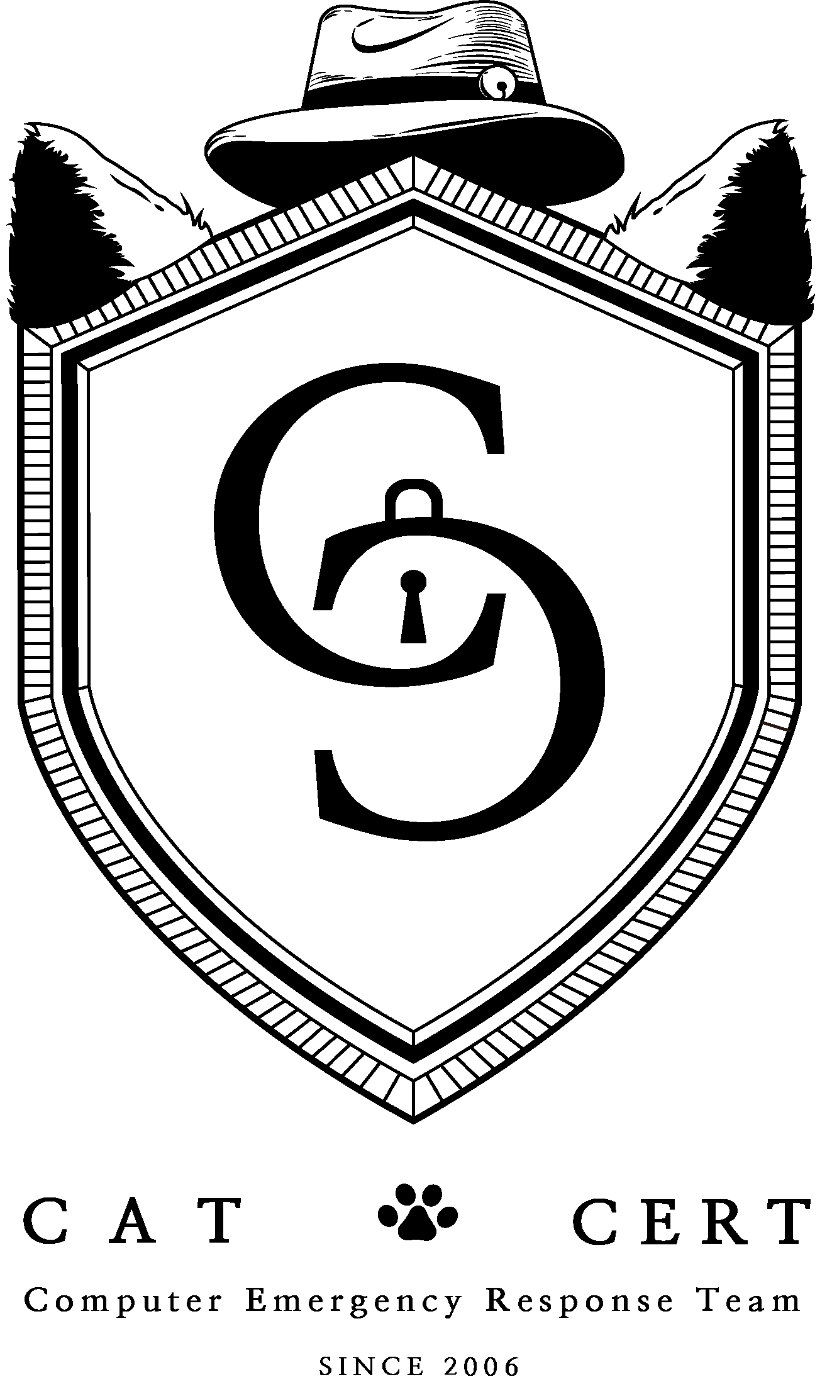
**캣서트 보고서**



202221649 인공지능학과 김민영

Web\_Study3

3.

**실습 #1**

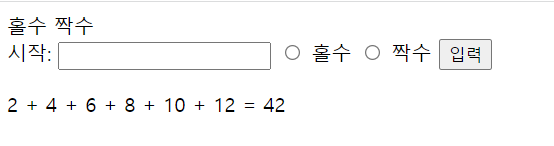
|  |
| --- |
| <html>  <head>  </head>  <body  <form method="post" action=""> <!post 방식을 이용해서 gugu 값 받기>  입력: <input type="text" name="GUGU"/>  <input type="submit" value="전송"/>  </form>  </body>  </html>  <?php  $i=0;  for ($i=1; $i<=9; $i++){  if(isset($\_POST['GUGU'])){ //isset함수를 이용 / gugu 값을 받으면 구구단 실행  echo $\_POST['GUGU']. "\*" . $i . "=". $\_POST['GUGU']\*$i. "<br>";  }  }  ?> |

테이블이(가) 표시된 사진

자동 생성된 설명

**실습 #2**

|  |
| --- |
| <html>  <head> 홀수 짝수  </head>  <body>  <form method="post" action="">  시작: <input type="text" name="NUM"/>  <input type="radio" name="iradio" value="radio1"> 홀수  <input type="radio" name="iradio" value="radio2"/> 짝수  <input type="submit" value="입력"/>  </form>  <form method="post" action="">  시작: <input type="text" name="NUM"/>  <input type="radio" name="iradio" value="radio1"> 홀수  <input type="radio" name="iradio" value="radio2"/> 짝수  <input type="submit" value="입력"/>  </form>  <body>  </html>  <?php  $i=0;  $even=0;  $odd=0;  if ($\_POST['iradio']=='radio1'){  for ($i=1; $i<=$\_POST['NUM']-2; $i++){  if($i%2==1){  $even += $i;  echo $i." + ";  }  }  $even+=$\_POST['NUM'];  echo $\_POST['NUM']." = ".$even;  }  if ($\_POST['iradio']=='radio2'){  for ($i=1; $i<=$\_POST['NUM']-2; $i++){  if($i%2==0){  $odd += $i;  echo $i." + ";  }  }  $odd+=$\_POST['NUM'];  echo $\_POST['NUM']." = ".$odd;  } |



**숫자야구**

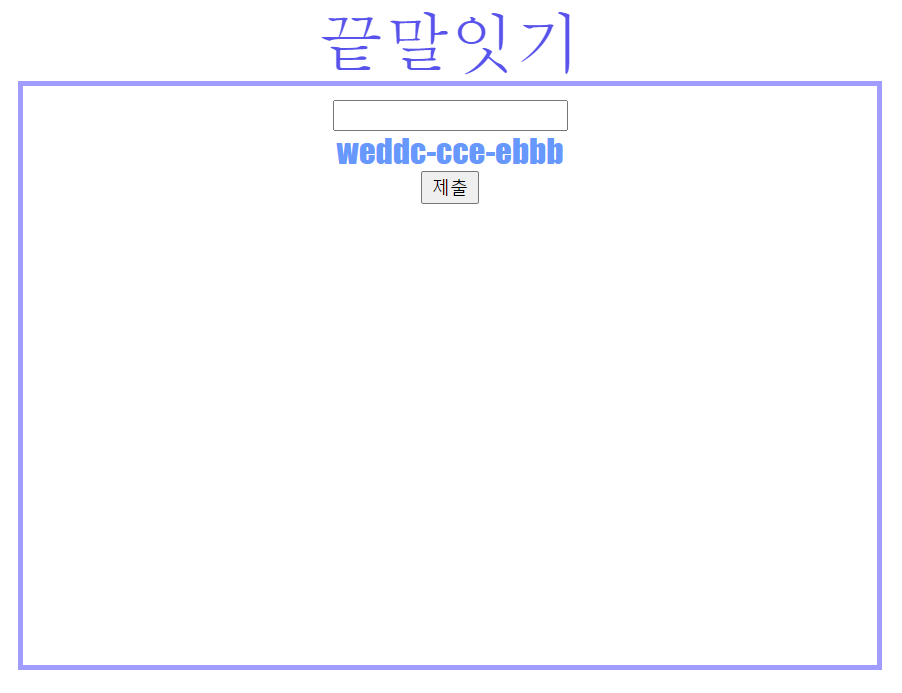
|  |
| --- |
| <?php  $word=$\_POST['start'];  $count=0;  $strike=0;  $ball=0;  $out=0;  $z=0;  if (empty($word)){  $num=rand(100,999); // rand 함수를 이용해서 난수 생성  $before=$\_POST['before']; //hidden 을 이용해 저장해둔 before값을 다시 저장  $before=$num; //before 에 난수 저장  $try=0;  }  else {  $before=$\_POST['before'];  for ($count=0; $count<3; $count++){ //3글자니깐 반복문 3번 반복  if ($word[$count]==$before[$count]){  $strike++; //글자와 자리 다 겹칠 때  continue;  }  else if ($word[$count]!=$before[$count]){  for ($z=0; $z<3; $z++){  if ($word[$count]==$before[$z]){  $ball++; // 글자는 겹치는데 자리가 안 겹칠 때  break;  }  if ($word[$count]!=$before[$z]&& $z==2){  $out++; // 글자와 자리 모두 안 겹칠 때  }  }  }  }  }  echo "<center><font size='10' color='#5853EB' face='Impact'>".'Base Ball';  ?>  <html>  <head> <title> 숫자야구 </title></head>  <body>  <div style="border:solid 4px #A19CFF; width:400px ; height:250px ; padding:10px;">  <div>  <center>  <?php  if (isset($word)){  $try=$\_POST['try'];  $try++;  if (($try%10==0) && ($try!=0)){ //시도 횟수가 10번이 되면 정답 출력  echo "<font size='2' color='474747'>"."정답은".$before."입니다."."<br>";  }  if(($try%10==1) && ($try!=1)){ // 시도 횟수가 10이 넘어가면 0으로 초기화  $try=1;  $num=rand(100,999);  $before=$\_POST['before'];  $before=$num;  }  }  echo"<font size='6' color='#FFE400' face='Comic Sans MS'>".$strike." "."<b>"."Strike"."<br>";  echo"<font size='6' color='#6ABC64' face='Comic Sans MS'>".$ball." "."<b>"."ball"."<br>";  echo"<font size='6' color='#FF6C6C' face='Comic Sans MS'>".$out." "."<b>"."Out"."<br>"."<br>";  echo "<font size='4' color='474747'>"."시도횟수". $try ;  ?>  <form method="POST" action="">  <input type="text" name="start">  <input type="hidden" name="before" value="<?=$before;?>" /> //hidden을 이용해서 난수 저장  <input type="hidden" name="try" value="<?=$try;?>" /> // hidden을 이용해서 시도값 저장  <input type="submit">  </form>  </center>  </body>  </html> |

텍스트이(가) 표시된 사진

자동 생성된 설명

**끝말잇기**

|  |
| --- |
| <?php  $input = $\_POST['start']; //가장 최근에 입력한 단어  $before = $\_POST['before']; //hidden에 저장해둔 before값 가져옴  $last=substr($before,-1,1); // before의 마지막 글자수를 가져옴  $first=substr($input,0,1); // input의 첫글자를 가져옴  echo "<center><font size='10' color='#5853EB' face='serif'>"."끝말잇기";  ?>  <html>  <head> <title> 끝말잇기 </title>  </head>  <body>  <div style="border:solid 4px #A19CFF; width:600px ; height:400px ; padding:10px;">  <div>  <center>  <form method="POST" action="">  <input type="text" name="start">  <?php  if (empty($before)){  $before=$input;  echo "<center><font size='5' color='#6798FD' face='fantasy'>".$before;  }  else {  if ($last==$first){ //끝말잇기 조건이 맞았을 때  $before =$before."-".$input; // 이전에 입력한 값들과 최근에 저장한 값이 연달아서 저장되게 함  echo "<center><font size='5' color='#6798FD' face='fantasy'>".$before;  }  else if ($last!=$first) { //끝말잇기 조건이 틀렸을 때  echo "<script>alert('실패');</script>";  $before="";  }  }  ?>  <input type="hidden" name="before" value="<?=$before; ?>"/>  <div>  <input type="submit"/>  </div>  <div id="result">  </div>  </form>  </center>  </div>  </div>  </body>  </html> |

****

**완성된 회원가입 폼에서 보낸 정보 출력**

|  |
| --- |
| <html>  <head>  <title> HTML TAG</title>  <style>  table {  width: 40%; height: 50%;  border-right:none; border-left:none;  border-bottom: 5px solid #466093;  border-top: 5px solid #466093;  }  th, td {  border: 1px solid #444444;  padding: 5px;  }  tbody tr:nth-child(n+2):nth-child(-n+7) {  background-color: #C4DEFF ;  }  tbody tr:nth-child(1) {  background-color: #8EA8DB;  }  tbody tr:nth-child(8) {  background-color: #8EA8DB;  }  tbody tr:nth-child(9) {  background-color: #C4DEFF;  }  </style>  </head>  <body>  <form method="post" action="">  <table border="2" width="700px" height="300px">  <tr>  <td colspan="2" div style="text-align:center">  <span style="font-size: 18px; font-style: italic ; font-weight: bold; ">  회원가입  </span> </td>  <tr>  <td div style="text-align:center">  <span style="font-size: 15px; font-style: italic ; font-weight: bold; ">이름 </span> </td>  <td> <input type="text" name="name"/><br> </td>  <tr>  <td div style="text-align:center">  <span style="font-size: 15px; font-style: italic ; font-weight: bold; "> 성별 </span> </td>  <td> <input type="radio" name="gender" value="남"/> 남  <input type="radio" name="gender" value="여"/> 여 <br> </td>  <tr>  <td div style="text-align:center">  <span style="font-size: 15px; font-style: italic ; font-weight: bold; "> 회원 ID </span> </td>  <td> <input type="text" name="id"/><br> </td>  <tr>  <td div style="text-align:center">  <span style="font-size: 15px; font-style: italic ; font-weight: bold; "> 비밀번호 </span> </td>  <td> <input type="password" name="password"/> </td>  <tr>  <td div style="text-align:center">  <span style="font-size: 15px; font-style: italic ; font-weight: bold; "> 생일 </span> </td>  <td> <input type="date" name="birthday">  <tr>  <td div style="text-align:center">  <span style="font-size: 15px; font-style: italic ; font-weight: bold; "> 전화번호 </span> </td>  <td> <select name="phone1">  <option value="1"> 선택 </option>  <option name="phone" value="010"> 010 </option> <option name="phone" value="02"> 02 </option> <option name="phone" value="070"> 070  </select>  - <input type="text" name="phone2"/> - <input type="text" name="phone3"/> </td>  <tr>  <td colspan="2" div style="text-align:center">  <span style=" font-size: 18px; font-style: italic ; font-weight: bold; ">  자기소개  </span> </td>  <tr>  <td colspan="2"> <textarea rows="10" cols="50" name="text" placeholder='1000자 이상 입력하세요.'></textarea> </td>  <tr>  <td colspan="2"> <input type="submit"> </td>  </form>  </body>  </html>  <?php  if(isset($\_POST['name'])) {  echo " 이름: ".$\_POST['name']."<br>";  echo " 성별: ".$\_POST['gender']."<br>";  echo " 회원 아이디 : ".$\_POST['id']."<br>";  echo " 비밀번호 : ".$\_POST['password']."<br>";  echo " 생일 : ".$\_POST['birthday']."<br>";  echo " 전화번호 : ".$\_POST['phone1'].$\_POST['phone2'].$\_POST['phone3']."<br>";  echo " 자기소개 : ".$\_POST['text'];  }  ?> |

**1) Php에서 ‘와 “”의 차이점 조사**

|  |
| --- |
| <?php  $a = "Hello world";  print "double quote : $a"; // double quote : Hello world  print 'single quote : $a'; // single quote : $a |

“” – 문자열 내부에 변수를 치환해서 출력해줌

‘ - 치환할 값을 찾지 않고 문자열을 그대로 출력해줌

**2) Cookie와 Session 조사**

**Cookie** : 사용자의 컴퓨터에 저장하는 작은 기록 정보 파일

HTTP에서 클라이언트의 상태 정보를 클라이언트의 PC에 저장하였다가 필요시 정보를 참조, 재사용 할 수 있음

이름, 값, 만료일(저장 기간 설정), 경로 정보로 구성

이름 : 각각의 쿠키를 구별하는 데 사용되는 이름

값 : 쿠키의 이름과 관련된 값

유효시간 : 쿠키의 유지시간

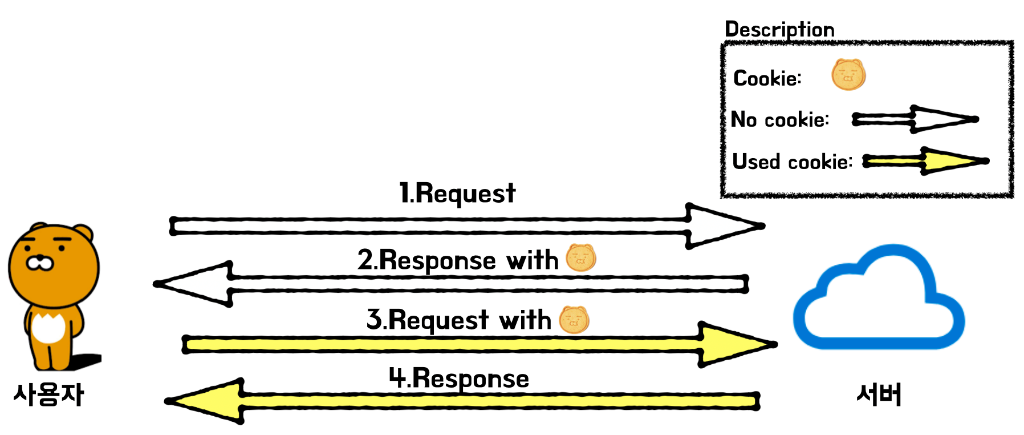
도메인 : 쿠키를 전송할 도메인

경로 : 쿠키를 전송할 요청 경로

클라이언트에 총 300개의 쿠키를 저장 가능

하나의 도메인 당 20개의 쿠키를 가질 수 있음

하나의 쿠키는 4KB(=4096byte)까지 저장 가능



1. 클라이언트가 페이지를 요청

2. 서버에서 쿠키를 생성

3. HTTP 헤더에 쿠키를 포함 시켜 응답

브라우저가 종료되어도 쿠키 만료 기간이 있다면 클라이언트에서 보관하고 있음

4. 같은 요청을 할 경우 HTTP 헤더에 쿠키를 함께 보냄

서버에서 쿠키를 읽어 이전 상태 정보를 변경 할 필요가 있을 때 쿠키를 업데이트 하여 변경된 쿠키를 HTTP 헤더에 포함시켜 응답

**Session** : 일정 시간동안 같은 사용자로부터 들어오는 일련의 요구를 하나의 상태로 보고, 그 상태를 일정하게 유지시키는 기술

웹 서버에 웹 컨테이너의 상태를 유지하기 위한 정보를 저장

웹 서버의 저장되는 쿠키(=세션 쿠키)

브라우저를 닫거나, 서버에서 세션을 삭제했을때만 삭제가 되므로, 쿠키보다 비교적 보안이 좋음.

저장 데이터에 제한 X

각 클라이언트 고유 Session ID를 부여  
Session ID로 클라이언트를 구분하여 각 클라이언트 요구에 맞는 서비스 제공

1. 클라이언트가 페이지를 요청 (사용자가 웹사이트 접근)

2. 서버는 접근한 클라이언트의 Request-Header 필드인 Cookie를 확인하여, 클라이언트가 해당 session-id를 보냈는지 확인

3. session-id가 존재하지 않는다면, 서버는 session-id를 생성해 클라이언트에게 돌려줌.

4. 서버에서 클라이언트로 돌려준 session-id를 쿠키를 사용해 서버에 저장  
쿠키 이름 : JSESSIONID

5. 클라이언트는 지 접속시, 이 쿠키(JSESSIONID)를 이용하여 session-id 값을 서버에 전달

**3) Index 파일의 역할에 대해서 조사**

Index 파일 : 홈페이지를 접속했을 때 기본으로 보여지는 파일의 파일명, 디렉토리의 대표 문서

파일이 없을 경우 – 디렉토리에 있는 파일들을 리스트 형식으로 보여줌 or forbidden 메시지가 뜨면서 접근이 금지

확장자는 Eclipse, Windows, Mac 및 Linux 운영 체제 용 통합 개발 환경과 연결되어 있음

Index 파일은 기여 된 도움말 컨텐츠에 대한 키워드 인덱스를 저장함.

→ 이런 형식의 파일은 주로 데스크톱 및 일부 모바일 장치에서 찾을 수 있음.

텍스트이(가) 표시된 사진

자동 생성된 설명