Orientação a objetos

1 – Boas práticas de programação

“Maus cheiros no código”

- Código Duplicado
  - Número no ranking de “mau cheiro”.
  - Lógica de negócio duplicada
  - Torna díficil a manutenção do sistema.

- O que fazer?
  - Operações repetidas ou utilizadas muitas vezes devem ser colocadas em métodos.

- Métodos longooooooossssss...
  - Fazem coisas demais
  - Complicam a lógica
  - Tornam a manutenção díficil.

- O que fazer?
  - Quebrar em métodos menores (curtos)
  - Métodos devem fazer apenas uma “coisa” e somente uma “coisa”, nada mais.

- Classes muitooooo grannnnddeeeeessss...
  - Muitas responsabilidades.
  - Fazem “coisas” que outras classes deveriam estar fazendo.
  - Podem conter lógica duplicada ou inútil.

- O que fazer?
  - Quebrar em classes menores com menos responsabilidades.

- Lista de parâmetros muito loooongaaaaas....
  - Significam muitas dependências.

- O que fazer?
  - Transforme a lista em um Objeto (classe).

- Inveja dos dados
  - Um objeto busca demais dados em outro objeto para realizar sua função.

- O que fazer?
  - Coloque os dados na classe que realmente utiliza os dados.

- Classe ociosa
  - Essa não faz nada

- O que fazer?
  - Remova a classe ociosa e coloque seus atributos em uma super classe ou outra classe.

- Comando Switch
  - Duplicação de código

- O que fazer?
  - Utilizar polimorfismo

- Campo/variável temporária
  - Tornam o código díficil de ler e entender o que o método está fazendo.

- O que fazer?
  - Váriaveis devem ter um nome que signifique realmente o que elas fazem.

- Comentários
  - Desodorante.
  - Código muito ruim.
  - Nos levam a todos os cheiros podres mostrados.

- O que fazer?
  - Refatore o código.
  - Utilize nomes siginificativos.

- Manutenção
  - Melhoria e otimização de um sistema já desenvolvido.
  - Correção de bugs.
  - Engenharia de software.

  - Boas práticas e padrões = fácil manutenção

“Qualquer tolo pode escrever códigos que um computador entenda. Bons programadores escrevem códigos que humanos conseguem entender.” *Martin Fowler*