



## PHP 05

# Cadenes

## **8)Cadenes**

- **Cadenes de caràcters** es poden definir indistintament:
  - Entre cometes dobles.
  - Entre cometes simples.

- Exemple:

```
$cadena = "IES Simarro";
```

o també

```
$cadena = 'IES Simarro';
```

- Per al tractament d'aquestes variables o cadenes:
  - Funcions bàsiques de manipulació.
    - `echo`
    - `strlen` / `mb_strlen`
    - `substr`
    - `str_word_count`
    - `strrev`
    - `strpos`
    - `str_replace`
    - `explode`
    - `strtolower`
    - `strtoupper`
  - `.` (concatenació)
- Veurem les més bàsiques i d'ús més comú.
  - Hi ha moltes **funcions de manipulació de cadenes en PHP.**
  - Algunes les veurem més endavant.



- **echo no és una funció** és una construcció de PHP per poder incrustar codi php dins de HTML
- Es pot utilitzar:
  - Amb i sense parèntesi.
  - Amb cometes dobles i cometes simples.
- Amb cometes **simples**, les **variables** i **seqüències d'escapament** per a caràcters especials **no seran expandides** en l'ordre **echo**. Per **exemple**:

```
$cadena = 'IES Simarro';  
echo 'Valor de la cadena: $cadena';
```

Resultat:

**Valor de la cadena: \$cadena**

- En canvi, si la cadena està tancada entre cometes dobles ("):
  - PHP substituirà els valors de les variables.
- Per **exemple**:

```
$cadena = 'IES Simarro';  
echo "Valor de la cadena: $cadena";
```

Resultat:

Valor de la cadena: IES Simarro

- A més, interpretarà, entre altres, els següents **caràcters especials**:

Seqüència	Significat
\n	nova línia (LF or 0x0A (10) en ASCII)
\r	retorn de carro (CR or 0x0D (13) en ASCII)
\t	tabulació horitzontal (HT or 0x09 (9) en ASCII)
\v	tabulació vertical (VT or 0x0B (11) in ASCII) (des de PHP 5.2.5)
\e	escapament (ESC or 0x1B (27) in ASCII) (des de PHP 5.4.4)
\f	avanç de pàgina (form feed) (FF or 0x0C (12) in ASCII) (des de PHP 5.2.5)
\\	barra invertida
\\$	símbol de dòlar
\"	cometes dobles

**Funció strlen:** strlen(string \$cadena): int

- Retorna un **nombre enter** amb la **longitud de la cadena** que se li passa com a paràmetre.
- Obtenim el **nombre de caràcters** que formen la **cadena completa**.

```
<?php
$cad="cadena";
$n=strlen($cad);
echo "La longitud de la cadena és: $n"; // Mostra 6
?>
```



- Si la cadena conté algun **caràcter que no pertany** a l'ASCII original:
  - El comptaria com a **doble**. Per **exemple**:
    - `strlen("adéu")` és 5, no 4 com esperaríem.
- Per comptar els caràcters segons el **joc de caràcters que estem utilitzant** en la pàgina, **millor** utilitzar `mb_strlen(string $cadena, ?string $codificacio = null): int`
- Perquè realment es puguin utilitzar les funcions `mb_`:
  - Cal instal·lar prèviament un paquet en ubuntu (exemple versió 8.3 de php):  
`$ sudo apt install php8.3-mbstring`

- Si la cadena conté algun **caràcter que no pertany** a l'ASCII original:
  - El comptaria com a **doble**. Per **exemple**:
    - `strlen("adéu")` és 5, no 4 com esperaríem.
- Per comptar els caràcters segons el **joc de caràcters que estem utilitzant** en la pàgina, **millor** utilitzar `mb_strlen(string $cadena, ?string $codificacio = null): int`

<?php

```
$cad = "adéu";
```

```
$n = mb_strlen ($cad);
```

```
echo "La longitud de la cadena és: $n"; // Mostra 4
```

?>

**Funció substr:** `substr(string $string, int $offset, ?int $length = null): string`

- Retorna una subcadena de l'original (una part de l'original).
- Aquesta funció té **3 paràmetres**:
  - La **variable que conté la cadena**.
  - Nombre **enter** que representa la **posició** (inclusivament) en la qual **comença la subcadena**. Posició d'**inici de la subcadena dins de la cadena**.
  - Nombre **enter** que indica la **quantitat de caràcters** que seran extrets.
    - Aquest argument és **opcional**.
    - Si **s'omet** s'extraurà la cadena des de la posició d'inici indicada **fins al final**.

- **Important**: numeració de la posició dels caràcters comença per zero.
- Per **exemple** per la cadena 'GPS', que té 3 caràcters:
  - El caràcter en la **posició 0** : G
  - El caràcter en la **posició 1** : P
  - El caràcter en la **posició 2** : S
  - **No hi ha la posició 3** a causa de que es comença a comptar pel zero.

- Exemple 02:

```
<?php
    $cad="cadena";
    $sub1=substr($cad, 2); // des del caràcter 2 fins al final
    $sub2=substr($cad, 2, 2); // des del caràcter 2, dos caràcters
    echo "La subcadena número u és la següent: $sub1 <br/>";
    echo "La subcadena número dos és la següent: $sub2";
?>
```

- Resultat:

La subcadena número u és la següent: dena

La subcadena número dos és la següent: de

- Cal fixar-se, en utilitzar **substr**:
  - La **numeració** de caràcters **comença en zero** i no en u.
  - Així en la cadena **"IES Lluís Simarro"**
    - El caràcter 0 és la I.
    - El caràcter 1 és la E.
    - El caràcter 2 és la S.
    - El caràcter 3 és l'espai en blanc i així successivament.
  - La **posició en què comença l'extracció va inclosa** en la subcadena extreta.
  - **substr(\$cad,0,1)** : només extreu el primer caràcter de la cadena.

- **Exemple 03:**

<?php

```
$cad="IES Lluís Simarro" ;  
$sub1=substr($cad,0 ); // La cadena completa  
$sub2=substr($cad,0,3 ); // La subcadena IES  
$sub3=substr($cad,4,5); // La subcadena Lluís  
$sub4=substr($cad,10); // La subcadena Simarro  
echo "La subcadena número u és la següent: $sub1<br/>";  
echo "La subcadena número dos és la següent: $sub2<br/>";  
echo "La subcadena número tres és la següent: $sub3<br/>";  
echo "La subcadena número quatre és la següent: $sub4";
```

?>

## Operador de concatenació: .

- **Concatenació**: unir **diverses porcions de text** en **una sola cadena**.
- Aquest **operador** s'expressa com un **punt** '.'.
- Per **exemple**:

```
<?php
    $inici = "Entorns ";
    $final = "de Desenvolupament";
    $completa = $inici.$final; //Entorns de Desenvolupament
?>
```



## Exemple 04:

```
<?php
    $cad1 = "Carpe" ;
    $cad2 = "Diem" ;
    $cadtotal = $cad1.$cad2;
    echo "La cadena 1 concatenada amb la cadena 2 és: $cadtotal <br/>";
    echo "Exemple: " . ($cad2.$cad1); // Concatenem Exemple: amb $cad2 i $cad1
?>
```

- Resultat:

La cadena 1 concatenada amb la cadena 2 és: CarpeDiem

Exemple: DiemCarpe

## Funció str\_word\_count:

`str_word_count(string $cadena, int $format = 0, ?string $characters = null): array|int`

- Compta la quantitat de **paraules que hi ha en una cadena** que se li passa com a **paràmetre**.
- Per defecte el **separador** entre paraules és el **caràcter d'espai**.
- Per **exemple**:

```
<?php
    echo str_word_count("Hello world!"); // Mostra 2
    echo str_word_count("A E I O U"); // Mostra 5
?>
```

**Funció strrev:** `strrev(string $cadena): string`

- **Inverteix el contingut de la cadena** que se li passa com a paràmetre.
- Per **exemple:**

```
<?php
    echo strrev("Hello world!"); // Mostra: !dlrow olleH
?>
```

**Funció strpos:** strpos(string \$paller, string \$agulla, int \$offset = 0): int|false

- Busca un text específic dins d'una cadena.
- Si **es troba** una coincidència:
  - Retorna **enter** amb la **posició de la primera coincidència**.
- Si no es troba cap coincidència:
  - Retorna **false: compte**, perquè la posició 0 pot assimilar-se a aquest valor.
- Té dos **paràmetres**:
  - **Cadena** en la que anem **a buscar** (**\$paller**)
  - **Text** que volem **buscar dins de la cadena** del paràmetre **anterior** (**\$agulla**).



**Exemple:**

```
<?php
    echo strpos("Hello world!", "world"); //Mostra 6

    if (strpos("Hello world!", "Hola")==false){
        echo "La cadena no s'ha trobat"; // no troba la cadena
    }

    if (strpos("Hello world!", "Hello")==0){
        echo "La cadena s'ha trobat"; // troba la cadena en la pos. 0
    }
?>
```

## Funció str\_replace:

str\_replace( array|string \$busca, array|string \$reemplaça, string|array \$cadena, int &\$count = null): string|array

- **Reemplaça alguns caràcters amb alguns altres caràcters** dins d'una **cadena**.
- **Retorna una cadena o un array** amb totes les ocurrències.
- Té quatre **paràmetres**, en aquest ordre:
  - Primer: **\$busca** : cadena a substituir.
  - Segon: **\$reemplaça** : cadena que substituirà a l'anterior.
  - Tercer: **\$cadena** : cadena sobre la que es van a realitzar les operacions.
  - Quart: **\$count** : és opcional. S'hi ficarà la quantitat de reemplaçaments fets.

### Exemple:

```
<?php
```

```
    $cadena = str_replace("world", "Dolly", "Hello world!");  
    echo $cadena; // mostra Hello Dolly!
```

```
?>
```

```
<?php
```

```
    $cadena = str_replace("a", "", "Sargantana"); //elimina les a  
    echo $cadena; // mostra Srgntn
```

```
?>
```

**Funció explode:** `explode(string $separador, string $cadena, int $limit = PHP_INT_MAX): array`

- **Divideix una cadena** en funció d'**una altra cadena** :
  - Crea com a resultat un **array de cadenes**.
- Té tres **paràmetres**:
  - El **primer**: **\$separador** : el delimitador en funció del qual dividirem la cadena.
  - El **segon**: **\$cadena** : la cadena que volem dividir.
  - El **tercer**: **\$limit** : és opcional.
    - **\$limit** > 0 : l'array retornat tindrà com a màxim **\$limit** elements. L'últim element és la resta de la cadena.
    - **\$limit** < 0 : es tornen tots els components excepte els darrers **-\$limit**.
    - **\$limit** = 0 : es tracta com si fora 1.



**Exemple:**

- Dividim una cadena pels espais que continga (obtenim les paraules).

```
<?php
```

```
    $cadena = "blau roig verd groc taronja negre";
```

```
    $colors = explode(" ", $cadena);
```

```
    echo $colors[0]; //Mostra blau
```

```
    echo $colors[3]; //Mostra groc
```

```
?>
```

**Exemple:**

- Dividim una cadena pel caràcter "o" (que es lleva de les cadenes resultat)

```
<?php
```

```
    $cadena = "blau roig verd groc taronja negra";
```

```
    $colors = explode("o", $cadena);
```

```
    echo $colors[0]; //Mostra blau r
```

```
    echo $colors[3]; //Mostra nja negra
```

```
?>
```

**Funció strtolower:** strtolower(string \$string): string

- Converteix a **minúscules** el **contingut de la cadena** que se li passa com a paràmetre.
- Per **exemple**:

```
<?php
    $cadena = strtolower("IES Lluís Simarro");
    echo $cadena; //Mostra ies lluis simarro
?>
```

## Funció strtoupper: strtoupper(string \$string): string

- Converteix a **majúscules** el **contingut de la cadena** que se li passa com a paràmetre.
- Per **exemple**:

```
<?php
    $cadena = strtoupper("IES Lluís Simarro");
    echo $cadena; //Mostra IES LLUÍS SIMARRO
?>
```