Exemplo de SOLID em C#

Exemplo de SOLID em C#

Código de Exemplo

```
// Single Responsibility Principle (SRP)

public class Order

{

    public int Id { get; set; }

    public string Product { get; set; }

    public int Quantity { get; set; }

}


// Open/Closed Principle (OCP)

public interface IOrderProcessor

{

    void Process(Order order);

}


public class OnlineOrderProcessor : IOrderProcessor

{

    public void Process(Order order)

    {

        // Process online order

    }

}
```

```
public class OfflineOrderProcessor : IOrderProcessor
{
    public void Process(Order order)
    {
        // Process offline order
    }
}

// Liskov Substitution Principle (LSP)
public abstract class Discount
{
    public abstract double Calculate(double amount);
}

public class PercentageDiscount : Discount
{
    private double _percentage;

    public PercentageDiscount(double percentage)
    {
        _percentage = percentage;
    }

    public override double Calculate(double amount)
    {
        return amount - (amount * _percentage / 100);
    }
}

public class FixedDiscount : Discount
{
    private double _amount;

    public FixedDiscount(double amount)
    {
        _amount = amount;
    }

    public override double Calculate(double amount)
    {
        return amount - _amount;
    }
}

// Interface Segregation Principle (ISP)
public interface IOrderRepository
{
    void Save(Order order);
}
```

Exemplo de SOLID em C#

```
public interface IOrderNotifier

{

    void Notify(Order order);

}


public class OrderRepository : IOrderRepository

{

    public void Save(Order order)

    {

        // Save order to database

    }

}


public class EmailOrderNotifier : IOrderNotifier

{

    public void Notify(Order order)

    {

        // Send email notification

    }

}


// Dependency Inversion Principle (DIP)

public class OrderService

{

    private readonly IOrderRepository _orderRepository;
    private readonly IOrderNotifier _orderNotifier;

    public OrderService(IOrderRepository orderRepository, IOrderNotifier orderNotifier)
    {
        _orderRepository = orderRepository;
        _orderNotifier = orderNotifier;
    }

    public void PlaceOrder(Order order)
    {
        _orderRepository.Save(order);
        _orderNotifier.Notify(order);
    }
}
```

Exemplo de SOLID em C#

Descrição do Código

Descrição do Código

1. **SRP (Single Responsibility Principle)**: A classe `Order` tem apenas uma responsabilidade: representar um pedido.

2. **OCP (Open/Closed Principle)**: A interface `IOrderProcessor` permite que novos tipos de processadores de pedidos sejam adicionados sem modificar a classe existente.

3. **LSP (Liskov Substitution Principle)**: A classe base `Discount` permite substituições com classes derivadas (`PercentageDiscount`, `FixedDiscount`) que mantêm o comportamento esperado.

4. **ISP (Interface Segregation Principle)**: As interfaces `IOrderRepository` e `IOrderNotifier` dividem as responsabilidades para evitar a implementação de métodos que não são necessários.

5. **DIP (Dependency Inversion Principle)**: A classe `OrderService` depende de abstrações (`IOrderRepository`, `IOrderNotifier`) em vez de implementações concretas.