**DESIGN PATTERNS**

|  |
| --- |
| O que é design patterns? |

Design Patterns são soluções padrões ou padrões de projeto para problemas e aplicações comuns e recorrentes na programação. Diferentemente das funções e bibliotecas, você não pode apenas copiar e colar os design patterns, são como formulas de matemática, deve ser aplicar e adaptar ao problema da sua aplicação e projeto, como um conceito que serve como solução.

|  |
| --- |
| Quais seus benefícios? |

O principal benefício é a economia de tempo de trabalho, quando você escolhe caminhar por um lado em que uma pessoa já passou se torna mais fácil e objetivo o trabalho, possibilitando essa economia. Os design patterns são soluções de problemas que antes eram difíceis de serem resolvidas, até que alguém criou uma solução padrão que facilitou o trabalho de outros.

|  |
| --- |
| Design Patterns é dividido em categorias, quais? |

Creational Patterns (Criacionais): Criação de Objetos; Structural Patterns (Estruturais): Resolução de problemas na composição das classes e objetos; Behavior Patters (Comportamentais): Resolução de problemas na comunicação entre os objetos.

|  |
| --- |
| Por que utilizar design patterns em projetos? |

Pois possibilita melhor organização, melhor visibilidade na solução de problemas e celeridade aos projetos de software, além de disponibilizar a implementação de soluções personalizadas que se adptem a realidade de seus negócios.

|  |
| --- |
| No decorrer do curso, utilizamos quais design patterns? |

Singleton; Prototype.

|  |
| --- |
| Em qual categoria se encaixa o Singleton? |

Se encaixa na categoria dos criacionais, usados para criação de objetos.

|  |
| --- |
| Quais os problemas que o Singleton resolve? |

Ele garante que um classe tenha apenas uma instancia.

2MDS

Rhian Henrique Custódio, Victor Gabriel, Victor Hugo, Luis Gustavo.