Solution Review: Calculate the Student's Total Marks

This review provides a detailed analysis to solve the 'Calculate the Student's Total Marks' challenge.

WE'LL COVER THE FOLLOWING

Solution

```
class Student {
  //fields
  private string _name;
  private double _physicsMarks;
  private double _chemistryMarks;
  private double _biologyMarks;
  //properties
  public string Name {
    get {
      return this._name;
    }
  }

  public double PhysicsMarks {
    get {
      return this._physicsMarks;
    }
  }

  public double ChemistryMarks {
    get {
      return this._chemistryMarks;
    }
  }

  public double BiologyMarks {
    get {
      return this._biologyMarks;
    }
  }
  // Parameterized constructor
  public Student(string name, double phy, double chem, double bio) {
    this._name = name;
    this._physicsMarks = phy;
    this._chemistryMarks = chem;
```

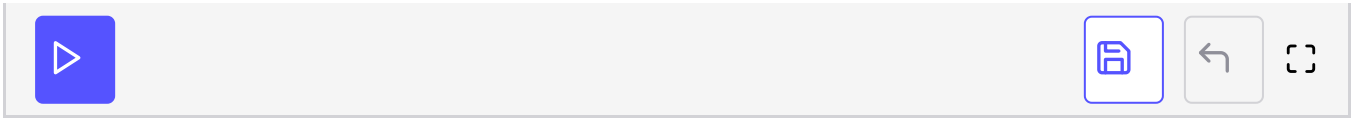
```
    this._biologyMarks = bio;
  }


  public double TotalObtained() {
    double totalMarks = PhysicsMarks + ChemistryMarks + BiologyMarks;
    return totalMarks;
  }

   public double Percentage() {
    return (TotalObtained()/300) * 100;
  }

}
class Demo {
  public static void Main(string[] args) {
    Student john = new Student("John", 75, 75, 90);
    Console.WriteLine("Total marks obtained: " + john.TotalObtained());
    Console.WriteLine("Percentage obtained:" + john.Percentage());
    Console.WriteLine("Physics Marks:" + john.PhysicsMarks);
  }

}
```



Explanation

- **Line 3-6:** Declared the four `private` class fields according to the naming conventions.
- **Line 8-24:** Defined `get` blocks in the respective properties.
- **Line 32-37:** Defined the parameterized constructor by assigning parameters to the respective fields.
- **Line 39-42:** Defined the `TotalObtained()` method inside the class.
- **Line 44-46:** Finally, defined the `Percentage()` method to return the calculated percentage.

In the next challenge, we'll solve another problem of implementing a calculator class.