

جامعة طرابلس  
كلية تقنية المعلومات  
قسم نظم المعلومات

المقرر الدراسي:

تطوير التطبيقات

Application development

ITIS311

المحاضرة ( 1 )

اعداد: أ. فاطمة بشير القاضي  
F.alqadhi@uot.edu.ly

# ما هي لغة PHP؟

- **PHP** اختصار لمصطلح **Hypertext Pre\_Processor** أي لغة المعالج المسبقة للنص التشعبي، وهي عبارة عن لغة برمجة مفتوحة المصدر تستخدم لأغراض البرمجة بشكل عام، ولغرض تطوير مواقع الويب بشكل خاص.
- تتميز لغة PHP بكونها لغة بسيطة وخفيفة الوزن تعمل من طرف الخادم **Server-side Languages** وهي قابلة للتضمين ضمن شفرة **HTML** ولغات البرمجة النصية الأخرى من جانب العميل **Client-side Languages**.
- كما تدعم لغة PHP أيضًا مفاهيم البرمجة كائنية التوجه **OOP** التي تتضمن مفاهيم الفئات أو الأصناف **Classes** والكائنات **Objects** والوراثة **Inheritance** وما إلى ذلك..

# ماذا يمكن أن تفعل لغة PHP؟

- إنشاء محتوى حيوي \_ديناميكي\_ لصفحات الانترنت.
- إنشاء الملفات وفتحها وقراءتها وكتابتها وحذفها وإغلاقها على الخادم.
- جمع بيانات النموذج.
- إرسال واستقبال ملفات تعريف الارتباط cookies.
- إضافة أو حذف أو تعديل البيانات في قاعدة البيانات الخاصة بك.
- التحكم في وصول المستخدمين.
- تشفير البيانات.

# أشهر مميزات لغة PHP

- مفتوحة المصدر Open Source: يتوفر كود PHP المصدر والبرمجيات مجانًا على الويب. بذلك يمكنك تطوير جميع إصدارات PHP وفقًا لمتطلباتك دون دفع أي تكلفة.
- الأداء Performance: يتم تنفيذ نص PHP بشكل أسرع بكثير من تلك النصوص المكتوبة بلغات أخرى مثل JSP و ASP، تستخدم PHP ذاكرتها الخاصة، لذلك يتم تقليل حجم عمل الخادم ووقت التحميل تلقائيًا، مما يؤدي إلى سرعة معالجة أسرع وأداء أفضل.
- بناء الجملة Syntax سهل: لغة PHP لها تركيب سهل الفهم.
- سهولة التضمين: يمكن تضمين تعليمات PHP البرمجية بسهولة في علامات HTML والنص البرمجي.
- متعددة المنصات: تعمل لغة PHP على أنظمة التشغيل Windows و Mac و Linux ويمكن تنفيذ تطبيق PHP الذي تم تطويره في نظام تشغيل واحد بسهولة في أنظمة تشغيل أخرى أيضًا.
- دعم قواعد البيانات: تدعم لغة PHP جميع قواعد البيانات الرائدة مثل MySQL و SQLite وقواعد بيانات كثيرة.
- الحماية: تعتبر لغة آمنة لتطوير المواقع، حيث تتكون من طبقات متعددة من الأمان لمنع الهجمات الضارة.
- بسيطة: لغة PHP سهلة التعلم والبدء فيها.
- سريعة: عادةً ما تعمل مواقع PHP بسرعة كبيرة.
- الدعم: لغة PHP لديها مجتمع نشط على الإنترنت يساعدك كلما واجهت مشكلة.

# ما معنى لغة من طرف الخادم أو لغة من طرف العميل؟

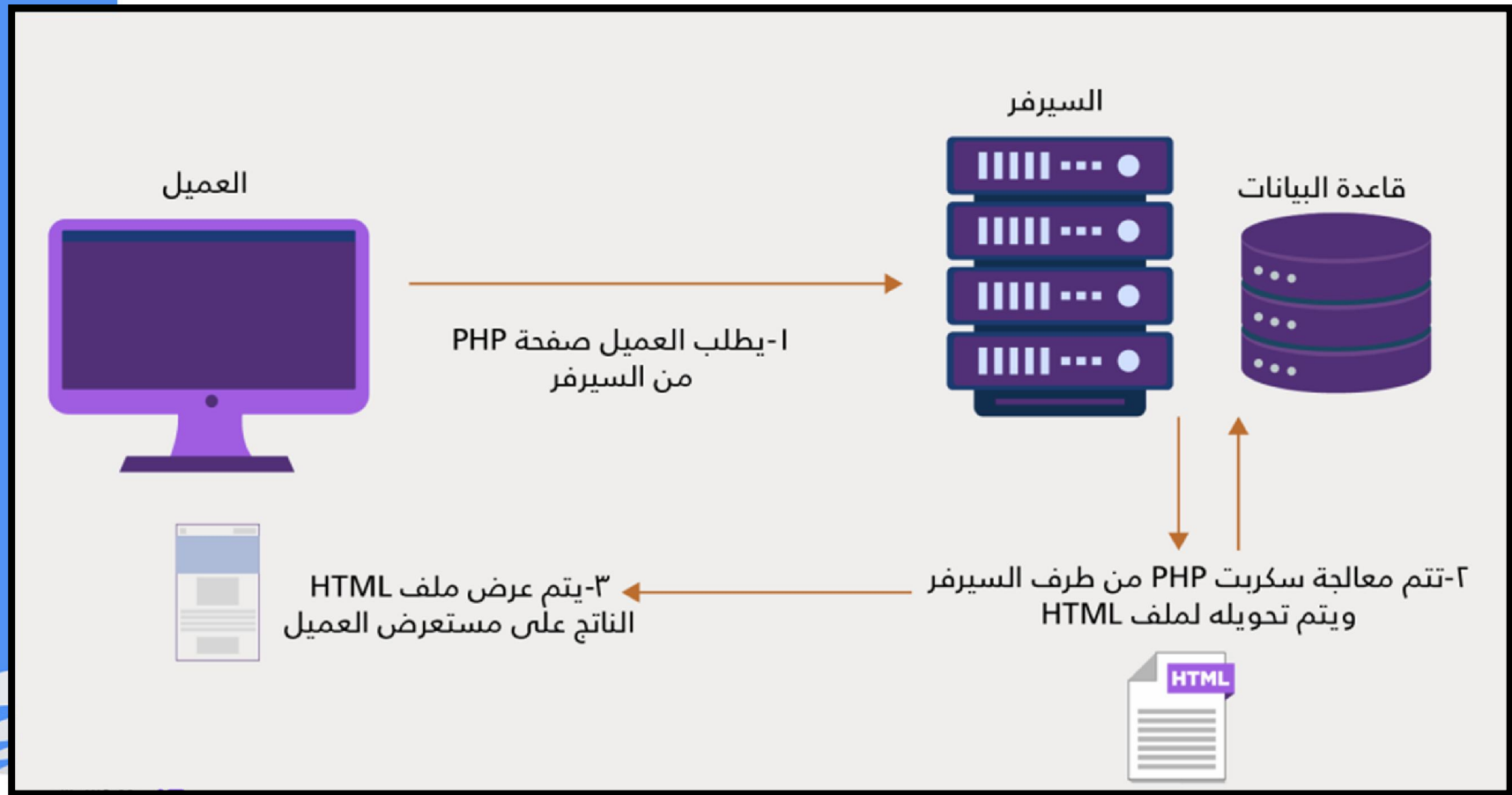
كما ذكرنا مسبقاً تعتبر لغة PHP لغة برمجة نصية من طرف الخادم بخلاف بعض لغات تطوير الويب الأخرى التي تكون من طرف العميل فماذا يعني هذا؟  
فعلياً يتم تصنيف لغات برمجة الويب إلى نوعين:

➤ لغة برمجة من جانب العميل Client side Programming

➤ لغة برمجة من جانب الخادم Server side Programming

عندما تقوم بزيارة صفحة ويب مكتوبة بلغة من طرف العميل مثل (HTML أو CSS أو JavaScript) يتم إرسال الكود مباشرة إلى متصفحك ليقوم المتصفح بمعالجة المعلومات وإخراجها بشكل صفحة ويب، وبالتالي هذه اللغات تعتبر لغات من طرف العميل لأن المعالجة الخاصة بها تحدث على جهاز المستخدم النهائي. أما عندما تقوم بزيارة صفحة ويب مكتوبة بلغة PHP أو أي لغة أخرى من طرف العميل ففي هذه الحالة تتم أولاً عملية معالجة كود PHP بالكامل على خادم الويب البعيد، ثم يتم الحصول على نتيجة هذه المعالجة بشكل كود HTML وبعدها يقوم الخادم البعيد بإرسال هذه الصفحة الناتجة إلى متصفح المستخدم.

ثم يتلقى المستخدم كود HTML الذي يمثل صفحات ديناميكية تتضمن محتويات الموقع، ويتم عرض هذه الصفحات على متصفح الإنترنت الخاص بجهازه المحلي.



# ما معنى المواقع الثابتة والمواقع الديناميكية ؟

## المواقع الثابتة Static Websites

هي مواقع بسيطة تحتوي على محتوى ثابت، حيث يرى كل مستخدم نفس المحتوى في كل صفحة ولا تتغير الصفحة استجابة لإجراءات وتفاعل المستخدم. لا يتم تحديث محتوى الموقع إلا عن طريق شخص لديه المعرفة عن كيفية تطوير المواقع وهو الشخص المبرمج للموقع. ومن أشهر الأمثلة على المواقع الثابتة مواقع السيرة الذاتية، المواقع الشخصية، والمواقع الغير هادفة للربح.

## المواقع الديناميكية أو المتغيرة Dynamic Websites

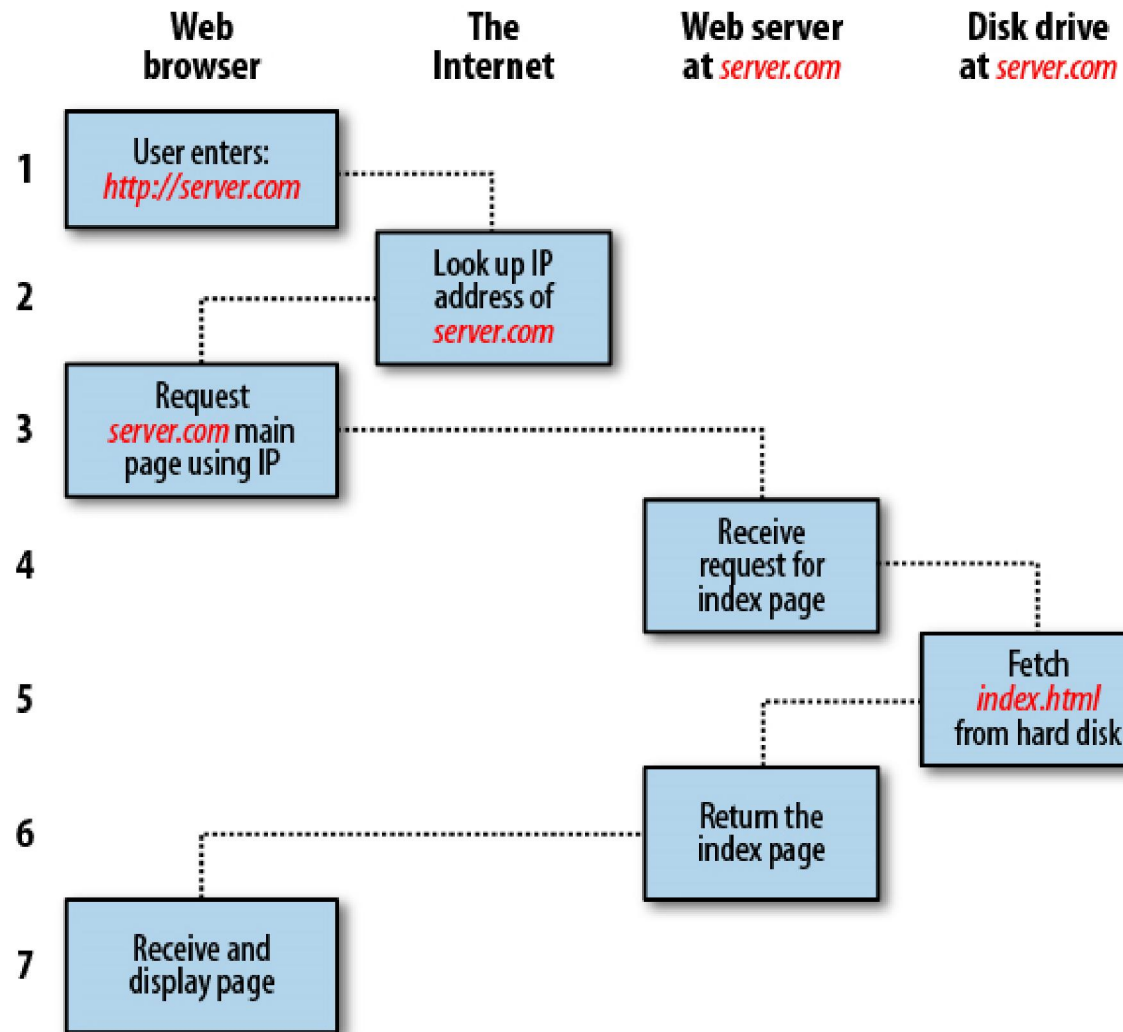
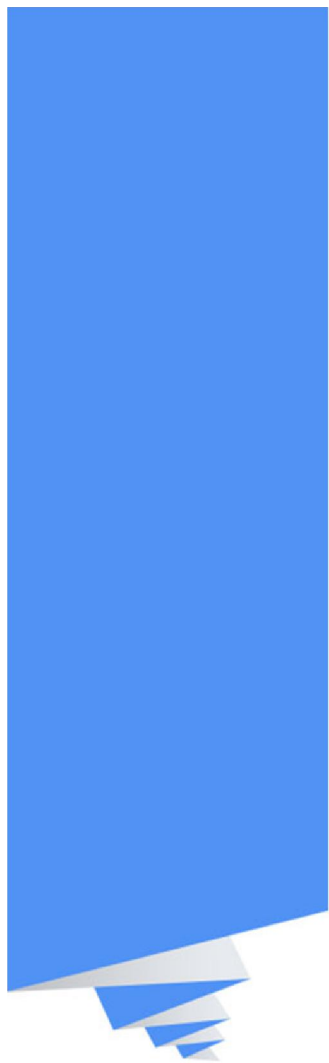
على عكس المواقع الثابتة فإن المواقع الديناميكية لا تعرض نفس المحتوى لجميع المستخدمين، المواقع الديناميكية تعرض محتوى مختلفا للزائر على حسب، موقع الزائر، التوقيت المحلي، الاهتمامات السابقة، الإجراءات التي اتخذها على الموقع، وغيرها. تعتبر هذه المواقع أكثر تطورا من المواقع الثابتة وأكثر تكلفة أيضاً، لكنها تعطيك القدرة على تحديث المحتوى، إضافة محتوى جديد، التحكم في نظام الإدارة، نظام الإعلانات، صلاحيات المستخدمين، بكل سهولة ودون الحاجة إلى خبرة برمجية مسبقة، ومن أشهر الأمثلة على المواقع الديناميكية مواقع التجارة الإلكترونية، مواقع التواصل الاجتماعي.

# The Request/Response Procedure

خطوات الطلب و الاستجابة request and response steps:

1. تقوم بإدخال `http://server.com` في شريط عنوان المتصفح الخاص بك.
2. يبحث متصفحك عن عنوان الـ IP الخاص بـ `server.com`.
3. يصدر متصفحك طلبًا للصفحة الرئيسية على `server.com`.
4. يعبر الطلب الإنترنت ويصل إلى خادم الويب `server.com`.
5. بعد أن يتلقى خادم الويب الطلب، يبحث عن صفحة الويب على القرص الثابت الخاص به.
6. يقوم الخادم باسترداد صفحة الويب وإعادتها إلى المتصفح.
7. يعرض متصفحك صفحة الويب.





# مراجعة سريعة للغة HTML

مثال لتوضيح تركيبية ملف الـ HTML

```
<html>
<head>
<title> عنوان الصفحة </title>
</head>
<body bgcolor="#0000CC" >
هذه هي أول صفحة نقوم بتصميمها.
<b>
ما رأيكم في هذا النص العريض ؟
</b>
</body>
</html>
```

# تابع مراجعة سريعة للغة HTML

من خلال المثال السابق نجد مجموعة من وسوم الـ HTML سنقوم بشرحها الآن حيث نجد أن ملف الـ HTML يجب أن يبدأ بالوسم `<html >` فهذا الوسوم يخبر المتصفح هذا هو بداية ملف الـ HTML، كما يجب أن ينتهي الملف بالوسم `</html >` وهذا الوسوم يخبر المتصفح بأن هذه هي نهاية ملف الـ HTML.

و تنقسم ملفات الـ HTML إلى جزئيين رئيسيين هما جزء الـ `head` أو ما يسمى برأس الملف و جزء الـ `body` أو ما يسمى بجسم الملف.

➤ رأس الملف الـ `head`

الجزء المحصور بين الوسمين `<head >` و `</head >` يسمى رأس الصفحة ، و النص الموجود بين هذين الوسمين هو عبارة عن معلومات رأس الصفحة ، و تلك المعلومات لا يتم عرضها داخل نافذة المتصفح.

أما النص الموجود بين الوسمين `<title >` و `</title >` هو عبارة عن عنوان الصفحة ، هذا العنوان يتم عرضه عنوان المتصفح " هو الشريط الافقي الموجود أعلى شاشة المتصفح ".

# تابع مراجعة سريعة للغة HTML

➤ جسم الملف الـ body

الجزء الموجود بين الوسمين `< body >` و `</ body >` هو ما يسمى بجسم الملف ، و النص الموجود بين هذين الوسمين هو ما يعرض داخل نافذة المتصفح . غالبا ما نستخدم عناصر الـ HTML – الوسوم Tags- لعرض العناصر داخل الصفحة بشكل مرتب وجميل.

➤ وسوم الـ HTML العادية تكتب بشكل مزدوج كما بالشكل التالي:

**`</b >` محتوى العنصر `< b >`**

ملاحظات مهمة:

- \* الوسم الأول هو وسم بداية الكود، و الوسم الثاني هو وسم إغلاق الكود.
- \* النص الموجود بين وسم البداية و وسم الإغلاق هو محتويات العنصر.
- \* هناك بعض وسوم الـ HTML التي لا تتطلب وسم نهاية ومحتوى تسمى بالعنصر الفارغ empty element، أي ليس له علامة إغلاق close tag، مثل `<hr >` و `<br >` و `<img >`.
- \* وسوم الـ HTML غير حساسة لحالة الأحرف، بمعنى أن الوسم `< b >` هو ذاته الوسم `< B >`

# تابع مراجعة سريعة للغة HTML

## ➤ سمات الوسم Tag Attributes

الوسوم يمكن أن تحتوي على سمات، تلك السمات تزودنا بمعلومات إضافية حول عناصر الـ HTML داخل الصفحة. تكتب داخل وسم البداية للعنصر كما بالشكل التالي:

```
<body bgcolor="#0000CC" >
```

ملاحظات مهمة:

- \* قيمة السمة دائماً يجب أحاطتها بأقواس الاقتباس " علامات التنصيص " أقواس الاقتباس المزدوجة هي الأكثر شيوعاً، و لكن هذا لا يمنع استخدام الأقواس المفردة، وفي بعض الحالات النادرة مثل أن تكون قيمة السمة تحتوي على علامة اقتباس يكون من الضروري استخدام علامات الاقتباس المفردة.
- \* يمكن أن يتم كتابة سمة واحدة أو أكثر، أو قد لا يتم كتابة أي سمة، وتختلف السمات التي يمكن كتابتها باختلاف العنصر، ولكن توجد بعض السمات العامة التي يمكن كتابتها في جميع العناصر.
- \* كما أن بعض السمات اختيارية وبعضها الآخر إجباري، أي بعضها تتم كتابته عند الحاجة، وبعضها الآخر يجب كتابته كي يعمل العنصر بشكل صحيح، مثل السمة src داخل العنصر <img>.

# مراجعة سريعة للغة CSS

بشكل عام لغة الـ CSS تعتبر لغة تصميم Style Language و ليس لغة برمجة Programming Language فهي تستخدم في تصميم صفحات الويب التي يتم بناؤها بواسطة HTML أو الصفحات التي يتم بناؤها بواسطة XML.

يتم اضافة كود الـ CSS بهدف تحسين طريقة ظهور العناصر الموجودة في صفحة الويب و التي تم إضافتها بالأساس بواسطة وسوم الـ HTML من هذا المنطلق فإنه عند كتابة كود CSS يجب تحديد الوسم المراد التعديل على تصميمه و من ثم تمرير الخصائص و القيم المناسبة له.

الكثير من سمات CSS تشبه تلك المستخدمة في HTML، لذلك إذا تعلمت HTML واستخدمتها لإنشاء التصاميم فأنت في الغالب ستتمكن من تعلم CSS بسهولة.

يتألف كود لغة الـ CSS من القواعد Rules وهي تقابل العناصر Elements في لغة HTML، ويكون شكل القاعدة في CSS كما يلي:

```
HTML Element {  
    Property: Value; }
```

# تابع مراجعة سريعة للغة CSS

أين تكتب كود الـ CSS

توجد ثلاث أساليب يمكنك إتباعها حتى تضيف كود الـ CSS في صفحات الويب:

## 1. أسلوب Inline CSS

أي وسم تضيفه في الصفحة يمكنك إضافة السمة `style` له و تمرير كود الـ CSS لها كقيمة بشكل مباشر.

```
<h1 style="color:blue; text-align:center;">Hello World!</h1>
```

## 2. أسلوب Internal CSS

يمكنك إضافة الوسم `< style >` في وسم رأس الصفحة و بداخله تستطيع إضافة خصائص CSS لأي وسم موجود في الصفحة.

```
<style>
h1 { color: blue;
text-align: center; }
</style>
```

# تابع مراجعة سريعة للغة CSS

## 3. أسلوب External CSS

يمكنك وضع كود الـ CSS بداخل ملف إمتداده .css و من ثم تضمينه في الصفحة بواسطة الوسم `< link >` و الذي يجب وضعه في محتوى الوسم `< head >`

```
style.css
h1 {
  color: blue;
  text-align: center;
}

index.html
<!DOCTYPE html>
<html>
  <head>
    <!-- style.css الملف بتضمين الملف -->
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>Hello World!</h1>
  </body>
</html>
```



# تابع مراجعة سريعة للغة CSS

كيفية تحديد عناصر الـ HTML

1. تحديد عناصر HTML بناءً على اسم العنصر.

```
p {  
  text-align: center;  
  color: red;  
}
```

2. تحديد عناصر HTML بناءً على المعرفات ID's

معرف العنصر فريد داخل الصفحة، لذلك يتم استخدام محدد المعرف لتحديد عنصر فريد واحد فقط، لتحديد عنصر بمعرف محدد يتم كتابة رمز (#) متبوعاً بمعرف العنصر.

سيتم تطبيق تصميم CSS أدناه على عنصر HTML المتبوع بالمعرف (id="para"):

```
#para {  
  text-align: center;  
  color: red;  
}
```

# تابع مراجعة سريعة للغة CSS

## 3. تحديد عناصر HTML بناءً على الفئات Classes

لتحديد عنصر بفئة محددة يتم كتابة رمز (.) متبوعاً بمعرف العنصر. يمكن أن نستخدم نفس اسم الفئة لأكثر من عنصر، كما يمكن أن نستخدم أكثر من فئة لعنصر واحد.

سيتم تطبيق تصميم CSS أدناه على عنصر HTML المتبوع بالمعرف (class="center"):

```
.center {  
  text-align: center;  
  color: red;  
}
```

يمكنك أيضاً تحديد أن عناصر HTML المحددة فقط هي التي يجب أن تتأثر بالفئة.

```
p.center {  
  text-align: center;  
  color: red;  
}
```

# تابع مراجعة سريعة للغة CSS

يمكن أن تتأثر عناصر HTML أيضًا بأكثر من فئة واحدة.

```
<p class="center large">This paragraph refers to two classes.</p>
```

4. المحدد العام لـ CSS

يقوم المحدد العام (\*) بتحديد كافة عناصر HTML الموجودة في الصفحة.

```
* {  
  text-align: center;  
  color: blue;  
}
```

5. محدد تجميع CSS

يقوم محدد التجميع بتحديد كافة عناصر HTML التي لها نفس التصميم.

```
h1, h2, p {  
  text-align: center;  
  color: red;  
}
```