

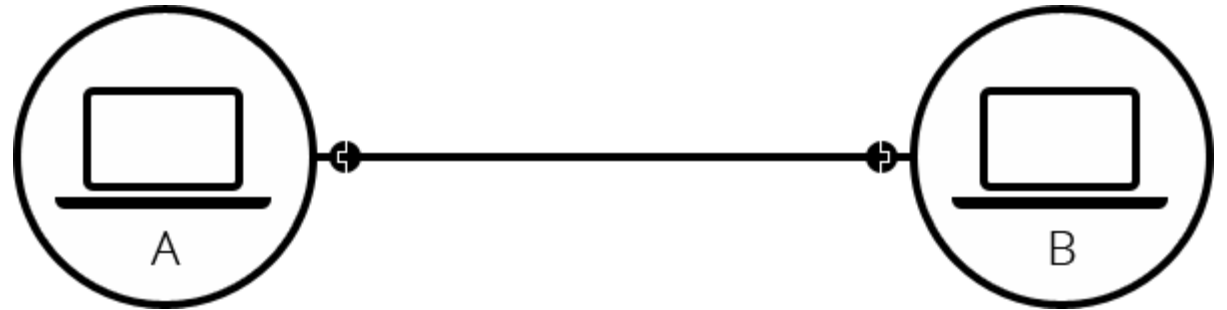
DoiT! Web Project

웹 백엔드 개발

인터넷의 동작 원리

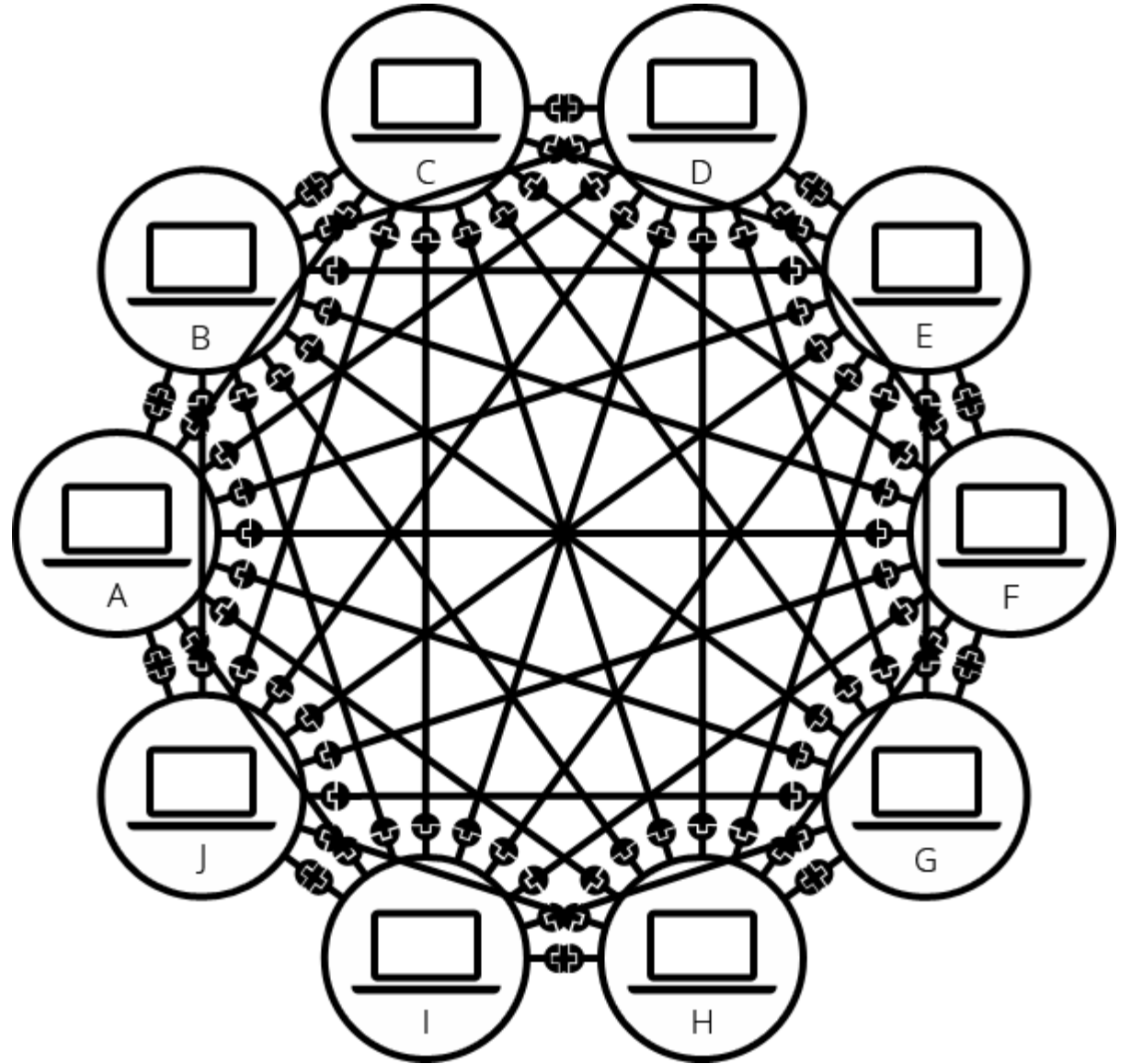
가장 기본적인 네트워크

두 컴퓨터는 하나의 케이블을 연결해
통신할 수 있다.



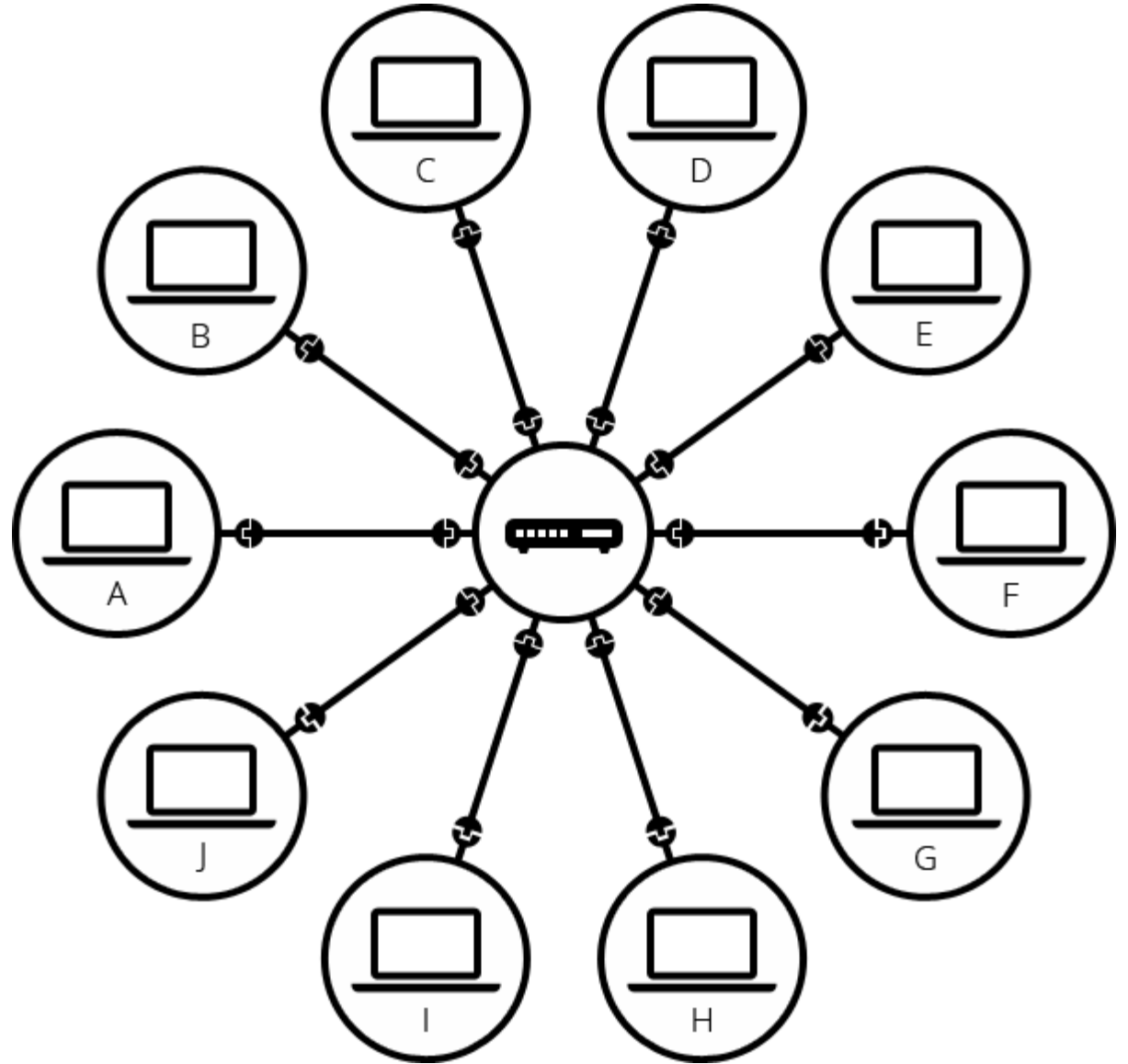
복잡한 네트워크

만약 10대의 컴퓨터가 서로 통신해야 한다면
45개의 케이블이 필요하다.



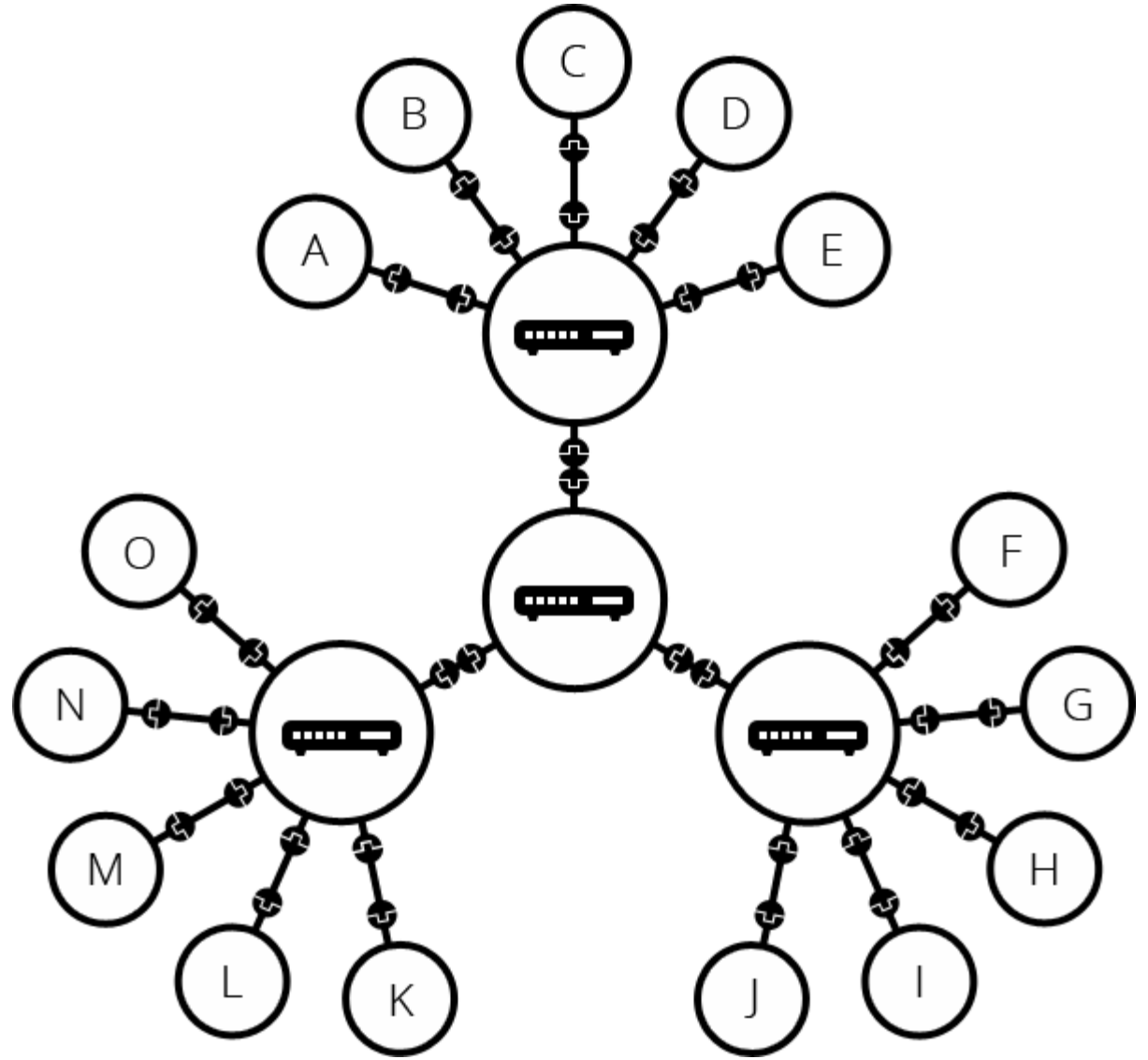
더 나은 방법

하나의 라우터를 통해 10대의 컴퓨터가
통신한다면 10개의 케이블만 있어도 된다.



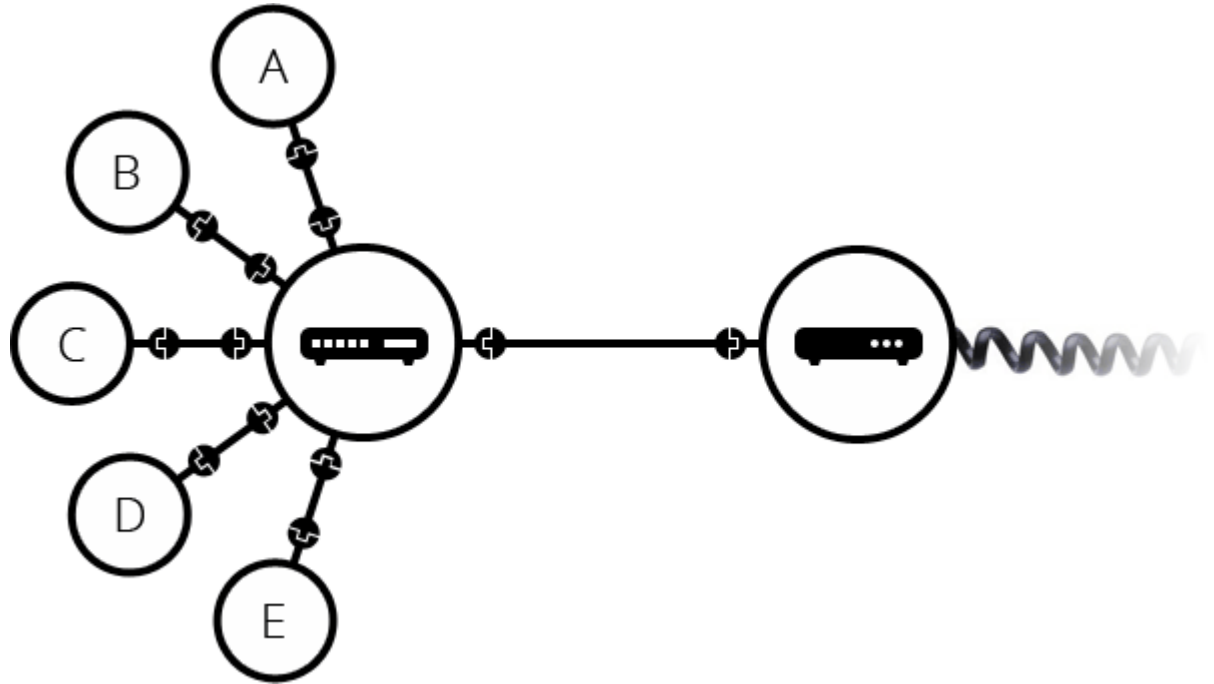
네트워크의 네트워크

라우터들을 서로 연결한다면 훨씬 많은
컴퓨터를 연결할 수 있다.



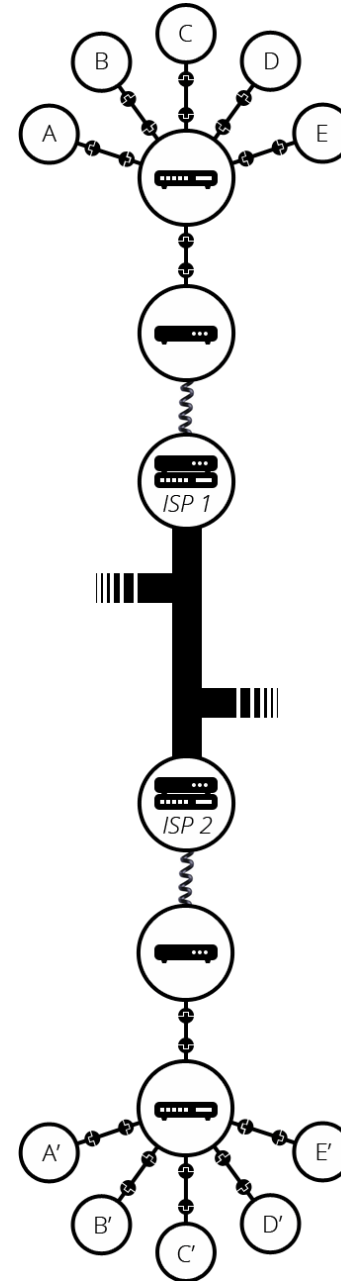
더 넓은 세상으로

라우터를 통한 네트워크는 물리적 제약이 있다.
따라서 전화선이나 광케이블과 같이 이미
설치된 회선을 이용해 다른 곳의 네트워크와
연결해야 한다. 이때 사용되는 장비를
모뎀(Modem)이라고 부른다.



인터넷

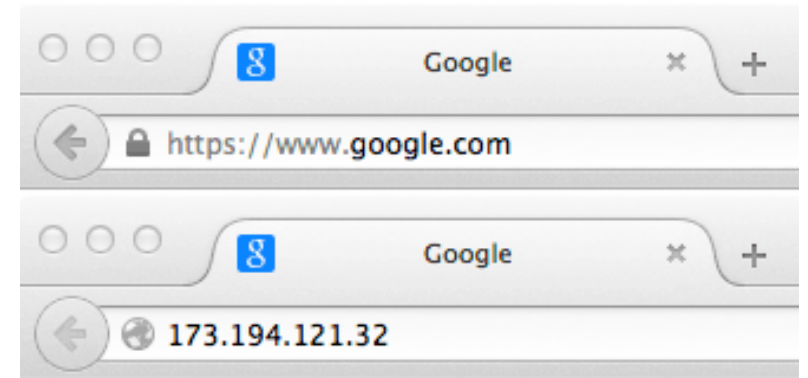
회선에 네트워크가 연결되었다면, ISP(Internet Service Provider)를 거쳐야 한다. ISP는 특수한 라우터를 관리하는 회사로, 한국에는 KT 인터넷, SK 브로드밴드, LG U+ 등이 있다. 인터넷은 이러한 네트워크 인프라로 구성되어 있다.



IP Address

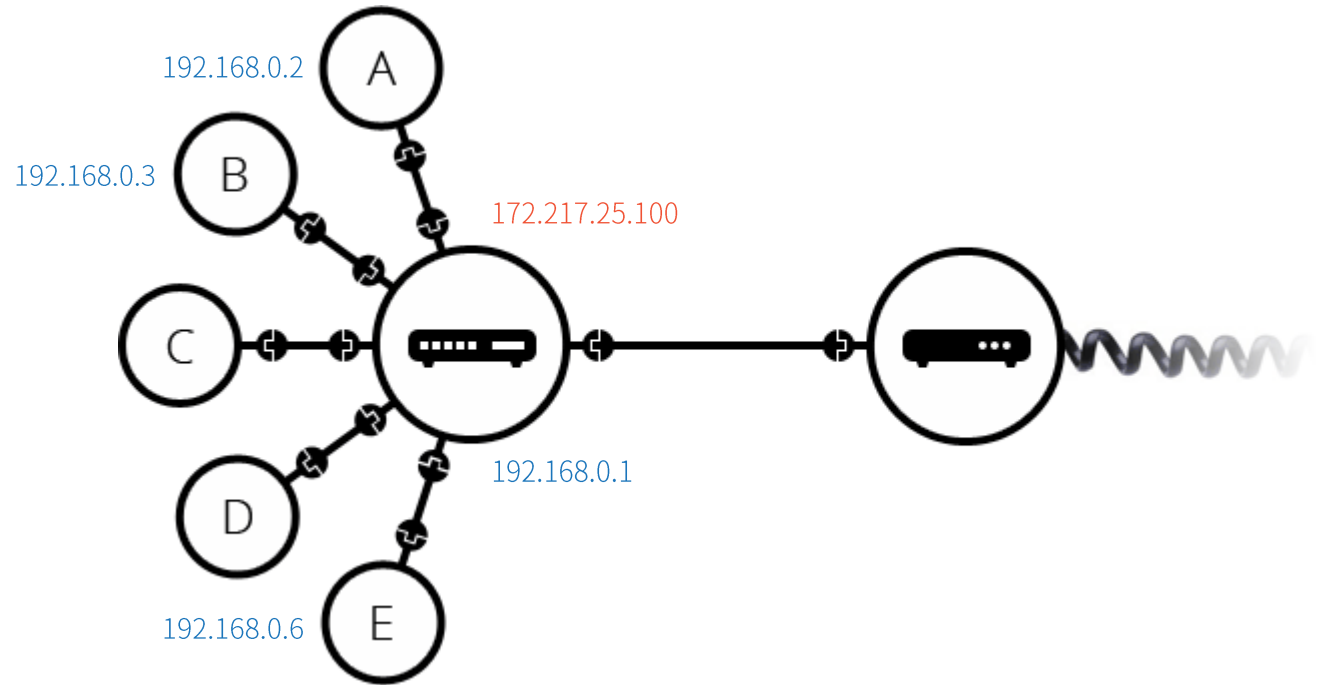
컴퓨터에는 IP 주소라는 고유 번호가 부여된다.

구글 서버의 IP 주소는 172.217.25.100이며, 사람이 기억하기
쉽게 google.com처럼 이름 붙인 것을 domain name이라고 한다.



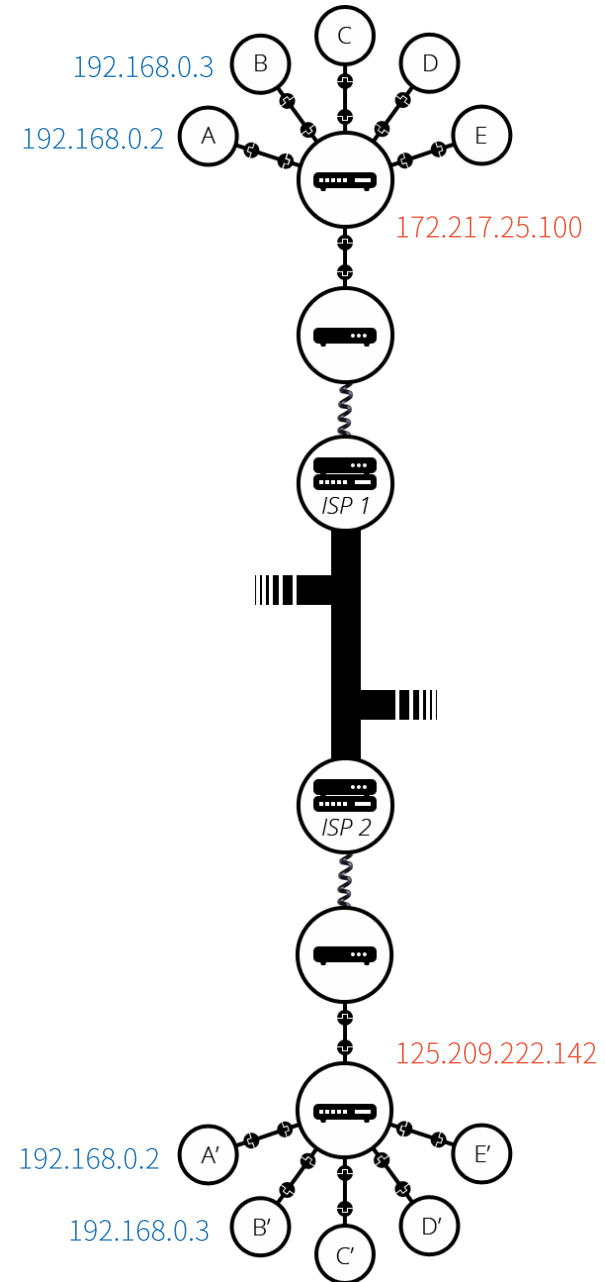
사설 IP와 공인 IP

1. ISP와 계약하고 케이블을 라우터의 WAN 포트에 연결하면 **공인 IP**가 부여된다.
2. 노트북, 스마트폰 등을 라우터의 LAN 포트에 연결하면 **사설 IP**가 부여된다.



사설 IP와 공인 IP

사설 IP는 다른 네트워크의 사설 IP와 중복될 수 있지만,
공인 IP는 세상에 단 하나만 존재한다. (공인 IP는 ISP가 부여한다.)
즉, 라우터는 IP를 절약하는 효과를 낸다.



유동 IP와 고정 IP

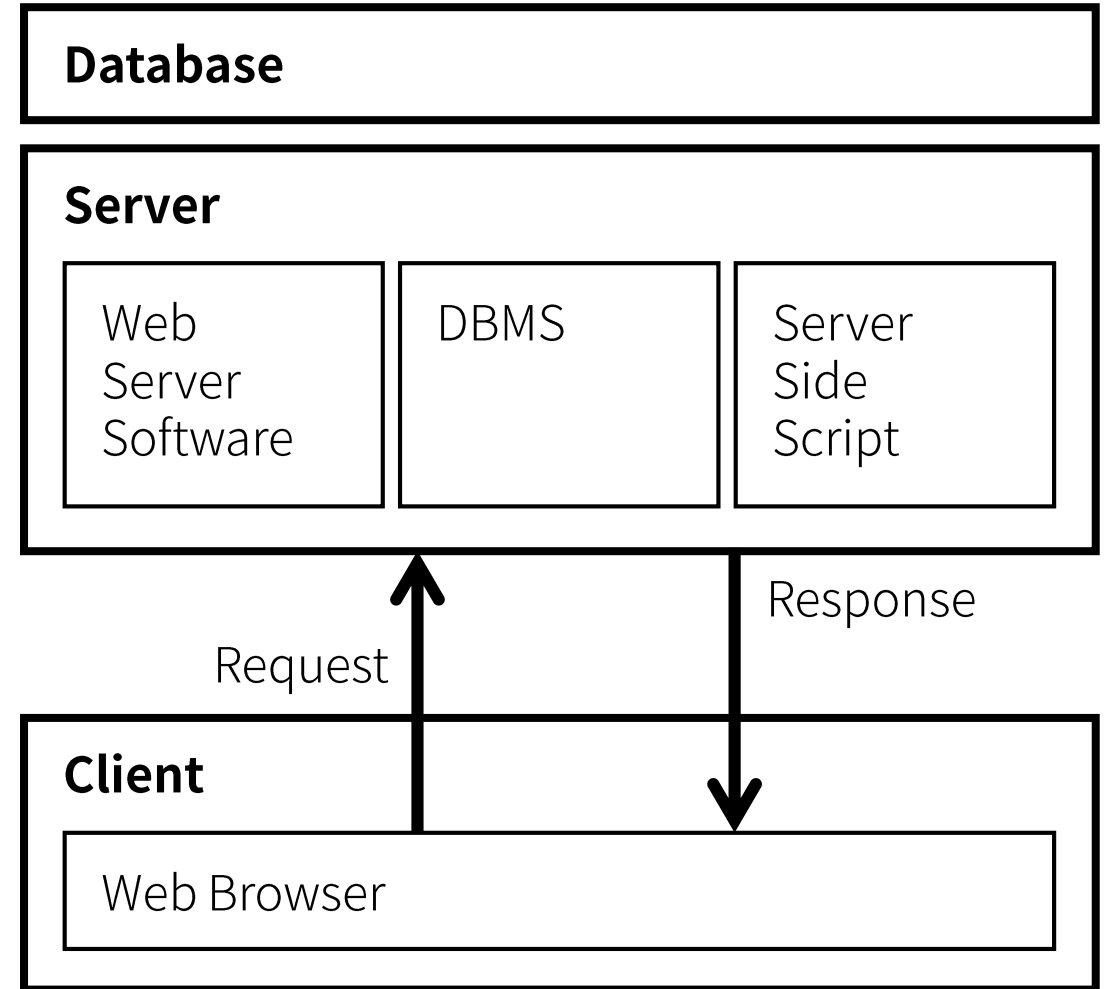
ISP는 IP 주소를 더욱 절약하기 위해 인터넷을 사용하지 않게 된 사용자의 IP를 다른 고객(컴퓨터)에게 부여한다.

ISP에 추가 비용을 지불하면 변경되지 않는 고정 IP를 받을 수 있다.

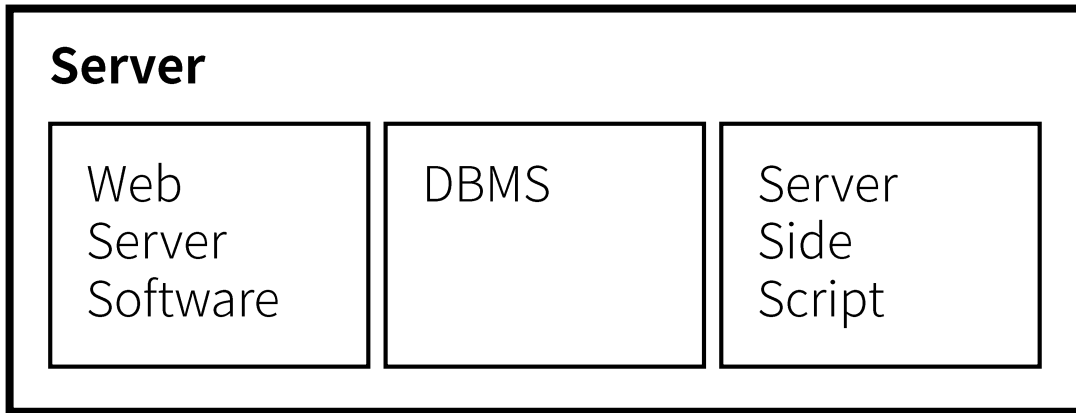
Client-Server Model

파일을 요청하는 클라이언트와 요청 받은 파일을 제공하는 서버로 구성된 네트워크 아키텍처. 다수의 클라이언트가 인터넷을 통해 서버와 통신한다.

HTML/CSS/JS는 클라이언트의 브라우저에서 해석, 실행되는 client-side 언어이고, PHP/JSP/Node.js는 서버에서 실행되는 server-side 언어다.



백엔드 개발 환경 세팅



Web Server Software

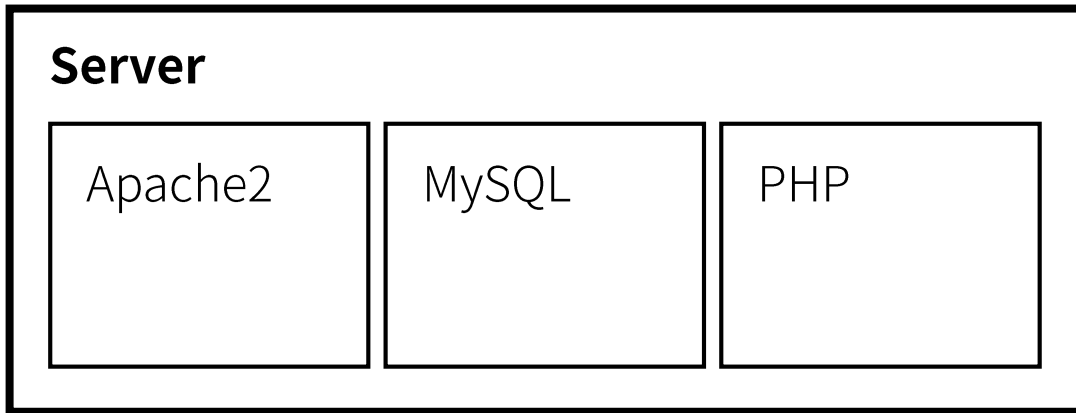
컴퓨터를 웹 서버로 만들어주는 프로그램.

DBMS

데이터베이스 관리 시스템(Database Management System)

Server Side Script

서버에서 실행되는 스크립트 언어.



따로 따로 설치할 필요없이 한 번에

AMP(Apache, MySQL, PHP)

서버를 구축할 수 있다!

→ <https://bitnami.com/stack/wamp>

Web Server Software

Apache2: 가장 많이 사용되는 웹 서버 소프트웨어.

DBMS

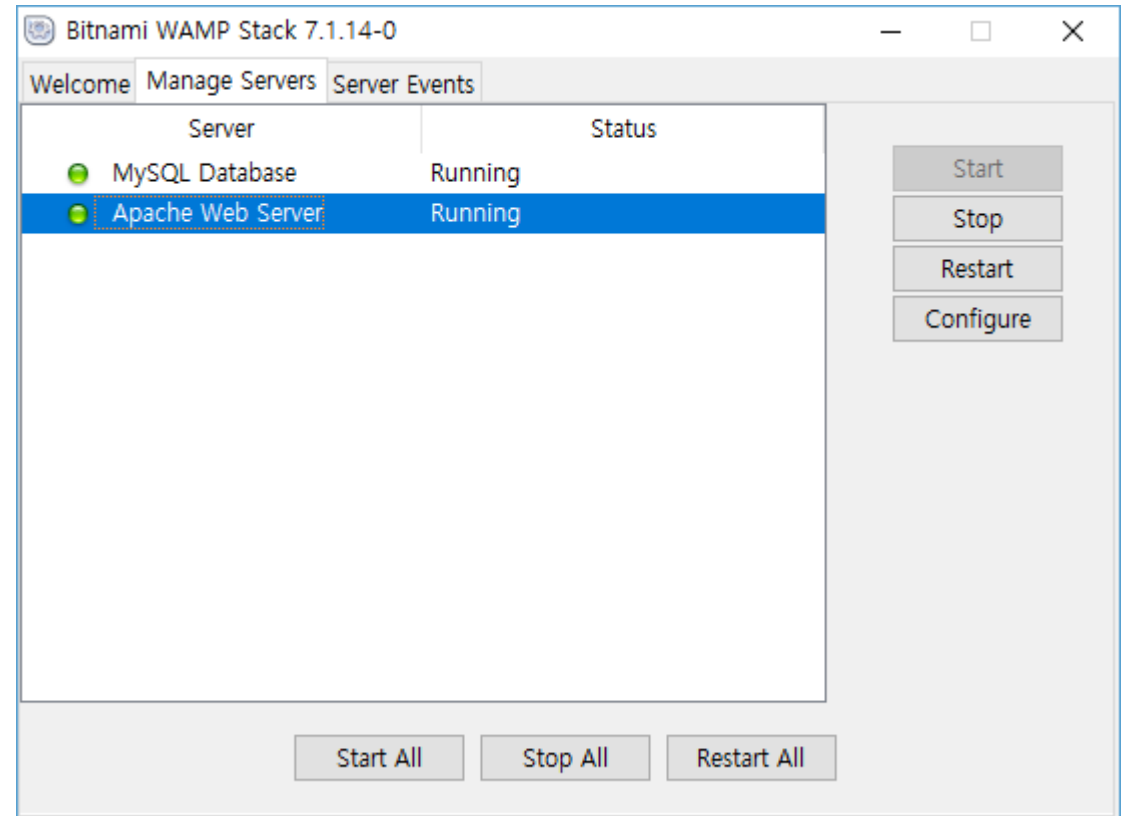
MySQL (MariaDB 호환): 가장 많이 사용되는 DBMS.

Server Side Script

PHP: 'PHP-Hypertext Preprocessor'. 높은 점유율을 자랑하는 서버사이드 스크립트 언어.

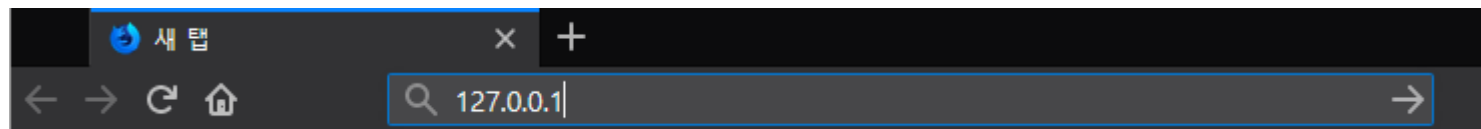
Bitnami WAMP Stack

‘Manage Servers’ 탭에서 데이터베이스와 웹 서버를 켜고 끌 수 있다.



로컬 서버 접속

브라우저에서 127.0.0.1에 접속하면...



로컬 서버의 첫 페이지를 볼 수 있다!

(127.0.0.1은 컴퓨터 자신을 뜻하는 루프백 아이피이며, localhost로도 접속 가능하다.)

첫 페이지 수정

윈도우 환경 기준, 디폴트 설정으로 설치했다면 탐색기를 열어

C:\Bitnami\wampstack-7.1.14-0\apache2\htdocs에 들어간다.

서버의 첫 페이지는 index.html이며, 해당 코드를 수정하고 브라우저를 새로고침하면 내용이 바뀐 것을 확인할 수 있다.

같은 네트워크의 기기에서 서버 접속

터미널에 ifconfig(UNIX) 또는 ipconfig(Windows)를 입력하면 IP 주소를 볼 수 있다.

```
wifi0  Link encap:UNSPEC  HWaddr C8-F7-33-C0-0C-0B-00-00-00-00-00-00-00-00-00-00
      inet addr:192.168.0.6  Bcast:192.168.0.255  Mask:255.255.255.0
      inet6 addr: fe80::c190:ab90:ed7:5322/64 Scope:Global
      UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
      RX packets:0 errors:0 dropped:0 overruns:0 frame:0
      TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0
      RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

```
무선 LAN 어댑터 Wi-Fi:

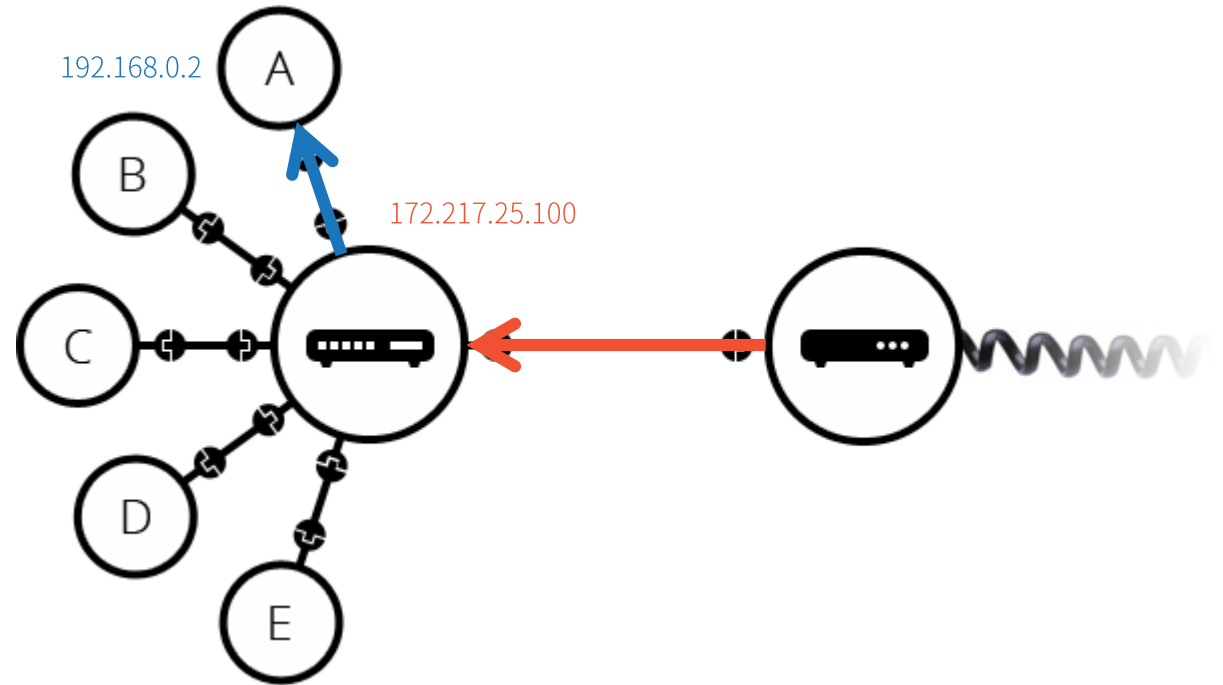
연결별 DNS 접미사. . . . . :
링크-로컬 IPv6 주소 . . . . . : fe80::c190:ab90:ed7:5322%4
IPv4 주소 . . . . . : 192.168.0.6
서브넷 마스크 . . . . . : 255.255.255.0
기본 게이트웨이 . . . . . : 192.168.0.1
```

같은 네트워크에 연결된 다른 기기의 브라우저에서 IP를 입력하면 서버에 접속할 수 있다.

다른 네트워크의 기기에서 서버 접속

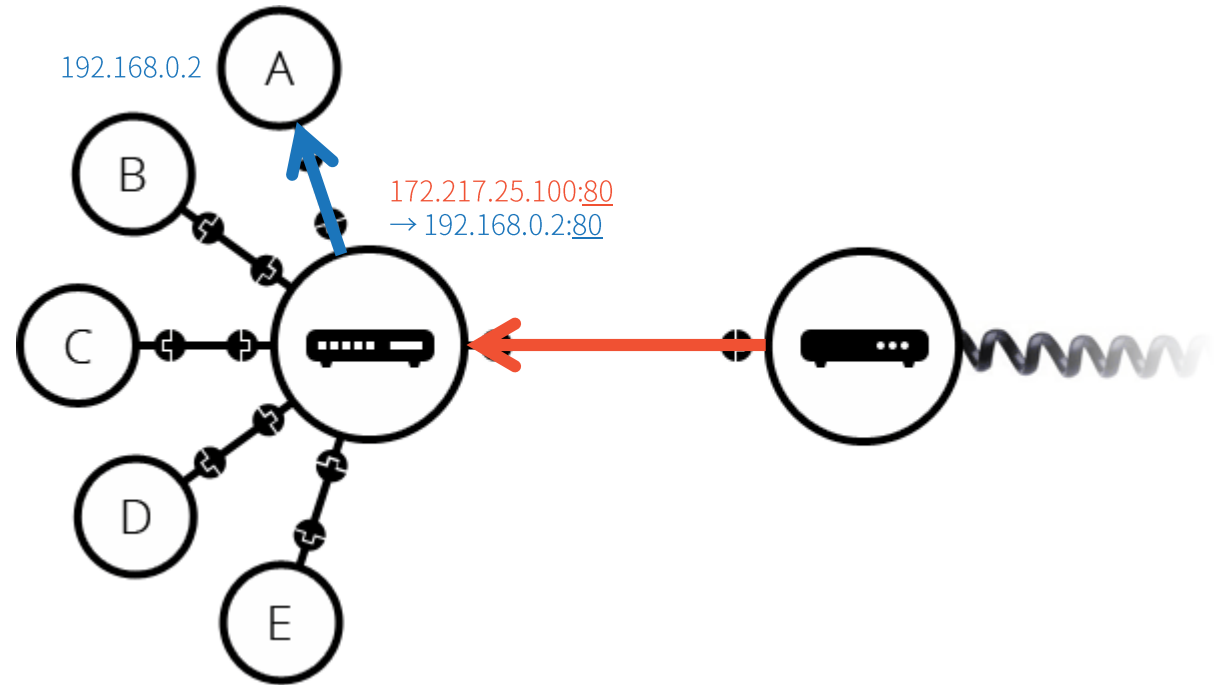
외부 네트워크에서는 A 서버 (192.168.0.2)에 바로 접속할 수가 없다.

따라서 라우터에 접속했을 때 A 서버로 연결되도록 해야 한다.



다른 네트워크의 기기에서 서버 접속

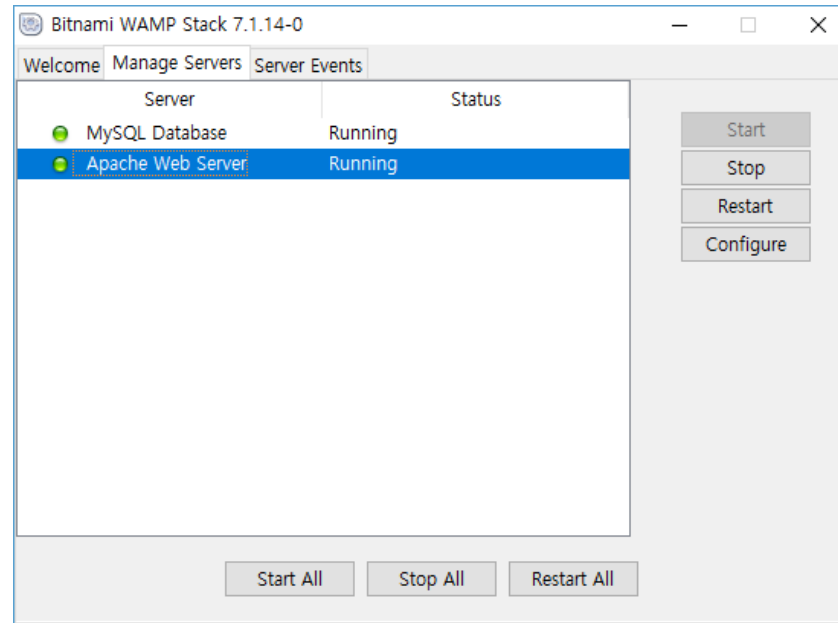
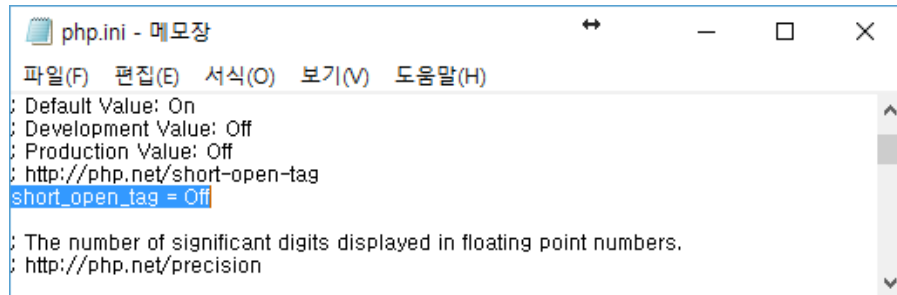
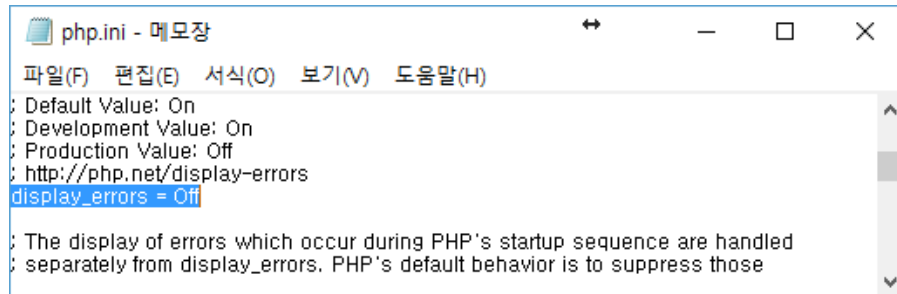
접속 포트(Port)에 따라 네트워크의 특정
기기로 연결되도록 포트 포워딩(Port
forwarding)하면 어디서든 접속이 가능하다.



PHP 맛보기

간단한 설정

1. C:\Bitnami\wampstack-7.1.14-0\php에 있는 php.ini 파일을 연다. (메모장 또는 vscode 등을 사용)
2. display_errors 옵션을 On으로 바꿔준다. (에러가 났을 때 어떤 에러인지 보여주는 옵션.)
3. short_open_tag 옵션을 On으로 바꿔준다. (단축구문을 허용하는 개발 편의 옵션.)
4. 저장하고 Apache Web Server를 restart해준다.



PHP 파일 만들기

PHP 문서의 확장자는 .php이다.

C:\Bitnami\wampstack-7.1.14-0\apache2\htdocs에 있는

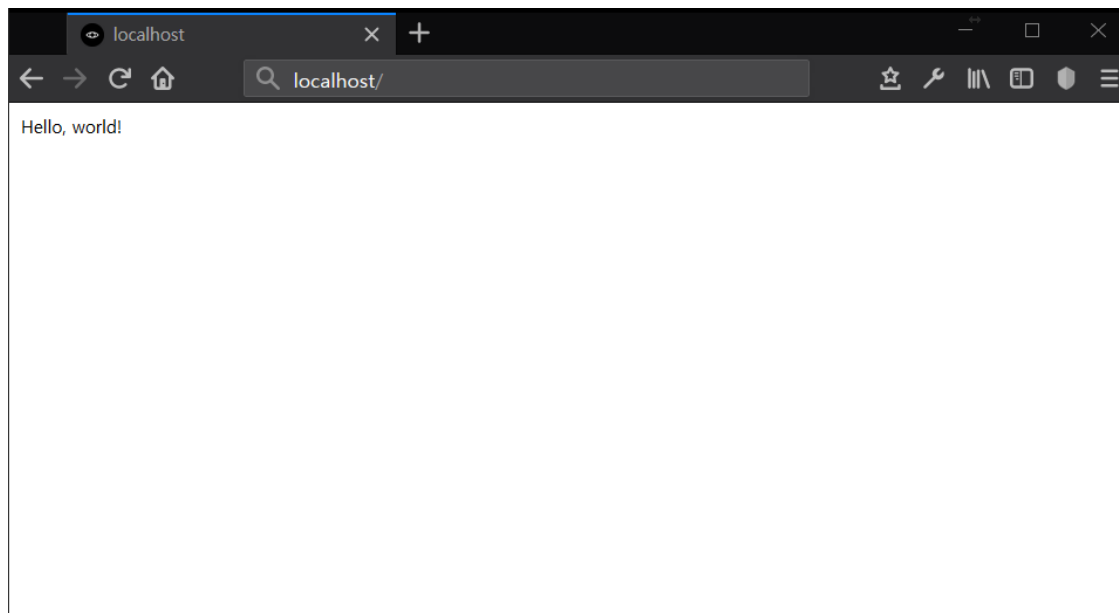
index.html 파일의 이름을 index.php로 바꾼다.

(또는 .php 형식의 새로운 파일을 만든다.)



index.php

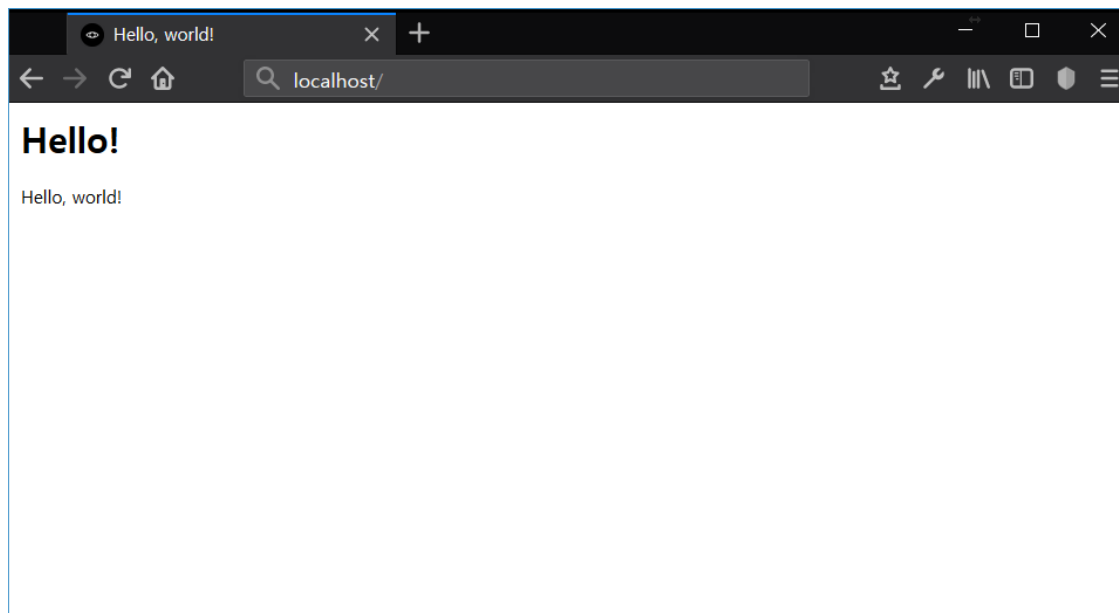
```
<?php  
    echo "Hello, world!";  
?>
```





index.php

```
<html>
<head>
  <meta charset="UTF-8">
  <title>Hello, world!</title>
</head>
<body>
  <h1>Hello!</h1>
  <?php
    echo "Hello, world!";
  ?>
</body>
</html>
```





\$a = 10;

\$b = 20;

echo \$a + \$b;

변수 산술 연산



```
$num = 10;  
$str = "hello";  
$arr = [10, 20, 30];  
echo $a;  
echo $str;  
echo $arr[1];
```

변수 타입



```
if ($a < $b) {  
    echo "True";  
} else {  
    echo "False";  
}
```

조건문



```
for ($i = 0; $i < 10; $i++) {  
    echo "counter: ".$i;  
}
```

반복문



```
function foo($num) {  
    return $num * $num;  
}  
echo foo(2);
```

함수



index.php

```
<html>
<head>
    <meta charset="UTF-8">
    <title>Hello, world!</title>
</head>
<body>
    <?php
        $hello = "Hello, world!";
        echo "<h1>".$hello."</h1>";
    ?>
</body>
</html>
```




index.php

```
<html>
<head>
    <meta charset="UTF-8">
    <title>Hello, world!</title>
</head>
<body>
    <h1>Hello!</h1>
    <?php
        function foo($num) {
            for ($i = 1; $i < 10; $i++) {
                echo $num." * ".$i." = ".$num * $i."<br/>";
            }
        }
        foo(2);
    ?>
</body>
</html>
```

데이터베이스와 MySQL

데이터베이스 (DB; Database)

자료들의 집합. 데이터를 체계적으로 조직해 DB를 구축한다.

관계형 데이터베이스 (Relational Database)

현재 가장 흔히 쓰이는 데이터베이스 모델. 데이터를 열과 행으로 구성된 테이블로 관리한다. 엑셀 표를 생각하면 된다.

비관계형 데이터베이스 (Non-Relational Database)

보다 유연한 데이터베이스 모델. 데이터의 관계가 복잡해져 구조를 변경해야 하는 상황에서 쉽게 대처할 수 있다.

데이터베이스 관리 시스템 (DBMS; Database Management System)

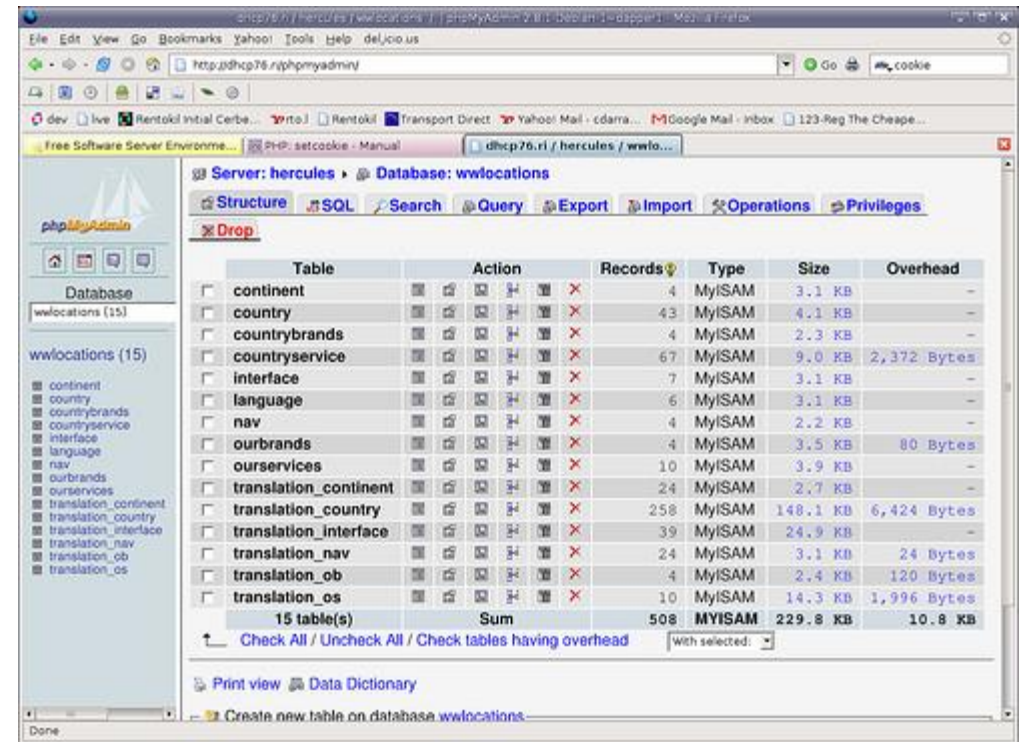
DB에 데이터를 Create, Read, Update, Delete (CRUD) 하고,
여러 관리자가 DB에 접근하는 것을 관리하기 위한 소프트웨어 시스템.

MySQL

세계에서 가장 많이 쓰이는 관계형 데이터베이스 관리 시스템.
라이선스 문제로 현재는 MySQL과 완벽 호환되는 MariaDB로 옮기는 추세다.

phpMyAdmin

웹을 통해 MySQL을 다루기 위해 만들어진 소프트웨어 툴.
앞서 AMP를 설치할 때 같이 설치됐기 때문에 별도로
다운로드할 필요없이 브라우저에서 바로
127.0.0.1/phpmyadmin에 접속하면 된다.



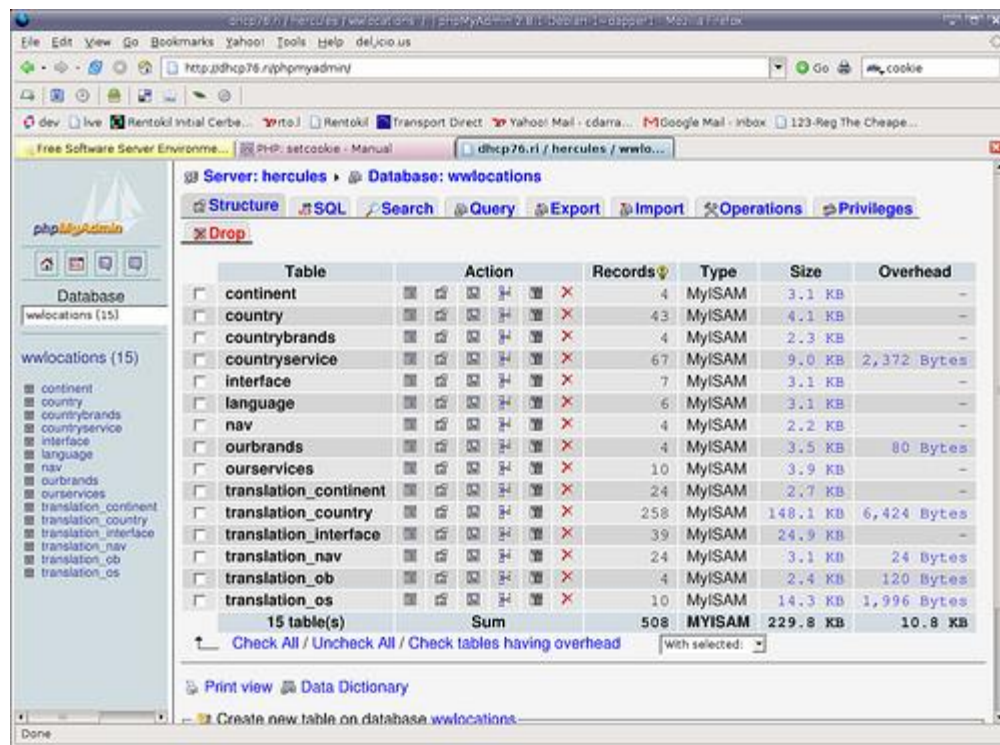
DB 만들기

사이드바의 'New'를 클릭하면 새로운 DB를 만들 수 있다.

DB는 여러 개의 테이블(Table)로 구성되어 있고,

테이블은 여러 개의 컬럼(Column)과

로우(Row)로 이뤄져 있다.



SQL 구문

user DB에서 id가 0인 row의 name을 선택:

```
SELECT name FROM user WHERE id = 0
```

user DB에 name이 park인 row를 추가:

```
INSERT INTO user (name) VALUES ("park")
```

user DB에서 id가 0인 row의 name을 kim으로 변경:

```
UPDATE user SET name = "kim" WHERE id = 0
```

user DB에서 id가 0인 row를 삭제:

```
DELETE FROM user WHERE id = 0
```

PHP로 SQL 다루기: DB 연결

```
<?
$user = "root";
$password = "pw";
$db = "timetable";

$conn = new mysqli("localhost", $user, $password, $db);
?>
```


PHP로 SQL 다루기: 데이터를 추가하는 경우

```
<?
```

```
... // 데이터 연결 부분 생략
```

```
$query = "SELECT name FROM user WHERE id = 0";
```

```
$result = mysqli_query($conn, $query);
```

```
while ($data = mysqli_fetch_array($result, MYSQLI_ASSOC)) {  
    echo $data["name"]; // id가 0인 row들의 name을 출력  
}
```

```
?>
```

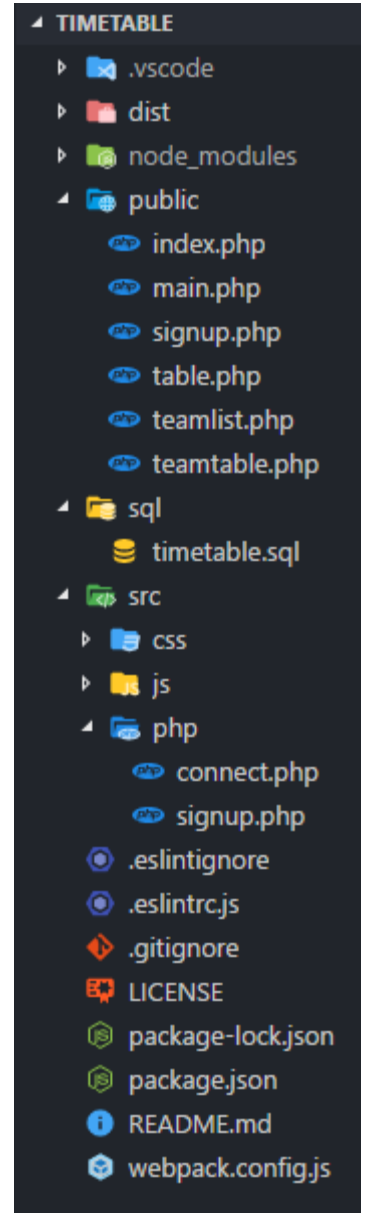
실전! 웹개발

이제부터는 실전 웹 개발을 위한
실용적인 내용을 다뤄볼게요.

개발을 위한 가이드라인

0. 먼저 `git pull`을 해주세요.
1. 모든 html 파일을 php 파일로 변경했습니다.
2. 더 이상 터미널에서 `npm run server`를 사용하지 않아도 됩니다. `npm run watch`만 실행해두고 `localhost/timetable/dist`에 접속하면 됩니다.
3. DB를 만들었습니다. phpmyadmin의 '가져오기' 탭에서 sql 폴더의 `timetable.sql` 파일을 선택하고 import해주세요.
4. team 테이블을 추가했습니다. timetable DB의 '가져오기' 탭에서 sql 폴더의 `team.sql` 파일을 선택하고 import해주세요.
5. php폴더의 `_connect.php` 파일의 `$user`, `$password`를 변경하세요.

+ 절대로 DB 비밀번호가 적힌 `php/_connect.php` 파일을 커밋하지 마세요!



PHP와 html이 섞이는 예제

php는 html과 섞어 쓸 수 있는 언어.

사진처럼 teamlist.php 파일 안에서 html과 php를 섞는 것도 가능하다.

```
You, a few seconds ago | 1 author (You)
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>팀 목록</title>
</head>
<body>
  <div id="container">
    <h1>내 팀 목록</h1>
    <ul id="team-list">
      <?
        require "../src/php/_connect.php";

        $query = "SELECT id, name FROM team ORDER BY id DESC"; // team 테이블에서 id 기준 내림차순으로 id와 name을 선택한다.
        $result = mysqli_query($conn, $query);

        while ($data = mysqli_fetch_array($result, MYSQLI_ASSOC)) { // 반복문을 돌며 data 변수에 선택한 데이터를 담는다.
          <?>
          <li>
            <a href="./teamlist?i=<?echo $data["id"];?>">
              <?echo $data["name"];?>
            </a>
          </li>
          <?>
        }
      <?>
    </ul>
  </div>
  <script src="../src/teamlist.min.js"></script>
</body>
</html>
```

PHP와 html이 섞이지 않는 예제

너무 섞이면 머리가 아프다.

_teamlist.php에서 DB 데이터를 받아와
JSON형식으로 인코딩하고, teamlist.js에서
AJAX 통신으로 _teamlist.php에서 인코딩한
데이터를 받아오는 식으로 구현할 수 있다.

```
teamlist.js x
You, a few seconds ago | 1 author (You)
1 import $ from "jquery";
2 import "../css/teamlist.css";
3
4 $.ajax({
5   type: "POST",
6   url: "../src/php/_teamlist.php",
7   dataType: "JSON",
8   success: function(result) {
9     renderList(result);
10  }
11 });
12
13 function renderList(list) {
14   for (let i = 0; i < list.length; i += 1) {
15     $("#team-list").append(
16       `<li><a href="/teamtable.html?i=${list[i].id}">${list[i].name}</a></li>`
17     );
18   }
19 }
```

```
_teamlist.php x
You, a few seconds ago | 1 author (You)
1 <?
2 require "../_connect.php";
3
4 $query = "SELECT id, name FROM team ORDER BY id DESC"; // team 테이블에서 id 기준 내림차순으로 id와 name을 선택한다.
5 $result = mysqli_query($conn, $query);
6
7 $teams = array();
8
9 while ($data = mysqli_fetch_array($result, MYSQLI_ASSOC)) { // 반복문을 돌며 data 변수에 선택한 데이터를 담는다.
10   $team = array(
11     "id" => $data["id"],
12     "name" => $data["name"]
13   );
14   $teams[] = $team;
15 }
16
17 echo json_encode($teams); // teams를 JSON 형식으로 인코딩
18 ?>
```

teamtable 페이지 결과

어느 방법이든 결과는 같다.

내 팀 목록

-
-
-