

Welcome To Computer Science HUB



മലയാളത്തിൽ

String functions

**strlen|strpos| strstr| strcmp| substr|
str_replace|string case**

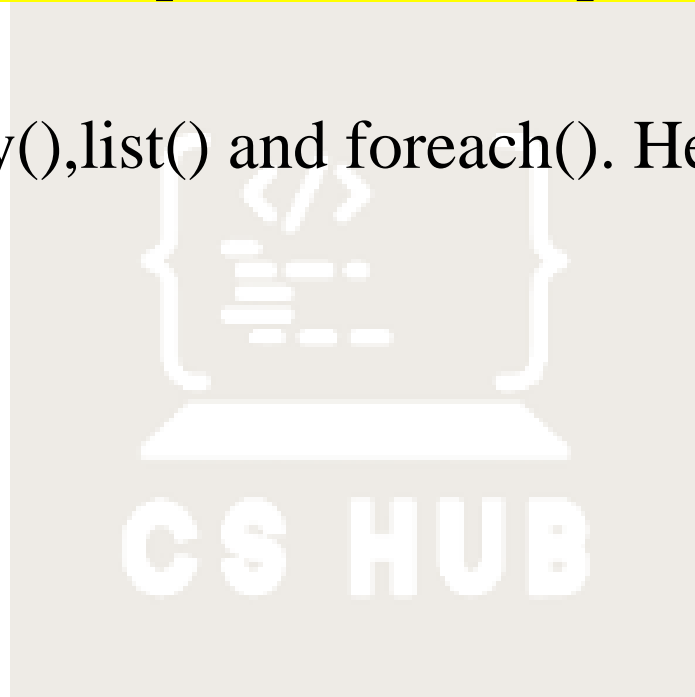


CS HUB

BCA5B09 | BCS5B09-Web Programming using PHP

Unit IV [10 T + 10 L]

- Working with PHP: Passing information between pages
- HTTP GET and POST method, Cookie, Session.
- String functions: strlen, strpos, strstr, strcmp, substr, str_replace, string case
- Array constructs: array(),list() and foreach(). Header().



String functions

- `$my_string="hello";`
- `strlen()` //returns length of string including white spaces

- `<?php`

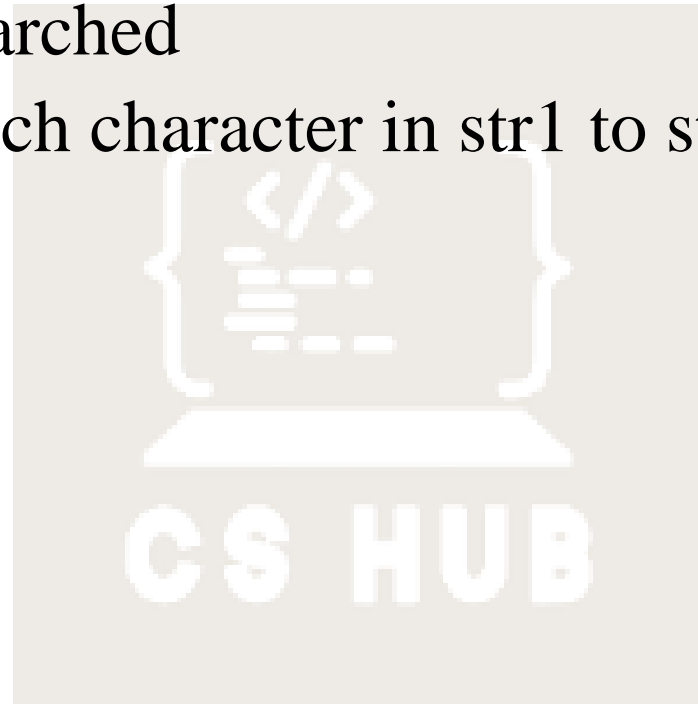
```
$str1='abcdef';  
echo "strlen($str1).<br>" //6  
$str2="ab cd ";  
echo strlen($str2)."<br>"; //6  
echo $str1[strlen($str1)-1]; //f
```

?>



String functions

- **strpos()**- finds position of first occurrence of a string
- **Syntax:** strpos(\$str1,\$str2[,offset=0])
 - **\$str1** :string to search in
 - **\$str2**:item to be searched
 - **Offset:** specify which character in str1 to start searching



String functions

■ <?php

```
$mystring='abc';
```

```
$findme='a';
```

```
$pos=strpos($mystring,$findme);
```

```
if($pos===false)
```

```
    echo "the string $findme is not found in the string $mystring"
```

```
else
```

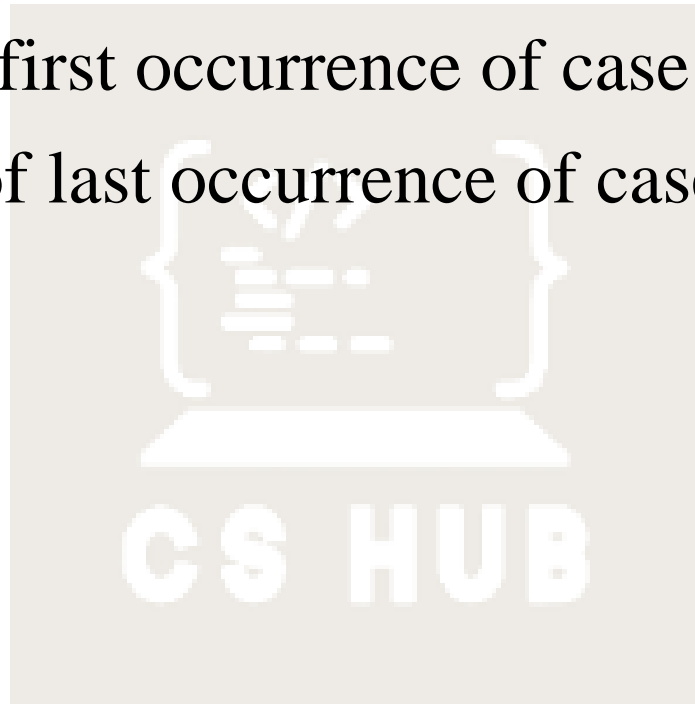
```
    echo "the string is found at the position $pos";
```

```
?>
```



String functions

- **strrpos()**-position of last occurrence of substring in a string
 - `$mystring='abcaba';`
`$findme='a';`
`$pos=strrpos($mystring,$findme); // o/p:5`
- **strpos()**-position of first occurrence of case insensitive string
- **strripos()**- position of last occurrence of case insensitive string



String functions

- **strstr()-find first occurrence of a string**
- **Syntax:** strstr(\$str1,\$str2[, \$pos=false])
- Returns part of \$str1 from the first occurrence of \$str2 to the end of \$str1
- If \$pos is true it returns part of str1 before the first occurrence of the \$pos
- **<?php**

```
$email='name@example.com';  
$domain=strstr($email,'@');  
echo $domain;    // @example.com  
$user=strstr($email,'@',true);  
echo $user;      //name
```

?>



String functions

- `stristr()`- case insensitive
- `substr()`- returns part of a string
 - **`substr($string, $start[$length])`**
- Returns the portion of `$string` specified by `$ start` and `$length` parameters
 - `<?php`
 `echo substr('abcdef',1)."
"; //bcdef`
 `echo substr('abcdef',1,3)."
"; //bcd`
 `echo substr('abcdef',-1,1)."
"; //f`
 `?>`



String functions

- **strcmp()- string comparison**
 - strcmp(str1,str2);
 - Returns <0 if str1<str2
 - >0 , str1>str2
 - =0, str1=str2
- **strcasecmp()- case insensitive string comparison**
- **strncmp()- string comparison of first n characters**
 - strncmp(\$str1,\$str2,\$len)
- **strncasecmp()-**
- **trim()- strips whitespace from the beginning and end of a string**
- **ltrim()-from beginning**
- **rtrim()- from end**



String functions

- **str_replace**- replaces all occurrences of the search string with the replacement string
- **str_replace (\$searchVal, \$replaceVal, \$subjectVal, \$count)**
- **\$searchVal**: This parameter can be of both string and array types. It specifies the string to be searched and replaced.
- **\$replaceVal**: This can be of both string and array types. It specifies the string with which we want to replace the \$searchVal string.
- **\$subjectVal**: string or array being searched and replaced
- **\$count**: This parameter is optional and if passed, its value will be set to the total number of replacement operations performed on the string.
- ```
<?php
 echo str_replace("world","Peter","Hello world!");
?>

//o/p:Hello Peter
```



# String functions

- **str\_replace**
- **Rules**
- If the **\$search** and the **\$replace** arguments are arrays, then all the elements of the \$search argument are searched in the \$subject string and replaced by the corresponding elements in the \$replace argument.
- `$subject = "You eat fruits, vegetables, fibre every day."`
- `$search = array("fruits", "vegetables", "fibre")`
- `$replace = array("pizza", "burger", "ice cream")`
- `str_replace($array1, $array2, $subject);`
- o/p: You eat pizza, burger,ice cream every day.



# String functions

- **str\_replace**

- **Rules**

- If number of elements in \$replace is less than that in \$search array, then if there are any occurrences of the additional elements of \$search argument in the \$subject argument then they will be replaced by an empty string.

- `$find = array("Hello","world");`

`$replace = array("B");`

`$arr = array("Hello","world","!");`

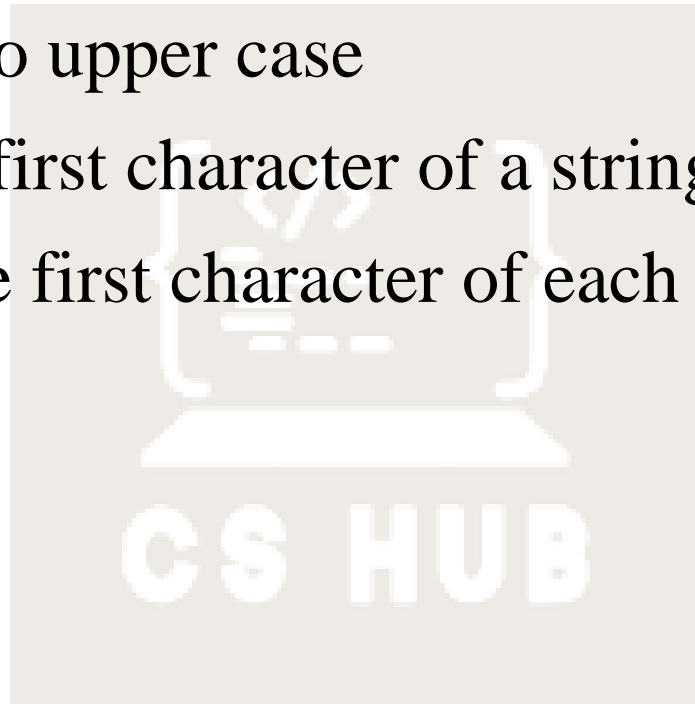
`print_r(str_replace($find,$replace,$arr));` //B !

- If the \$subject parameter is also an array instead of string then all of the elements of \$subject will be searched.



# String functions

- `str_repeat()`- repeats a string
- `str_tolower()`- convert to lowercase
- `strtoupper()`-convert to upper case
- `ucfirst()`- convert the first character of a string to uppercase
- `ucwords()`-convert the first character of each word in a string to uppercase



**Thank You !!**

