陣列

宣告與存取

■ 陣列索引值從 0 開始

```php
$arr = [5, "hi", 13.4];

echo $arr[0];   // 5
echo $arr[1];   // hi
echo $arr[2];   // 13.4
```

陣列與迴圈

- for 迴圈與 foreach 迴圈

  $animals = ["Lion", "Elephant", "Giraffe", "Zebra"];

```php
for($i = 0; $i < count($animals); $i++) {
    echo $animals[$i];
}
```

```php
foreach($animals as $animal) {
    echo $animal;
}
```

練習

- 將陣列內容以表格形式呈現在網頁上

輸出陣列內容

```php
<?php
$a = ["Hi", "Guys"];
?>
```

- print_r($a)

```
Array
(
    [0] => Hi
    [1] => Guys
)
```

- var_dump($a)

```
array(2) {
  [0]=>
  string(2) "Hi"
  [1]=>
  string(4) "Guys"
}
```

- var_export($a)

```
array (
  0 => 'Hi',
  1 => 'Guys',
)
```

字典

■ Key => Value 形式的陣列

```php
$users = [];
$users[count($users)] = ['name' => 'John', 'age' => 36];
$users[count($users)] = ['name' => 'Mei', 'age' => 27];
```

```php
foreach($users as $user) {
    echo $user['name'] . ": " . $user['age'] . "\n";
}
```

```php
foreach($users as $user) {
    foreach($user as $key => $value) {
        echo $key . ": " . $value . "\n";
    }
}
```

排序

- sort、ksort、asort

- rsort、krsort、arsort

JSON字串轉陣列

```php
<?php
$json = '[{"name": "David", "age": 36}, {"name": "Mei", "age": 27}]';
$arr = json_decode($json);
print_r($arr);
?>
```

```
Array
(
    [0] => stdClass Object
        (
            [name] => David
            [age] => 36
        )

    [1] => stdClass Object
        (
            [name] => Mei
            [age] => 27
        )

)
```

陣列轉JSON字串

```php
<?php
$arr = ['text' => '中文嘛也通'];
$str = json_encode($arr, JSON_UNESCAPED_UNICODE);
echo $str;
?>
```

{"text":"中文嘛也通"}

```php
<?php
$arr = [
    ['text' => '中文嘛也通'],
    ['text' => '今天天氣晴']
];
$str = json_encode($arr, JSON_UNESCAPED_UNICODE);
echo $str;
?>
```

[{"text":"中文嘛也通"},{"text":"今天天氣晴"}]