

Boas práticas de codificação:

Realizar comentários apenas se necessário, curtos e diretos. Evitar comentários inúteis que poluem o código.

Exemplo de comentário desnecessário:

```
//método que retorna verdadeiro caso encontre o desenhista  
public boolean consultarDesenhista(idDesenhista){  
    ...  
    return true;  
}
```

Nomes de variáveis, métodos, classes e tabelas no banco de dados, devem ser indicativos, claros e legíveis.

Exemplos de nomes indicativos:

```
private String assinatura;  
private String arquivo;  
public void alterarCasaArtistica;
```

Exemplos inconsistentes:

```
private String ass;  
private String arq;  
public void altCasaArt;
```

Utilizar chaves { } para definir bem os escopos, mesmo quando não é necessário para compilar.

Exemplo incorreto: `if(idProcurado == idDesenhista) return idDesenhista;`

Exemplo correto:

```
if(idProcurado == idDesenhista){  
    return idDesenhista;  
}
```

Utilizar o máximo de recursos prontos possíveis para evitar gasto de tempo em codificação de um problema já resolvido.

Exemplo: usar o atributo padrão num_rows para retornar o número de resultados encontrados em uma consulta sql.

Utilizar indentação correta. Definir um padrão de indentação (sempre por espaços ou sempre por tab).

Exemplo incorreto:

```
if(idProcurado == idDesenhista){  
    if(administrador){  
        return true;  
    }  
else {  
    return false;  
}  
}
```

Exemplo correto:

```
if(idProcurado == idDesenhista){  
    if(administrador){  
        return true;  
    }  
else {  
    return false;  
}  
}
```