

일정관리 서비스

2020. 05. 13

도원진

1. 일정관리 웹

1.1 목적

한 주의 목표 및 일일 계획 세우고
성취도 체크를 통한 업무효율 높이기

1.2 기능

- 목표/결과(주 단위) 조회 및 등록
- 계획/결과(일 단위) 조회 및 등록
- 팀원의 일정조회

주간 업무 계획		
일자	유형	내용
05-10(일)	계획	PHP 일정관리 서비스 개발완료하기
	결과	
05-11(월)	계획	
	결과	
05-12(화)	계획	
	결과	
05-13(수)	계획	
	결과	

1. 일정관리 웹

1.3 개발 단계별 소개

1단계	2단계	3단계	4단계	5단계
write기능 구현 시 별도의 화면전환 후write 수행 조회기능	현재 화면 modal창을 통 해 write수행 Ajax처리 추가	[쓰기]버튼 제거 <div> <textarea> 전환	BootStrap SBAAdmin적용	팀원 일정 조회
				

2. read, write기능

1단계

2.1. read, write 기능 빠르게 구현하기

- <textarea>를 통해 글 보여주기
- [쓰기] 버튼을 통해 다음페이지 이동

2020
<< 05.10 ~ 05.16 >>
Index.php
wonjin@bizspring.co.kr 환영합니다.

주간 업무 목표

계획 쓰기
결과 쓰기

주간 업무 계획

일자	유형	내용	
05-10(일)	계획		쓰기
	결과		
05-11(월)	계획		쓰기
	결과		쓰기
05-12(화)	계획		쓰기
	결과		쓰기
05-13(수)	계획		쓰기

버튼클릭

페이지 이동

write.php

새 일정 등록 테스트

제출

2.2. 등록 완료 후 페이지

Index.php

일자	유형	내용		
05-10(일)	계획			쓰기
	결과	<u>새 일정 등록 테스트</u>		쓰기

2.3. 구현 시 고려사항

- **SELECT**

- Php Date함수 -> 날짜계산 후 화면에 주 7일 표기
- Js보다 url query값을 함께 다루기 편리

- **INSERT**

- DB테이블 한 row에 두 컬럼 plan, result에 대해 insert와 update진행
- 조회된 row가 없는지 select문으로 선 확인 후 insert 또는 update를 결정

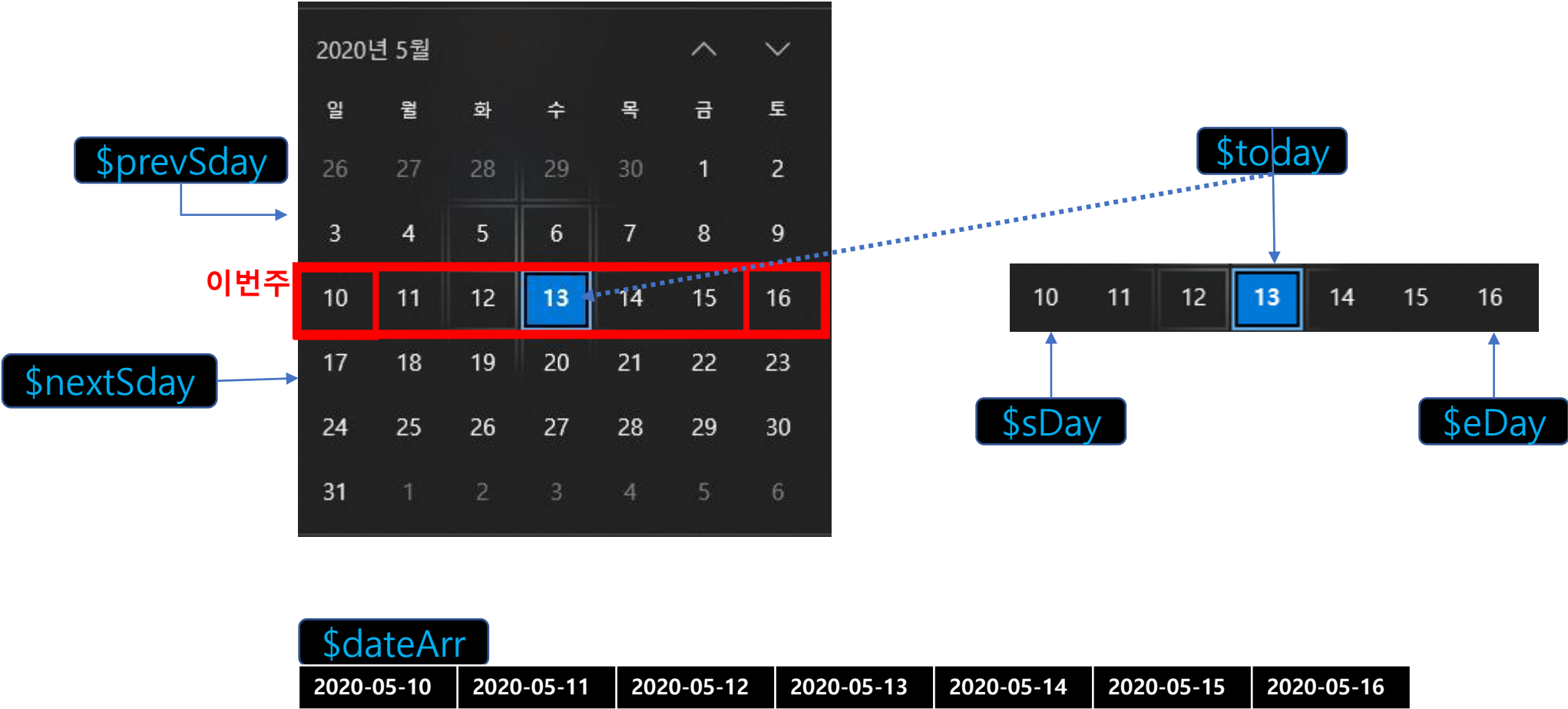
2.3. 구현 시 고려사항

[Table Entity]

Table	Type	Physical	Logical	Nullable	Key	Default	Extra
tbl_member	varchar(20)	id	아이디(이메일)	NO	pk		
tbl_schedule	int(11)	idx	식별번호	NO	pk		
tbl_schedule	varchar(20)	id	아이디(이메일)	NO	fk		tbl_member의 id
tbl_schedule	date	sdate	일자	NO			
tbl_schedule	char(1)	isDay	주간/일간	NO			"Y" 또는 "N"
tbl_schedule	varchar(100)	plan	계획				
tbl_schedule	varchar(100)	result	결과				
tbl_schedule	datetime	regidate	등록일				

2.3.1 주/ 요일 별 날짜 계산

앞 슬라이드의 로직을 통해 아래 날짜를 나타내는 여러 값을 변수에 담을 수 있다.



2.3.1 주/ 요일 별 날짜 계산

Index.php

```
$dayOftheweekArr = array("일 ", "월 ", "화 ", "수 ", "목 ", "금 ", "토 ");
$today = date('Y-m-d'); //오늘 날짜
global $sDay;
if (isset($_GET['sDay']) && !empty($_GET['sDay'])) {
    $sDay = $_GET['sDay'];
} else {
    $sDay = $today;
    $dayOftheweek = date('w', strtotime($sDay));
    $sDay = date('Y-m-d', strtotime($sDay . " -" . "{$dayOftheweek}day"));
}
$dateArr = array();
for ($i = 0; $i < 7; $i++) {
    array_push($dateArr, date('Y-m-d', strtotime($sDay . "+" . "{$i}day")));
}

$eDay = $dateArr[6];
$nextSday = date('Y-m-d', strtotime($sDay . " +" . "7day"));
$prevSday = date('Y-m-d', strtotime($sDay . " -" . "7day"));
```

해당 주에 해당하는 정보구하기

- \$sDay : 금주의 일요일
- \$eDay : 금주의 토요일
- \$dateArr : 금주의 모든 일자

2.3.2 일정 등록_insert / update

wirteProcess.php

```
if($_GET['isNew'] == 'Y'){
    echo "새로 입력하는 경우 insert <br>";
    $query = "INSERT INTO workplan_tbl
              (id,sdate,isday,{$_type},regidate)
              values(
                  '{$_GET['id']}',
                  '{$_GET['sdate']}',
                  '{$_GET['isday']}',
                  '{$_GET['text']}',
                  now())";
}else{
    //이미 있었던 경우 update
    $query = "UPDATE workplan_tbl
              SET
                  {$_type}='{$_GET['text']}',
                  regidate=now()
              WHERE isday='{$_GET['isday']}'
                  AND sdate = '{$_GET['sdate']}'
                  AND id = '{$_GET['id']}'";
```

```
mysqli_query($conn, $query);
mysqli_free_result($result);
Header("Location: ../weekPlan.php?sDay={$_GET['sDay']}");
```

선 작업

- 과거의 일정 등록 여부 기록조회

후 작업

- 등록기록 없을 경우 : insert 진행
- 등록기록 있을 경우 : update 진행

2.3.3 화면출력

Index.php

```
<?php foreach ($dateArr as $i => $date) { ?>
    <tr>
        <td class="day" id="<?php echo "day" . $i ?>" rowspan="2" data-day="<?=$date?>" ><?= date("m-d", strtotime($date)) ?>(<?= $dayOftheweekArr[$i] ?>)</td>
        <td class="center">계획</td>
        <td class="plan" id="<?php echo "plan" . $i ?>">
            <textarea cols="70" rows="5" readonly
            ><?php if ([isset($dayWorks[$date]) && !empty($dayWorks[$date])]) {
                ?><?=toHTML($dayWorks[$date]["PLAN"])?><?php } ?></textarea></td>
            <td><a href='writeView.php?SDATE=<?=$date?>&ISDAY=Y&TYPE=P&sDay=<?=$sDay?>'>쓰기</a></td>
        </tr>
```

\$dayWorks : DB에서 연관배열(key: 날짜(일~토), value: 일정)에서 가져온 데이터

1	wonjin@Bizspring.co.kr	2020-05-10	Y	일정 등록 test2		
2	wonjin@Bizspring.co.kr	2020-05-11	N	계획 등록 test2		

\$dateArr : 날짜 정보가 담긴 배열

2020-05-10	2020-05-11	2020-05-12	2020-05-13	2020-05-14	2020-05-15	2020-05-16
------------	------------	------------	------------	------------	------------	------------

구현 방법 : php배열 \$dateArr로 날짜 별 반복.

DB row정보를 갖는 \$dayWorks 키값과 일치한
날짜요소가 있을 때 \$dayWorks의 요소(일정) 출력

3. Modal창을 통한 등록

2단계

3.1. Modal창 show

index.php

Modal title

일정 등록 test2

Close Save changes

② modal

① 클릭

- [쓰기]버튼 클릭시 모달창 띄우기
- 글 작성 후 [Save changes]클릭 시 Ajax writeAPI호출

3.2. 등록 완료 후 페이지

주간 업무 계획				
일자	유형	내용		
05-10(일)	계획			쓰기
	결과	<u>일정 등록 test2</u>		쓰기

3.2. 등록버튼 클릭 시 진행과정

index.php

```
$("#submitBTN").click(function(){
    console.log("test: ",url);

    $.ajax({
        url: url,
        type: "post",
        data: {
            sdate: sdate,
            type: type,
            content: modaltextarea.val(),
            isday: isday
        }
    }).done(function(res) {
        console.log("res: ",res);
        if(res==`success`){
            content.val(modaltextarea.val());
            $("#closeBTN").click();
        }else{
            $("#closeBTN").click();
            alert("fail");
        }
    });
});
```

(요청)

Modal창 [Save change]버튼 클릭 시

Url(writeAjax.php)로 data가 함께 request에 담기고
요청진행

(응답)

writeAjax.php 에서 "success" 또는 "fail"의 문자열을
보내 성공과 실패여부만 응답

* writeAjax.php의 로직 *

writeProcess.php와 동일

4. <div>, <textarea> 사용하기

3단계

4.1. 화면구성 소개

주간 업무 계획		
일자	유형	내용
05-10(일)	계획	<div> 태그
	결과	
05-11(월)	계획	<textarea> 태그 (<div> 태그 클릭 시)
	결과	

4.1. 화면구성 소개

CSS style 및 이벤트 등록을 위한 CSS class 지정

.day	.type	.content
.day	.type	<div class="content view plan"> <textarea class="content write plan">
	.type	<div class="content view result"> <textarea class="content write result">
.day	.type	<div class="content view plan"> <textarea class="content write plan">
	.type	<div class="content view result"> <textarea class="content write result">

4.1. 화면구성 소개

[Ajax] html 텍스트 update를 위한 CSS id 지정

.day	.type	.content
#day0		<div id="plan0">
		<div id="result0">
#day1		<div id="plan1">
		<div id="result1">

4.2. 등록버튼 없는 등록과정

②

localhost 내용:
내용이
" 에서
'일정등록 테스트3' 으로 변경되었습니다. 수정하시겠습니까?
확인 취소

주간 업무 계획		
일자	유형	①<textarea>태그 내용
05-10(일)	계획	일정등록 테스트3
	결과	

<div> 태그 영역 클릭시

1. <div> display속성, "block"
2. <textarea> display속성, "none"

<textarea> 태그 영역을 빠져 나올 때(blur)

1. Confirm창
 - 취소) 이전 글로 복귀
 - 확인) Ajax로 insertAPI 호출
 - <div> display속성, "none"
 - <textarea> display속성, "block"
2. 변경사항 없을 경우
 - 이전 글로 복귀
3. <div> 태그로 복구

4.3. <div>와 <textarea>의 주의사항

<textarea> 태그

- 개행문자(\n)와 띄어쓰기 모두 잘 작성되어 DB에 모든 문자들이 insert된다.

<div> 태그

-
와 \n 모두 '줄바꿈'을 의미
 -
는 html태그로 인식하여 <div>안에서 줄 바꿈 수행
 - \n은 <div>에서 인식불가. 줄 바꿈을 할 수 없어 모든 문자가 한 줄에 출력
- HTML은 연속된 여러 번의 space를 하나의 space로 해석

<textarea>태그의 경우 정상

1. java 2. python 3. javascript jquery

<div>태그의 경우 문제발생

1. java 2. python 3. javascript jquery
--

<div>태그의 text (크롬개발자 도구)

```
<div class="view" data-sdate="2020-05-10" data-type="PLAN" data-isday="Y">1. java  
2. python  
  
3. javascript        jquery</div>
```

4.3. <div>와 <textarea>의 주의사항

따라서, 처음 화면에 접속 했을 때 보여주는 <div> 태그의 text를 가공할 것

1. " " 는 " "로 변경
2. "\n"은 "
" 로 변경

예)

"안녕하세요.\n반갑습니다."



"안녕하세요.
반 갑습니다."

4.3. <div>와 <textarea>의 주의사항

Whitespace를
태그와 특수문자인 로 변경

```
function toHTML($str)
{
    $str = str_replace(" ", "&nbsp;", $str);
    $str = str_replace("\n", "<br>", $str);
    return $str;
}
```

함수로 가공된 text로 <div>의 text를 교체

```
<?php
if (isset($dayWorks[$date]["PLAN"]) && !empty($dayWorks[$date]["PLAN"])) { ?>
    <div class="view" data-sdate="<?= $dateArr[$i] ?>" data-type="PLAN" data-isday="Y"><?= toHTML($dayWorks[$date]["PLAN"]) ?></div>
<?php } else { ?>
```

4.3. <div>와 <textarea>의 주의사항

개행과 띄어쓰기 정상출력

계획	1. java 2. python
	3. javascript jquery <div> 태그

[illegible]

<div>태그의 text (크롬개발자 도구)

5. Bootstrap SBAdmin 적용

4단계

LOGGER DEV

Search for...

Dashboard

REPORTS

Trends

업무일지

Work Diary

<< >> 2020 05.10 ~ 05.16

주간 업무 목표

계 획 DOM 자바스크립트 독서

결 과 50% 독서 완료

주간 업무 계획

일자	유형	내용
05-10(일)	계획	
	결과	
05-11(월)	계획	
	결과	
05-12(화)	계획	
	결과	
05-13(수)	계획	
	결과	
05-14(목)	계획	
	결과	
05-15(금)	계획	
	결과	
05-16(토)	계획	
	결과	

Copyright © BizSpring Inc.

클릭



wonjin@bizspring.co.kr

주간 업무 목표

계 획 DOM 자바스크립트 독서

결 과 50% 독서 완료

주간 업무 계획

일자	유형	내용
		PHP 일정관리 서비스 개발완료하기
05-10(일)	계획	
	결과	
05-11(월)	계획	
	결과	

6. 팀원의 일정 보기

5단계

6.1. 일정 조회 및 타인의 글 수정방지

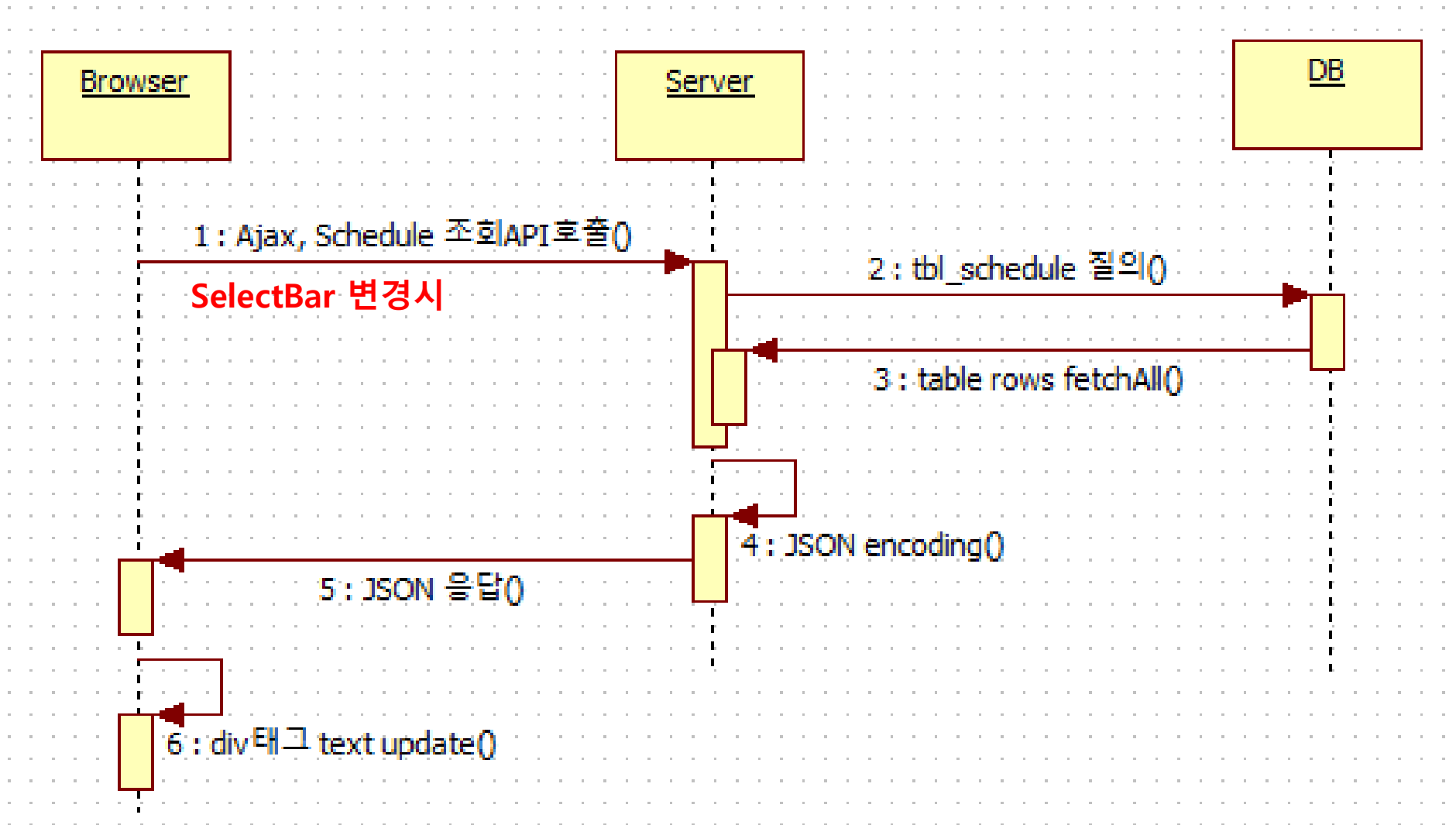
현재 계정 wonjin

hansuk 계정의 일정 조회

일정 수정 불가

Copyright © BizSpring Inc.

6.2. Sequence Diagram



6.3. JSON 결과값 (디코딩 이전)

```
{
  "weekWork": {
    "PLAN_IDX": 86,
    "ID": "hansuk@bizspring.co.kr",
    "SDATE": "2020-05-11",
    "RESULT": "100% \uc644\ub8cc",
    "ISDAY": "N",
    "PLAN": "1. DOM \uc2a4\ud06c\ub9bd\ud2b8 \ub9c8\ubb34\ub9ac\ud558\uae30",
    "REGIDATE": "2020-05-13 13:19:13"
  },
  "dayWorks": {
    "2020-05-10": {
      "PLAN_IDX": 87,
      "ID": "hansuk@bizspring.co.kr",
      "SDATE": "2020-05-10",
      "RESULT": "80% \ub3c5\uc11c\uc644\ub8cc",
      "ISDAY": "Y",
      "PLAN": "\uc2dc\uc2a4\ud15c \uc131\ub2a5\uad6c\uc870 \ub3c5\uc11c \uc644\ub8cc\ud558\uae30",
      "REGIDATE": "2020-05-13 13:19:28"
    }
  }
}
```

6.4. JSON 결과값 (디코딩 이후)

```
▼ {weekWork: {...}, dayWorks: {...}} ⓘ  
  ▼ dayWorks:  
    ▼ 2020-05-10:  
      ID: "hansuk@bizspring.co.kr"  
      ISDAY: "Y"  
      PLAN: "시스템 성능구조 독서 완료하기"  
      PLAN_IDX: 87  
      REGIDATE: "2020-05-13 13:19:28"  
      RESULT: "80% 독서완료"  
      SDATE: "2020-05-10"  
      ▶ __proto__: Object  
      ▶ __proto__: Object  
    ▼ weekWork:  
      ID: "hansuk@bizspring.co.kr"  
      ISDAY: "N"  
      PLAN: "1. DOM 스크립트 마무리하기"  
      PLAN_IDX: 86  
      REGIDATE: "2020-05-13 13:19:13"  
      RESULT: "100% 완료"  
      SDATE: "2020-05-11"
```

[workDiaryPDO.php:576](#)

7. 개발을 진행하며 배운 점

1. 효율적인 개발방법

- 필수기능
 - 빠르고 짧은 시간안에 구현
 - Model1방식의 단순 명료한 코드
- 성능고도화
 - 많은 시간을 투자하고 단계 별로 진행.
 - 성능의 고도화가 한 번에 이뤄지길 바라는 것은 금물
- PDO
 - String템플릿을 이용한 query에서 preparedStatement으로의 변화
 - SQL injection 차단기능
- Front Library
 - 개발 초기단계에 적용
 - 실제 제품을 만드는 과정처럼 전문가가 지닌 습관을 익힐 것
 - 개발을 마무리한 후 Front를 적용하기 보다 실제 제품 모습을 마주하며 개발할 것

7. 개발을 진행하며 배운 점

2 추후 과제

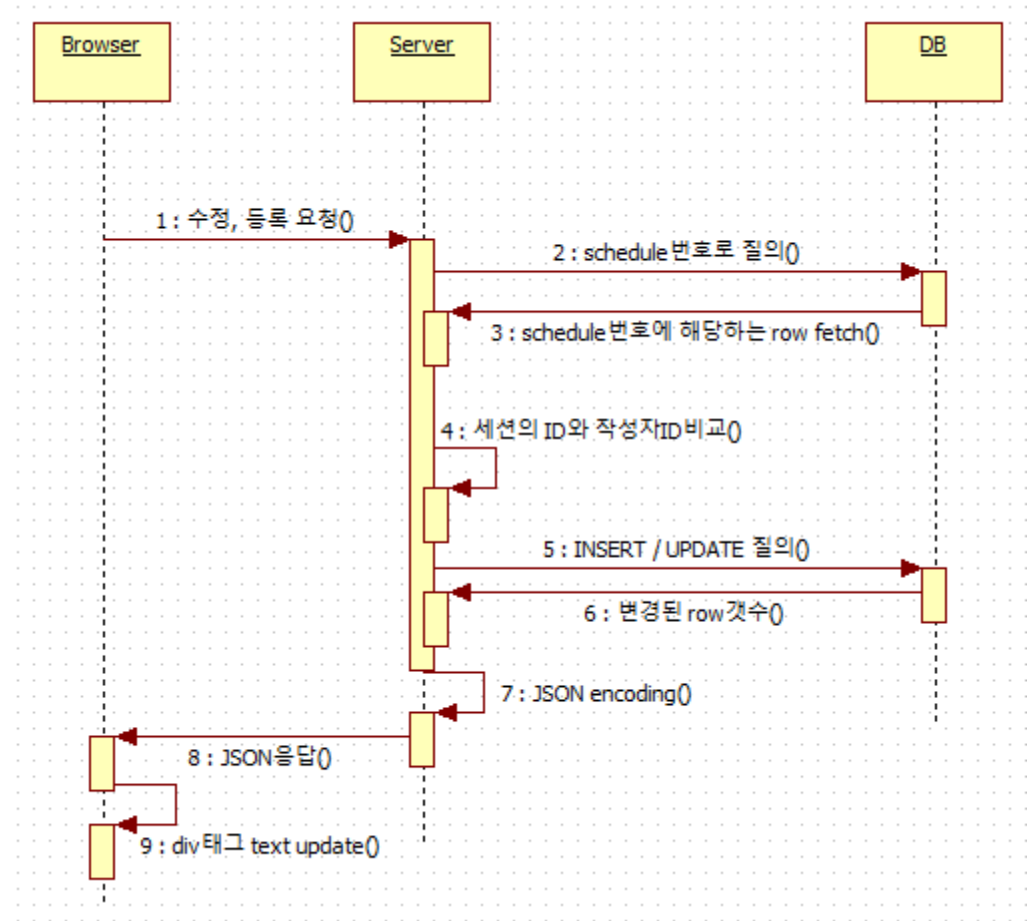
- 계정정보 구체화
- 로그인 / 회원가입 기능 추가
- Laravel 전환
- 응답JSON 형식 변경
 - 예)

```
{ status : "Error",  
  msg   : "you need login",  
  data  : $data }
```

7. 개발을 진행하며 배운 점

2 추후 과제

- Url요청 악용으로 인한 데이터 변경 방지
 - Front에서 url요청을 차단할 수 없음
 - 'schedule작성자'를 함께 서버로 전송하여도 신뢰할 수 없음
 - Schedule 고유번호를 포함한 URL에 필요한 요청 파라미터 전송
 - **SELECT**질의를 통해 신뢰할 수 있는 값(schedule작성자)을 DB에서 일정 작성자의 **id** 얻기
 - Session의 id와 비교를 통해 JSON의 status코드를 결정한다



감사합니다