

# Match VS Switch

BY:

FATMA Elkassaby

# Syntactical Differences

```
switch (subject) {  
    case expr:  
        // execute block of statements  
        break; // or continue or return  
    default:  
        // execute block of statements  
        break; // or continue or return  
}
```

```
$result = match (subject) {  
    expr1, expr2 => value,  
    default => value,  
};
```

## THE Other Differences

1. **match** is an expression, while **switch** is a statement
2. In a **switch** statement, comparisons are made based on loose/weak equality (==).

```
SWITCH ('FOO') {  
    CASE 0:  
        $RESULT = "OH NO!\N";  
        BREAK;  
    CASE 'FOO':  
        $RESULT = "THIS IS WHAT I EXPECTED\N";  
        BREAK;  
    }  
    ECHO $RESULT;  
    //> OH NO!
```

- 3- In a **match** expression, comparisons are made based on strict equality (===).

```
ECHO MATCH ('FOO') {  
    0 => "Oh no!\n",  
    'FOO' => "THIS IS WHAT I EXPECTED\n",  
};  
  
//> THIS IS WHAT I EXPECTED
```

- 4- ال → **switch** → هتفضل تنفذ لحد متلاقى عبارة توقفها → break ,continue,....

```
$SUBJECT = 'TWO';  
  
SWITCH ($SUBJECT) {  
    CASE 'ONE':  
        ECHO 'ONE;';  
    CASE 'TWO':  
        ECHO 'TWO;';  
    CASE 'THREE':  
        ECHO 'THREE;';  
    DEFAULT:  
        ECHO 'DEFAULT;';  
}  
  
// OUTPUT: 'TWO;THREE;DEFAULT;'
```

- 5- ال → **match** → لا → بتنفذ الصح فقط من غير عبارات توقف

```
$SUBJECT = 'TWO';  
  
$RESULT = MATCH ($SUBJECT) {  
    'ONE' => 'ONE;',
```

```
'TWO' => 'TWO;',  
'THREE' => 'THREE;',  
DEFAULT => 'DEFAULT;',  
};  
  
ECHO $RESULT; // 'TWO;'
```

## Switch

### Pros:

It evaluates the expression once.

Makes the code more readable.

### Cons:

It does the loose comparison, sometimes not what we want.

## Match expression :

is recently added in php, is an expression and not a statement because it can be evaluated to a value that matches the condition, so it can be set to a variable or print it directly as we did above

### Pros:

It does the strict comparison, like we do manually with triple equal operator e.g. `10 === 10` true, `10 === '10'` return false, strict comparison helps us to catch errors and bugs at an early stage.

### Cons:

Not allowed to use multiline expressions so far like: