

المصفوفات والدوال :-

الفصل الثاني

* عناصرها سابقاً أنه يمكن للمتغيرات أن تحتوي فقط واحدة فقط، فجانبة المصفوفات لتل هذا القصور وتكون المبرمج من تخزين عدة قيم في متغير واحد يسمى المصفوفة.

* المصفوفة تتكون من ما يعرف بمفتاح أو معرف العنصر داخل المصفوفة وهو ال Key أو Index للمصفوفة ويسا من 0 إلى أقل من عدد عناصر المصفوفة بحد واحد.

* في ال PHP يمكن أن يكون النوع أي نوع من أنواع البيانات سواء عدد صحيح أو سكري أو قيمة منطقية أو القيمة الخارجية [] لتخزينية

حالة (1) مثال :-

Value = 10 // Key = 0 ← \$myArr[0] = 10;
Value = 12.5 // Key = 1 ← \$myArr[1] = 12.5;
Value = true // Key = 2 ← \$myArr[2] = true;
Value = user // Key = 3 ← \$myArr[3] = "user";

10	12.5	true	user
0	1	2	3

تبين لنا أنه يمكن تخزين أنواع مختلفة من البيانات داخل المصفوفة سواء كانت نص أم رقم صحيح أو سكري

```
for($i=0; $i<4; $i++)
{
    echo -----
}
```

حالة (2)

في هذا المثال تم استخدام حلقة
الكرار for لإخراج عناصر المصفوفة
حيث أن تدرج من 0 وحتى عدد عناصر
المصفوفة ناقص واحد
أي من 0 إلى 3

```
$myArr[0] = 10;  
$myArr[1] = 12.6;  
$myArr[2] = true;  
$myArr[3] = "user";
```

```
for($i=0; $i<4; $i++)  
{  
    echo _____  
}
```

حالة (3) إعطاء قيم للمصفوفة دفعة واحدة نستخدم الكلمة المحبوزة
Array ونوقع العناصر بين قوسين

مثال و
\$myArr = array(10, 12.6, true, "user");
for(\$i=0; \$i<4; \$i++)
{
 echo "\$myArr[\$i]";
}

حالة (4) لمعرفة عدد العناصر في المصفوفة نستخدم دالة Count

```
$myArr = array(10, 12.6, true, "user");  
for($i=0; $i<count($myArr); $i++)  
{  
    echo _____  
}
```


المصفوفات متعددة الأبعاد

نعمل على المصفوفة أحادية متراخلة

مثال

\$myarr[0][0] = "username" و

\$myarr[0][1] = "password" و

\$myarr[0][2] = 10 و

\$myarr[1][0] = 12 و

\$myarr[1][1] = 45199 و \Rightarrow

\$myarr[1][2] = true و

	0	1	2
0	user	Pass	10
1	12	45199	true

for (\$i = 0, \$i < count(\$myarr); \$i++)

حالة (2) السناد ممتلئة دفعة واحدة و -

\$myarr = array(

array('username', 'password', 10);

array(12, 45199, true)

);

...

هذه الحالة يتم السناد مصفوفة داخل مصفوفة بأي عدد من الأبعاد التي تريدها، أي أنه يمكن إنشاء مصفوفات ذات عدد أبعاد يكن لا يمكن الخاطئ منها بسهولة.

* المصفوفات المترابطة :

تكونت باستخدام سلاسل نصية لا يمكن للمصفوفة بدلاً من الأرقام حيث كل عنصر في المصفوفة يتألف من قسمين :- الأول المفتاح والثاني هو القيمة

مثال

```
$myarr['name'] = 'ahmad';
$myarr['age'] = 30;
$myarr['phone'] = 12568522;
$myarr['city'] = 'Nablus';
echo $myarr['name'].['age'] ----- و
```

يتم استخدام الدوال `var_dump` و `var_export` و `print_r` لعرض المصفوفة
وتحتوي المصفوفة من القيم ذات `key` وتقبل هذه الدوال وسيطاً واحداً
وهو المصفوفة المراد طباعتها

حالة ① `$myarr['name'] = 'username'`

لتم اسناد القيم من
هذه النوع عند تعريف دفعة
واحدة ع -

```
$myarr[] = "username 2"
$myarr['age'] = 30;
$myarr[] = 40;
$myarr['city'] = 'luxor';
$myarr[] = 'cairo';
$myarr['phone'] = 123456;
$myarr[] = 124891011 و
echo var_export($myarr);
```

```
$myarr = array('name' => 'username',
               'city' => 'luxor' و
               'phone' => 123456);
echo var_export($myarr);
```


دالة Foreach

مذافضل لطريقة للدوران على عناصر المصفوفة وبالأخص للترابطة

foreach (\$array as \$key => \$value)

{
 متاح المصفوفة
 البيانات المترابطة
}

* دوال التحكم بالمصفوفات :-

دالة explode

تقوم بتقطيع زفها وتحويله الى المصفوفة حيث تقبل وسيطين اجباريين (تستخدم هذه الدالة بكثرة عند لقراءة هذا الملفان (التي) الوسيط الاول ← الفاصلة الثاني ← النص

دالة implode

تكون هذه الدالة عكس explode اي انها تقوم بتحويل عناصر المصفوفة الى نص يفصل بينها فاصلة

الوسيط الاول ← الفاصلة الوسيط الثاني ← المصفوفة لمرار تحويل جميع عناصرها

الدالة array_is_array

تقوم هذه الدالة بالتحقق انه الوسيط يمرر لها هو مصفوفة - وذلك باعادة القيمة true او false

اضافة قيمة الى المصفوفة :-

* يتم اضافة عنصر الى المصفوفة بواسطة [] ← 'Sa' = \$array[]

* او باستخدام الدالة array-push حيث تقبل وسيطين الاول هو المصفوفة والثاني هو القيمة المراد اضافها ←

array-push(\$array, 'Sa')

* البحث داخل المصفوفة :-

نستخدم الدالة in_array للبحث داخل المصفوفة عند قيمته معينة وهذه الدالة تعيد true في حال نجاحها. ^{الدراسة}

```
if (in_array('sa', $array) == true)
{
    echo 'sa is found in $array'
}
```

↑ ^{النتيجة}

* قلب المصفوفة :-

نستخدم الدالة array_reverse لقلب ترتيب مصفوفة اي جعل اول عنصر اخر عنصر وهكذا

```
$array = array('1', '2', '3', '4');
$new_array = array_reverse($array);
print_r($new_array);
```

#output : Array ([0] => 4 [1] => 3 [2] => 2 [3] => 1)

الدالة array_unique تقوم بإزالة اي قيمة تتكرر في المصفوفة، حيث تعيد مصفوفة جديدة بدون اي عناصر مكررة

```
$new_array = array_unique($array);
```

* ترتيب عناصر المصفوفة :-

يتم ذلك بواسطة الدالتين sort ، asort حيث تقوم الدالة sort بترتيب عناصر مصفوفة تصاعدياً والشكل العام

حيث لا تعيد هذه الدالة اي قيمة تقوم بتعديل المصفوفة مباشرة. الوسيط الاول ← المصفوفة

```
sort($array);
```

لاحظ ان المصفوفة المرتبة لا تحتفظ بمفاتيح المصفوفة الاصلية ولا تحتفظ بها. نستخدم الدالة asort التي تقوم بعمل نفسه لدالة sort لكنها تحتفظ بقيم المفاتيح او المحرر

الدوال

توفر لغة PHP عدداً كبيراً من الدوال يتجاوز عددها الالف دالة ، لكنه يمكن
اذاً نحن سنحتاج الى دالة تقوم بجملة معينة والمكد العام لها :-

```
function functionname ( )  
{
```

جسم الدالة

وسائط الدالة : يمكنها تقبل اي نوع البيانات
وكذلك يمكنها اعادة جاي نوع من البيانات
ادعهم الرجوع باي قيمة هذه الدالة لا تأخذ اي وسيط
ولا تعيد اي قيمة فقط تطبع جملة المتصفح

```
function f_echo()  
{
```

```
echo "PHP: Hypertext Processor" ;  
}
```

اي الدالة التالية مني تأخذ وسيطاً لتقوم بجملة معينة ونمذد سي hi ولاحظ انه
ان المتغير \$in متغير محلي معرف داخل الدالة فقط

```
function f-eco($in)  
{
```

```
echo "<hi>$in</hi>" ;  
}
```