



تطبيقات الويب Web Application

2023-2024

Dr.-Eng. Samer Sulaiman

<https://manara.edu.sy/>

مفردات المنهاج

- أساسيات برمجة تطبيقات الويب
- تطبيقات الويب الستاتيكية
 - أساسيات لغة الـ HTML
 - أساسيات الـ CSS
 - أساسيات "JS" الـ Java-Script
- تطبيقات الويب الديناميكية
 - ربط صفحات الويب بقواعد البيانات
 - لمحة عن لغة الـ PHP

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



- ربط صفحات الويب بقواعد البيانات:
- صفحات الويب التفاعلية:
- في الأساس صفحة ويب تستخدم أنواعًا مختلفة من البرامج لإنشاء تجربة تفاعلية ثرية للمستخدم
- أي يسهل على المستخدم التفاعل بنشاط مع الموقع.
- مثال: موقع الويب الذي يعرض تنبؤات الطقس لمنطقة معينة.
- إذا كان موقع الويب تفاعليًا ، فإنه يتيح للمستخدم من كتابة الموقع ويعرض تقرير الطقس المفصل لتلك المنطقة. وبالتالي ستتغير محتويات صفحة الويب اعتمادًا على الموقع المعطى
- من أجل إنشاء موقع ويب تفاعلي ، نحتاج إلى قاعدة بيانات متصلة بالموقع لتوفير المعلومات المطلوبة.
- ما هي قاعدة البيانات؟
- مجموعة منظمة من المعلومات المهيكلة ، أو البيانات ، المخزنة عادةً إلكترونيًا.
- عادة ما يتم التحكم في قاعدة البيانات بواسطة نظام إدارة قواعد البيانات (DBMS).
- لربط قواعد البيانات بصفحات الويب لا بد من استخدام لغة وسيطة تتعامل مع السيرفر كـ (php,java,.Net,Node.js)
- حيث تعتبر لغة الHTML والJavaScript لغة الfront end وتعمل في جهة المستخدم ولا يمكن ان تتصل مباشرة بقاعدة البيانات بل يجب استخدام لغة الbackend لفعل هذا الشيء والتي تعمل ضمن المخدم ولا يراها المستخدم

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة الPHP:

- لغة PHP هي لغة برمجة نصية من جانب المخدم Server Side مفتوحة المصدر صُممت لتطوير تطبيقات برمجية مختلفة، كالمواقع الإلكترونية التفاعلية والمواقع الثابتة Static، وتستخدم بشكلٍ واسعٍ في عالم الويب كونها تمتلك عدة مميزات أهمها أنها تستطيع الاتصال مع قواعد البيانات
- يمكن تمثيل بنية الجملة البرمجية بلغة PHP عن طريق وضع شيفرة PHP ضمن وسّي بداية ونهاية خاصين هما `php?>` و `<?>` ويمكن تضمينها لغة HTML أو ضمن ملفات مستقلة
- حتى تعمل لغة PHP تحتاج إلى متصفح ومخدم ويب ومفسر php (Interpreter)، الذي يُنفذ التعليمات البرمجية ويجمع النتائج مباشرةً بنفس اللغة دون الحاجة إلى تحويلها إلى لغة آلة بعكس عمل المترجم (Compiler).
- يقتصر عملها على مخدم الويب Server-side بشكلٍ أساسي لمعالجة المدخلات من تفاعلات المستخدمين، والاستجابة لطلبات المتصفحات والأحداث Events، بعد الرجوع إلى المخدم والموافقة على الطلب. على عكس اللغات التي تعمل على جانب العميل Client-side، مثل JavaScript التي تُجري جميع الطلبات والأحداث على حاسوب المستخدم مباشرةً.
- لا يقتصر عملها على إخراج مستندات HTML، إذ يمكنها عرض أي نص بتنسيقاتٍ مختلفة كملفات XML أو XHTML

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة ال PHP:

- تتشارك لغة ال PHP مع اللغات الأخرى بأغلب الميزات من ناحية استخدام الجمل الشرطية والتكرار والعمليات الحسابية والتعامل مع المتغيرات حيث يتم التصريح عن متغير بإضافة البادئة \$ لاسم المتغير وهنا لا يوجد ضرورة لتحديد نوع المتغير قبل استخدامه
- تستخدم علامات التعليقات المعروفة في أغلب لغات البرمجة بالإضافة إلى استخدام # لتعليق سطر واحد
- من الممكن إدراج محتوى ملف PHP ضمن ملف PHP آخر (قبل أن ينفذه مخدم الويب) عن طريق استخدام `include` أو `require`
- عند حدوث خطأ أو عدم العثور على الملف المضمن:
 - ينتج عن `require` خطأ فادح (`E_COMPILE_ERROR`) ويوقف البرنامج النصي
 - بينما يُنتج عن `include` تحذيراً فقط (`E_WARNING`) وسيستمر النص البرمجي
 - تأخذ الشكل التالي:
 - `include 'filename';`
 - `require 'filename';`

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة ال PHP:

- يوجد عدة مكتبات متاحة ضمن لغة ال PHP تمنحها إمكانية كتابة `web applications` وأيضاً تضم مكتبات تمكن من الوصول إلى `FTP` ومنها يمكن من الوصول إلى قواعد البيانات `database` مثل `MySQL` و `SQLite` وهناك عدة توابع جاهزة أخرى
- تابع الاظهار: `echo` أو `print`
- مثال:

- ```
<?php
$txt1 = "Learn PHP";
$txt2 = "W3Schools.com";
$x = 5;
$y = 4;
echo "<h2>" . $txt1 . "</h2>";
echo "Study PHP at " . $txt2 . "
";
echo $x + $y;
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## أساسيات لغة الـ PHP:

- يوجد عدة مكتبات متاحة ضمن لغة الـ PHP تمنحها إمكانية كتابة web applications وأيضا تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات database مثل MySQL وSQLite وهناك عدة توابع جاهزة أخرى

• تابع الوقت والتاريخ: date(): وله الشكل التالي date(format,timestamp)

• يمكن تمثيل صيغة الوقت والتاريخ بإحدى المعاملات التالية:

• d - Represents the day of the month (01 to 31)

• m - Represents a month (01 to 12)

• Y - Represents a year (in four digits)

• l (lowercase 'l') - Represents the day of the week

• H - 24-hour format of an hour (00 to 23)

• h - 12-hour format of an hour with leading zeros (01 to 12)

• i - Minutes with leading zeros (00 to 59)

• s - Seconds with leading zeros (00 to 59)

• a - Lowercase Ante meridiem and Post meridiem (am or pm)

• مثال:

```
<?php
echo "Today is " . date("Y/m/d"). "
";
echo "Today is " . date("Y.m.d"). "
";
echo "Today is " . date("Y-m-d"). "
";
echo "Today is " . date("l");
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## أساسيات لغة الـ PHP:

- يوجد عدة مكتبات متاحة ضمن لغة الـ PHP تمنحها إمكانية كتابة web applications وأيضا تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات database مثل MySQL وSQLite وهناك عدة توابع جاهزة أخرى

• تابع الوقت والتاريخ: date(): وله الشكل التالي date(format,timestamp)

• مثال:

```
<?php
date_default_timezone_set("America/New.York");
echo "The time is " . date("h:i:s");
?>
```

• يمكن استخدام التابع mktime() لإنشاء صيغة تاريخ ووقت والذي يأخذ الشكل التالي:

• mktime(hour, minute, second, month, day, year)

• يستخدم التابع strtotime() لتحويل التاريخ والوقت من الصيغة النصية إلى صيغة التاريخ والوقت

• مثال:

```
<?php
$date=strtotime("10:30pm April 15 2023");
echo "Created date is " . date("Y-m-d h:i:s", $date);
?>
```

```
<?php
$date=strtotime("tomorrow");
echo date("Y-m-d h:i:s", $date). "
";
$date=strtotime("next Saturday");
echo date("Y-m-d h:i:s", $date). "
";
$date=strtotime("+3 Months");
echo date("Y-m-d h:i:s", $date). "
";
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## • أساسيات لغة ال PHP:

- يوجد عدة مكتبات متاحة ضمن لغة ال PHP تمنحها إمكانية كتابة web applications وأيضاً تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات مثل database My SQL و SQLite وهناك عدة توابع جاهزة أخرى

• تابع الوقت والتاريخ (date()): وله الشكل التالي

• مثال:

```
<?php
$startdate = strtotime("Saturday");
$enddate = strtotime("+6 weeks", $startdate);
while ($startdate < $enddate) {
 echo date("M d", $startdate) . "
";
 $startdate = strtotime("+1 week", $startdate);
}
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## • أساسيات لغة ال PHP:

- يوجد عدة مكتبات متاحة ضمن لغة ال PHP تمنحها إمكانية كتابة web applications وأيضاً تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات مثل database My SQL و SQLite وهناك عدة توابع جاهزة أخرى

• توابع التعامل مع الملفات:

• readfile() و fopen():

• تقوم الدالة readfile() بقراءة الملف وكتابته في المخزن المؤقت للإخراج.

• يأخذ الصيغة التالية: readfile(file, include\_path, context)

• البارامترات:

• File: مطلوب. يحدد الملف للقراءة

• include\_path: اختياري.. يتم إعطاؤها قيمة TRUE إذا كان المراد البحث عن الملف في include\_path أو في php.ini أيضاً

• Context: اختياري يحدد سياق الملف. السياق عبارة عن مجموعة من الخيارات التي يمكنها تعديل سلوك الدفق

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## • أساسيات لغة الـ PHP:

- يوجد عدة مكتبات متاحة ضمن لغة الـ PHP تمنحها إمكانية كتابة web applications وأيضا تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات database مثل MySQL و SQLite وهناك عدة توابع جاهزة أخرى
- توابع التعامل مع الملفات:
  - fopen() و readfile()
  - يعتبر التابع fopen أكثر مرونة بسبب تقديمه العديد من الخيارات الغير موجودة في التابع readfile
  - يمكن تلخيص البارامترات التي يأخذها التابع fopen:
    - r: Open a file for read only. File pointer starts at the beginning of the file
    - w: Open a file for write only. Erases the contents of the file or creates a new file if it doesn't exist. File pointer starts at the beginning of the file
    - a: Open a file for write only. The existing data in file is preserved. File pointer starts at the end of the file. Creates a new file if the file doesn't exist
    - x: Creates a new file for write only. Returns FALSE and an error if file already exists
    - r+: Open a file for read/write. File pointer starts at the beginning of the file
    - w+: Open a file for read/write. Erases the contents of the file or creates a new file if it doesn't exist. File pointer starts at the beginning of the file
    - a+: Open a file for read/write. The existing data in file is preserved. File pointer starts at the end of the file. Creates a new file if the file doesn't exist
    - x+: Creates a new file for read/write. Returns FALSE and an error if file already exists

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## • أساسيات لغة الـ PHP:

- يوجد عدة مكتبات متاحة ضمن لغة الـ PHP تمنحها إمكانية كتابة web applications وأيضا تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات database مثل MySQL و SQLite وهناك عدة توابع جاهزة أخرى
- توابع التعامل مع الملفات:
  - fopen() و readfile()
  - مثال:
    - ```
<?php
$myfile = fopen("webdictionary.txt", "r") or die("Unable to open file!");
echo fread($myfile, filesize("webdictionary.txt"));
fclose($myfile);
?>
```
 - يمثل البارامتر الأول لتابع fread على اسم الملف المراد القراءة منه بينما يحدد البارامتر الثاني الحد الأقصى لعدد وحدات البايت للقراءة.
 - من أجل قراءة سطر واحد تستخدم الدالة fgets()
 - تستخدم الدالة feof() للتحقق من الوصول إلى نهاية الملف
 - مثال:
 - ```
<?php
$myfile = fopen("webdictionary.txt", "r") or die("Unable to open file!");
while(!feof($myfile)) {
echo fgets($myfile) . "
";
fclose($myfile);
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## • أساسيات لغة الـ PHP:

- يوجد عدة مكتبات متاحة ضمن لغة الـ PHP تمنحها إمكانية كتابة web applications وأيضاً تضم مكتبات تمكن من الوصول إلى FTP ومنها يمكن من الوصول إلى قواعد البيانات database مثل MySQL و SQLite وهناك عدة توابيع جاهزة أخرى

• توابيع التعامل مع الملفات:

• fopen() و readfile():

- تستخدم الدالة fwrite لكتابة محتوى ضمن الملف
- اعتماداً على الخيار w أو a يتم الكتابة فوق الملف أو إضافة إلى الملف
- مثال:

- ```
<?php
$myfile = fopen("newfile.txt", "w") or die("Unable to open file!");
$txt = "John Doe\n";
fwrite($myfile, $txt);
$txt = "Jane Doe\n";
fwrite($myfile, $txt);
fclose($myfile);
?>
```

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة الـ PHP:

- يمكن أيضاً إنشاء دوال وتوابيع من قبل المستخدم ويتم ذلك عن طريق الكلمة المحجوزة function على الشكل التالي:

- ```
function functionName() {
 code to be executed;
}
```

• مثال:

- ```
<?php
function familyName($fname) {
    echo "$fname Refsnes.<br>";
}
function familyName($fname, $year) {
    echo "$fname Refsnes. Born in $year <br>";
}
function addNumbers(int $a, int $b) {
    return $a + $b;
}
familyName("Jani");
familyName("Hege", "1975");
echo addNumbers(5, "5 days");
?>
```

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة الـ PHP:

- يمكن أيضاً إنشاء دوال وتوابع من قبل المستخدم ويتم ذلك عن طريق الكلمة المحجوزة function على الشكل التالي:

```
function functionName() {
    code to be executed;
}
```

• مثال:

```
<?php declare(strict_types=1);
function addNumbers(int $a, int $b) {
    return $a + $b;
}
function setHeight(int $minheight = 50) {
    echo "The height is : $minheight <br>";
}
echo addNumbers(5, "5 days");// since strict is enabled and "5 days" is not an integer, an error will be thrown
setHeight(350);
setHeight();
?>
```

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة الـ PHP:

• التعامل مع النماذج Form:

- يتم استخدام المعاملات العامة \$_GET و \$_POST لتجميع البيانات من النماذج
- مثال:

```
<form action="welcome.php" method="post">
Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>

<?php echo $_POST["name"];?><br>
Your email address is: <?php echo $_POST["email"];?>

<form action="welcome_get.php" method="get">
Name: <input type="text" name="name"><br>
E-mail: <input type="text" name="email"><br>
<input type="submit">
</form>

<?php echo $_GET["name"];?><br>
Your email address is: <?php echo $_GET["email"];?>
```

• مثال:

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة الـ PHP:

• التعامل مع النماذج Form:

• يتم استخدام المعاملات العامة \$_GET و \$_POST لتجميع البيانات من النماذج
• مثال:

- ```
<form action="welcome.php" method="post">
Name: <input type="text" name="name">

E-mail: <input type="text" name="email">

<input type="submit">
</form>
```
- ```
<?php echo $_POST["name"];?><br>
Your email address is: <?php echo $_POST["email"];?>
```
- ```
<form action="welcome_get.php" method="get">
Name: <input type="text" name="name">

E-mail: <input type="text" name="email">

<input type="submit">
</form>
```
- ```
<?php echo $_GET["name"];?><br>
Your email address is: <?php echo $_GET["email"];?>
```

• مثال:

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



• أساسيات لغة الـ PHP:

• التعامل مع قواعد البيانات:

• يتم استخدام الدالة mysql لنشاء اتصال مع قاعدة البيانات mysql والتي سيتم اعتمادها في هذا المقرر
• مثال:

- ```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$conn = new mysqli($servername, $username, $password);
if ($conn->connect_error) {
 die("Connection failed: " . $conn->connect_error);
}
echo "Connected successfully";
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## أساسيات لغة الـ PHP:

- التعامل مع قواعد البيانات:
- انشاء قاعدة بيانات عن طريق صفحة الويب
- مثال:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$conn = new mysqli($servername, $username, $password);
if ($conn->connect_error) {
 die("Connection failed: " . $conn->connect_error);
}
$sql = "CREATE DATABASE myDB";
if ($conn->query($sql) === TRUE) {
 echo "Database created successfully";
} else {
 echo "Error creating database: " . $conn->error;
}
$conn->close();
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



## أساسيات لغة الـ PHP:

- التعامل مع قواعد البيانات:
- ادخال البيانات إلى قاعدة البيانات عن طريق تعليمات الـ SQL والدالة query
- مثال:

```
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDB";
$conn = new mysqli($servername, $username, $password, $dbname);
if ($conn->connect_error) {
 die("Connection failed: " . $conn->connect_error);
}
$sql = "INSERT INTO MyGuests (firstname, lastname, email)
VALUES ('John', 'Doe', 'john@example.com)";
if ($conn->query($sql) === TRUE) {
 echo "New record created successfully";
} else {
 echo "Error: " . $sql . "
". $conn->error;
}
$conn->close();
?>
```

<https://manara.edu.sy/>

# تطبيقات الويب الديناميكية



- أساسيات لغة الـ PHP:
- التعامل مع قواعد البيانات:
- الاستعلام عن البيانات من قاعدة البيانات
- يتم ذلك عن طريق تعليمات الـ SQL على الشكل التالي:
- `SELECT column_name(s) FROM table_name`
- `SELECT * FROM table_name`
- `SELECT column_name(s) FROM table_name WHERE column_name operator value`
- مثال:
- ```
<?php
$sql = "SELECT id, firstname, lastname FROM MyGuests WHERE lastname='Doe'";
$result = $conn->query($sql);
if ($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
        echo "id: " . $row["id"]. " - Name: " . $row["firstname"]. " " . $row["lastname"]. "<br>";
    }
} else {
    echo "0 results";
}
$conn->close();
?>
```

<https://manara.edu.sy/>

تطبيقات الويب الديناميكية



- أساسيات لغة الـ PHP:
- التعامل مع قواعد البيانات:
- الاستعلام عن البيانات ومن ثم ترتيبها:
- `SELECT column_name(s) FROM table_name ORDER BY column_name(s) ASC|DESC`
- مثال:
- `$sql = "SELECT id, firstname, lastname FROM MyGuests ORDER BY lastname";`
`$result = $conn->query($sql);`
- حذف سجل من قاعدة البيانات عن طريق التعليمات التالية:
- `DELETE FROM table_name`
`WHERE some_column = some_value`
- مثال:
- `$sql = "DELETE FROM MyGuests WHERE id=3";`
- تحديث سجل ضمن قاعدة البيانات عن طريق التعليمات التالية:
- `UPDATE table_name`
`SET column1=value, column2=value2,...`
`WHERE some_column=some_value`
- مثال:
- `$sql = "UPDATE MyGuests SET lastname='Doe' WHERE id=2";`

<https://manara.edu.sy/>