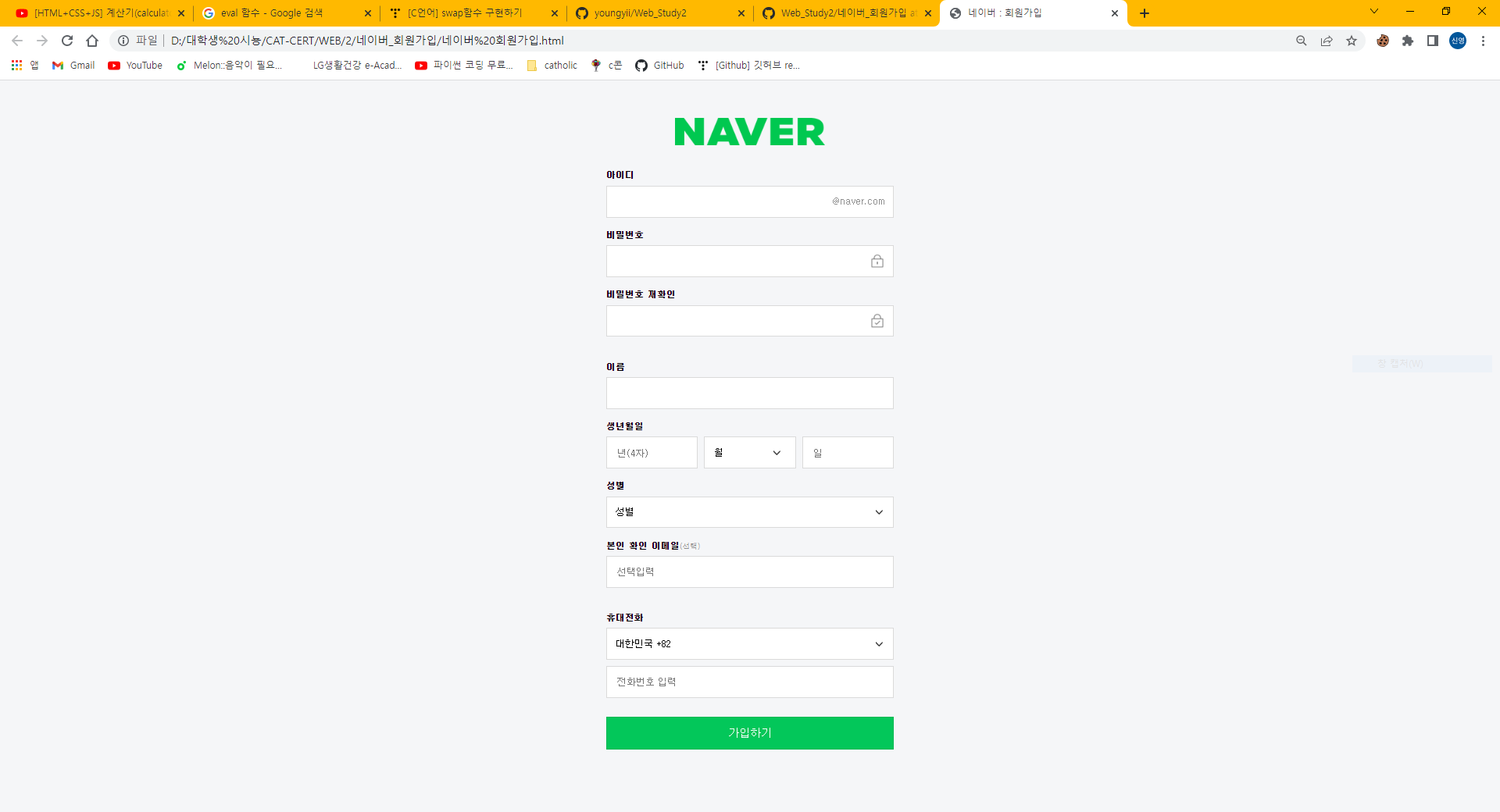
* 결과



• css를 사용하여, 회원가입 폼을 퀄리티있게 최대한 꾸며오기

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

* 결과

• Javascript 내장 함수 10가지 이상 조사 및 실습

* eval() 함수

|  |  |
| --- | --- |
| 내용 | \* 문자열로 표현된 자바스크립트 코드를 실행. |
| 실습 코드 | <script>  document.write**(**eval**(**"10 + 10 - 6 / 2 \* 2"**));**  </script> |
| 실습 결과 |  |

* parseInt(string, radix) 함수

|  |  |
| --- | --- |
| 내용 | \* 문자형 데이터를 정수형 데이터로 변환.  \* radix: string 문자열을 읽을 진법 (2~36의 수)  \* string의 첫 글자를 정수로 변경할 수 없으면 NaN(Not a Number) 값을 리턴함.  \* 0X, 0x로 시작하면 16진법으로 처리함. |
| 실습 코드 | <script>  document.write**(**parseInt**(**"10"**)** **+** ' '**);**  document.write**(**parseInt**(**"-9"**)** **+** ' '**);**  document.write**(**parseInt**(**"15문자열"**)** **+** ' '**);**  document.write**(**parseInt**(**"문자열15"**)** **+** ' '**)**  document.write**(**parseInt**(**"10"**,** 2**)** **+** ' '**);** ***// "10"을 2진수로 읽어서 10진수로 반환한 값 리턴***  document.write**(**parseInt**(**"0XF"**));** ***// "0XF"를 16진수로 읽어서 10진수로 반환한 값 리턴***  </script> |
| 실습 결과 |  |

* parseFloat() 함수

|  |  |
| --- | --- |
| 내용 | \* 문자형 데이터를 실수형 데이터로 변환  \* 양의 부호(+), 음의 부호(-), 숫자, 소수점, 지수(e, E) 외의 다른 글자를 발견할 경우 해당 문자 이전까지의 문자만 사용해 파싱함.  \* 공백으로 시작하면 공백은 무시함. |
| 실습 코드 | <script>  document.write**(**parseFloat**(**"10.123"**)** **+** ' '**);**  document.write**(**parseFloat**(**"-9.765"**)** **+** ' '**);**  document.write**(**parseFloat**(**"15.35문자열"**)** **+** ' '**);**  document.write**(**parseFloat**(**"문자열3.14"**)** **+** ' '**)**  document.write**(**parseFloat**(**"4e+2"**));**  </script> |
| 실습 결과 |  |

* isNaN() 함수

|  |  |
| --- | --- |
| 내용 | \* 전달된 값이 NaN인지의 여부를 boolean으로 반환.  \* NaN이면 true, NaN이 아니면 false를 반환. |
| 실습 코드 | <script>  document.write**(**isNaN**(**NaN**)** **+** ' '**);**  document.write**(**isNaN**(**"asdasd"**)** **+** ' '**);**  document.write**(**isNaN**(*true*)** **+** ' '**);**  document.write**(**isNaN**(*null*)** **+** ' '**);**  document.write**(**isNaN**(**37**));**  </script> |
| 실습 결과 |  |

* isFinite() 함수

|  |  |
| --- | --- |
| 내용 | \* 전달된 값이 유한한 수인지 boolean으로 반환  \* 유한한 숫자이면 true, 무한한 숫자 또는 숫자가 아니면 false를 반환. |
| 실습 코드 | <script>  document.write**(**isFinite**(**123.09**)** **+** ' '**);**  document.write**(**isFinite**(**NaN**)** **+** ' '**);**  document.write**(**isFinite**(**'young'**)** **+** ' '**);**  document.write**(**isFinite**(*null*)** **+** ' ' **);**  </script> |
| 실습 결과 |  |

* forEach() 함수

|  |  |
| --- | --- |
| 내용 | \* 주어진 함수를 배열 요소 각각에 대해 실행. |
| 실습 코드 | <script>  ***const*** arr **=** **[**1**,** 2**,** 3**,** 4**,** 5**];**  arr.forEach**(*function*(**s**)** **{**  document.write**(**s **+** ' '**);**  **});**  </script> |
| 실습 결과 |  |

* indexOf() 함수

|  |  |
| --- | --- |
| 내용 | \* 배열의 요소 값을 indexOf() 함수의 인자로 넘겨주면 해당하는 값이 몇 번째 인덱스인지 알려줌. |
| 실습 코드 | <script>  ***const*** arr **=** **[**'고양이'**,** '개'**,** '사자'**,** '이신영'**,** '거북이'**];**  document.write**(**arr.indexOf**(**'이신영'**));**  </script> |
| 실습 결과 |  |

* concat() 함수

|  |  |
| --- | --- |
| 내용 | \* 여러 개의 배열을 하나의 배열로 만들어 줌.  [배열명].concat([배열명]);  \* 여러 개의 문자열을 하나의 문자열로 만들어줌.  "[문자열]".concat("[문자열]"); |
| 실습 코드 | **<**script**>**  **const** str1 **=** "Shin"**;**  **const** str2 **=** "Young"**;**  **const** arr1 **=** **[**1**,** 3**,** 5**];**  **const** arr2 **=** **[**2**,** 4**,** 6**];**    **document.**write**(**str1**.**concat**(**str2**)** **+** "<br>" **+** arr1**.**concat**(**arr2**));**  **</**script**>** |
| 실습 결과 |  |

* escape() 함수

|  |  |
| --- | --- |
| 내용 | \* 영문 알파벳, 숫자, 일부 특수문자(@ , \* , - , \_ , . , /) 를 제외한 모든 문자를 인코딩함. |
| 실습 코드 | <script>  ***const*** a **=** ";,/?:@&=+$"**;**  ***const*** b **=** "ABC안녕123"**;**  document.write**(**escape**(**a**)** **+** "<br>"**);**  document.write**(**escape**(**b**));**  </script> |
| 실습 결과 |  |

* unescape() 함수

|  |  |
| --- | --- |
| 내용 | \* escape() 로 인코딩된 문자를 디코딩함 |
| 실습 코드 | **<**script**>**  **const** a **=** ";,/?:@&=+$"**;**  **const** b **=** "ABC안녕123"**;**  **document.**write**(unescape(escape(**a**))** **+** "<br>"**);**  **document.**write**(unescape(escape(**b**)));**  **</**script**>** |
| 실습 결과 |  |

• Server-Side와 Client-Side 조사

* **서버-클라이언트(server-client) 구조**

데이터를 저장하고 관리하는 서버 부분과 해당 서버에 접속하여 데이터를 열람하는 클라이언트 부분으로 구성된 네트워크 구조

* **Server-Side와 Client-Side**

|  |  |  |
| --- | --- | --- |
|  | **Server-Side** | **Client-Side** |
| 내용 | \* 네트워크의 한 방식인 서버-클라이언트 구조의 서버 쪽에서 행해지는 처리 | \* 네트워크의 한 방식인 서버-클라이언트 구조의 클라이언트 쪽에서 행해지는 처리 |
| 예시 | \* 브라우저에서 요청한 HTML 문서가 PHP 등의 서버 사이드 스크립트 언어를 포함하고 있으면 서버 쪽에서 이 부분을 처리하여 결과를 브라우저에 송신함.  \* MMORPG(대규모 다중 사용자 온라인 롤플레잉 게임)에서 클라이언트-서버 구조가 사용됨. 대부분의 게임에서는 게임 캐릭터 정보와 게임 아이템 정보의 위조를 방지하기 위해 이를 서버 사이드로 처리함. | \* HTML 문서가 동적인 부분을 갖고 있지 않다면 문서 수신이 끝나고부터는 서버와 교신하지 않고 브라우저가 클라이언트 사이드에서 처리하여 화면에 내용을 표시함.  \* MMORPG(대규모 다중 사용자 온라인 롤플레잉 게임)에서 클라이언트-서버 구조가 사용됨. 대부분의 MMORPG는 화려한 그래픽 효과를 사용하는데 이를 위해서는 많은 연산이 필요하며 이러한 연산을 서버 쪽에서 모두 부담할 수 없으므로 그래픽 처리나 소리 처리의 대부분을 클라이언트 사이드로 처리함. |
| 장점 | \* 서버 관리자의 입장에서, 데이터 위조의 가능성을 줄일 수 있음. 서버 쪽의 데이터가 확실한 진위이며 클라이언트 쪽에서 위조해서는 안 되는 민감한 데이터의 경우 서버 사이드로 처리해야 함. (ex/ 인터넷 뱅킹의 이체 관련 처리나 MMORPG의 게임 아이템 관련 처리에서는 클라이언트 사이드 처리를 최소화해야 함.)  \* 클라이언트 사용자의 입장에서, 클라이언트 컴퓨터의 처리 부담이 줄어듦. | \* 서버 관리자의 입장에서, 서버의 처리 부담을 줄여서 결과적으로 서버 비용을 줄일 수 있음.  \* 처리하는 데이터가 보안에 민감한 경우, 클라이언트 내에서 처리가 가능한 부분에 대해서는 통신에 대비하여 암호화할 필요가 없으므로 암호화 소요가 줄어든다. |
| 단점 | \* 서버 관리자의 입장에서, 서버의 처리 부담이 커져 결과적으로 서버 비용이 늘어날 수 있음. | \* 서버 관리자의 입장에서, 클라이언트 사이드에서 처리한 결과를 되받아야 하는 경우, 결과의 진위성을 알기 어려움. 클라이언트 쪽에서 데이터를 위조하기 쉬움.  \* 클라이언트 사용자의 입장에서, 클라이언트 컴퓨터의 처리 부담이 많아짐. |

• Javascript를 이용한 계산기 만들기

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

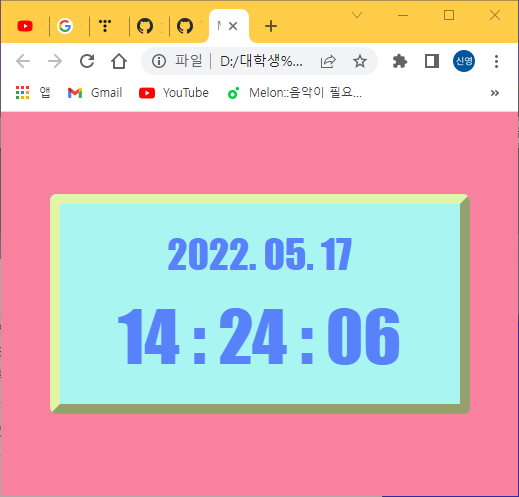
* 텍스트, 전자기기이(가) 표시된 사진

  자동 생성된 설명결과

• Javascript를 이용한 디지털 시계 만들기

* 소스 코드

<https://github.com/youngyii/Web_Study2.git> 에 업로드 하였음.

* 결과