GABRIELE ROMANATO

Introduzione al Test-Driven Development (TDD) in PHP

Il Test-Driven Development (TDD) è una metodologia di sviluppo software che prevede la scrittura dei test prima del codice di produzione. Questo approccio aiuta a garantire la correttezza del codice e facilita la manutenzione a lungo termine. In PHP, il framework più comunemente utilizzato per scrivere test è PHPUnit.

Principi del TDD

Il ciclo TDD si basa su tre passaggi fondamentali:

- **Red:** scrivi un test che fallisce.
- **Green:** scrivi il codice minimo necessario per far passare il test.
- **Refactor:** migliora il codice mantenendo i test verdi.

Installazione di PHPUnit

Per iniziare, puoi installare PHPUnit tramite Composer:

```
composer require --dev phpunit/phpunit
```

Esempio pratico

Supponiamo di voler creare una classe `calculator` con un metodo `add`. Ecco come si potrebbe applicare il TDD:

1. Scriviamo il test

```
use PHPUnit\Framework\TestCase;
```

```
class CalculatorTest extends TestCase
{
    public function testAddReturnsSumOfTwoNumbers()
    {
        $calculator = new Calculator();
        $result = $calculator->add(2, 3);
        $this->assertEquals(5, $result);
    }
}
```

2. Implementiamo la classe

```
class Calculator
{
    public function add($a, $b)
    {
        return $a + $b;
    }
}
```

Vantaggi del TDD

- Codice più affidabile e meno soggetto a bug.
- Migliore progettazione del software.
- Facilità di refactoring grazie alla copertura dei test.

Conclusione

Adottare il TDD in PHP significa scrivere codice più pulito, sicuro e facilmente mantenibile. Iniziare richiede disciplina, ma i benefici a lungo termine superano lo sforzo iniziale.