preg_match

The **preg_match()** function in PHP is used to perform a regular expression match. It searches a given string for a pattern and returns true if the pattern is found, false otherwise. Here's a detailed overview:

Syntax

int preg_match (string $pattern, string $subject, array &$matches = null, int $flags = 0, int $offset = 0)

Parameters

- **$pattern :** The regular expression pattern to search for.

- **$subject :** The input string to search within.

- **$matches :** (Optional) An array to store the matches. If provided, $matches[0] will contain the full match, $matches[1] will contain the first captured group, and so on.

- **$flags :** (Optional) A set of options that change how the matches array is structured. Common flags include PREG_OFFSET_CAPTURE and PREG_UNMATCHED_AS_NULL.

- **$offset :** (Optional) The position in the input string to start the search.

Return Values

- **1** : If the pattern matches the subject.

- **0** : If the pattern does not match the subject.

- **false** : If an error occurs[1].

Example

Here's an example of using preg_match() to find a pattern in a string:

```php
<?php
        $pattern = '/w3schools/i'; // Case-insensitive search for "w3schools"
        $subject = 'Visit W3Schools';
        if (preg_match($pattern, $subject, $matches)) {
                echo "Match found: " . $matches[0];
        } else {
```

```php
        echo "No match found.";

    }

?>
```

Explanation

In this example, preg_match() searches for the pattern /w3schools/i in the string "Visit W3Schools"[2]. Since the pattern is found, it prints "Match found: W3Schools".

Flags

- **PREG_OFFSET_CAPTURE** : Returns the position of the match in the input string[1].

- **PREG_UNMATCHED_AS_NULL** : Unmatched subpatterns are reported as null instead of an empty string[1].

Example with Flags

```php
<?php

    $pattern = '/(w3schools)/i';

    $subject = 'Visit W3Schools';

    preg_match($pattern, $subject, $matches, PREG_OFFSET_CAPTURE);

    print_r($matches);

?>
```

Output

```
Array
(
  [0] => Array
    (
      [0] => W3Schools
      [1] => 6
    )
)
```

In this example, PREG_OFFSET_CAPTURE is used to get the position of the match.