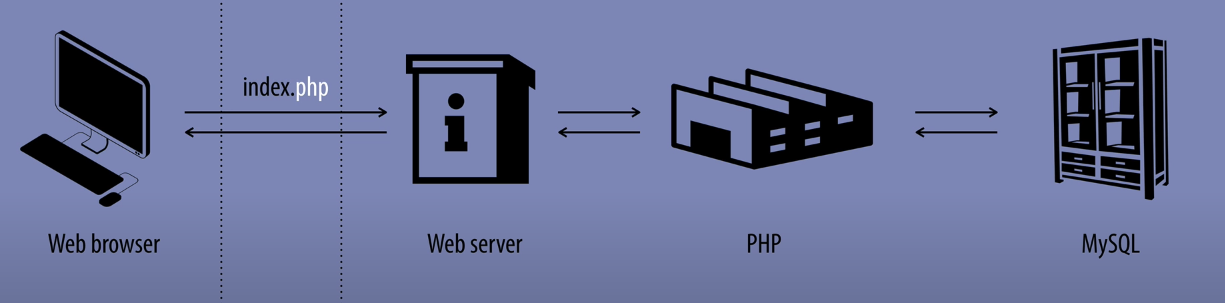
WEB3 - PHP & MySQL

사람의 욕심은 끝이 없다.

정보가 많고 다양해지면서 필요함에 따라 발달하게됨

파일에 저장하던걸 MySQL에 저장할거임



Php를 미들웨어라고도 부름

데이터베이스의 기능과 웹이 가지고 있는 접근성을 합쳐놓음

# 3.2.수업준비

1. WEB: Index.php

2. MySQL:

cd C:\Bitnami\wampstack-8.0.11-3\mariadb\bin

mysql -uroot -p

password:

create database

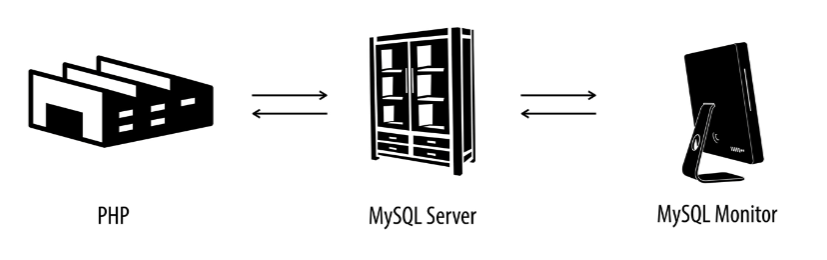
create table

primary key

1. 고유성?

2. 강력한 색인이 걸림(?)

MySQL 서버가 있음, PHP는 클라이언트 역할



Mysqli(improved)를 사용할거임(<https://www.php.net/manual/en/mysqli.quickstart.dual-interface.php>)

Php는 1.객체지향, 2.함수형식 2종류가 있음

우리는 이 중 함수형식으로 사용할거임

|  |
| --- |
| mysqli\_connect("localhost","root","ab97826431","opentutorials"); |
| mysqli\_query($conn,$sql); |

mysql generation\_log enable(?)처럼 php동작 로그를 분석하는 법을 나중에 가르쳐줄게

mysqli\_query

select, show, describe, explain을 mysql\_query를 성공했을때는 true, 에러떴을때는 false

$result = mysqli\_query($conn,$sql);

if($result == false) {

echo mysqli\_error($conn);

}

실제로는 에럴코드내용을 echo말고 파일로 저장하게 해서 보안을 늘림

/c/Bitnami/wampstack-8.0.11-3/apache2/logs 에 log파일 남음

절대로 log를 사용자에게 노출하지 말자

이렇게 사용하면 안됨

Select 할때는 데이터가 너무 많을 때 오래걸리기 때문에 limit을 걸자

Select.php

<?php

$conn=mysqli\_connect("localhost",'root','ab97826431','opentutorials');

$sql = 'select \* from topic limit 1000';

$result = mysqli\_query($conn,$sql);

var\_dump($result->num\_rows);

?>

mysqli\_fetch\_array()를 통해 mysql의 데이터를 array형식으로 가져옴

$row=(mysqli\_fetch\_array($result))

Index의 이름을 통해서 가져옴(연관 배열), 보통 index 번째를 이용해 가져옴(배열)

mysqli\_fetch\_array를 사용할때마다 한 row씩 지나감(read\_line비슷?)(더 이상 없으면 NULL)

\*모든 데이터형식에 대해 똑같이 동작하도록 설정해야함

while($row=(mysqli\_fetch\_array($result))){}

echo 안에 변수는 {~~}처리?

<?php

echo $list;

?>

* <?=$list?>

$list = $list."<li><a href=\"index.php?id={$row['title']}\">{$row['title']}</a></li>";

# 10.1.보안 - filtering

문제가 있는 정보가 안들어오게 하는 것(input 차단) filtering

문제가 있는 정보를 사용자한테 안주는 것(output 차단) escaping

사용자의 입력정보를 철저하게 불신하자

$filtered\_id=mysql\_real\_escape\_string($conn,$\_GET['id']);

# 10.2 보안 sql injection의 원리

-- : 주석

Die()

# 10.3 보안 - escaping

Htmlspecialchars()

<~> 을 php에서 입력했을 시 html에서는 태그로 동작할 수도 있음

* Htmlspecialchars()

# 11.1.활용 - 글쓰기 1

* 수정

Settype($foo,”integer”);

# 13.관계형데이터베이스의 도입

alter table topic add column author\_id INT(11);

update topic set author\_id=1 where id=6;

select \* from topic LEFT JOIN author ON topic.author\_id = author.id;

$select\_form .= '<select>';

while($row=(mysqli\_fetch\_array($result))) {

$select\_form .='<option>'.$row['name'].'</option>';

}

$select\_form .= '</select>';

->

<select name="author\_id"><option value="1">egoing</option><option value="2">duru</option><option value="3">taeho</option></select>

# 17.새로운 테이블 - 읽기

테이블 만들기

Redirection :header('Location: author.php');

Mysqli\_query에 사용할 sql문은 ‘’말고 “”에 묶여 있어야 함

DB삭제나 업데이트 작업같은경우는 중요하기 때문에 JS의 confirm같은 거로 재확인 필요

관계형 DB는 트리거 작업으로 관련된 유를 한번에 수정 가능

배운거로 놀다가 심심할떄 해볼만한 것

정보가 많아지면 검색 가능

<form method=”get” action=”search.php”>

색인 index 사용해봐

Mysqli말고

PDO(PHP DATA OBJECT)도 사용해봐

Doctrine, Propel, Aura SQL

AWS RDS, Google Cloud SQL for Mysql, Azure Database for MySQL