Ferramenta de desenvolvimento: IDE Eclipse.

Idioma padrão do arquivo : Inglês.

**Nomes dos arquivos**

Nome da classe com inicial maiúscula respeitando a regra CamelCase.

Ex: NomeDaClasse.java

**Cabeçalho da classe**

Será composto de comentário no seguinte modelo. Cada linha deve começar com palavras escritas com inicial maiúscula:

/\*\*

Nome da classe

Breve descrição de sua função

Endereço da classe no GitHub

\*/

Todo arquivo deve ter uma linha entre os packages e imports.

package example;

import java.util.\*;

import java.another.import;

**Estrutura do arquivo**

Toda classe deve seguir a seguinte ordem de estrutura:

1. Cabeçalho
2. Declaração da classe (ou interface)
3. Variáveis estáticas, públicas, protegidas e privadas, respectivamente
4. Variáveis de instâncias, públicas, protegidas e privadas, respectivamente
5. Construtores
6. Métodos

**Identação**

Deve ser usado o Tab, sempre com largura de quatro espaços.

A quantidade de caracteres por linha deve ser menor ou igual a 80. A quebra de linha de linha deve ser feita sempre após a vírgula e antes do operador. Caso existam parênteses, deve-se evitar quebrar a expressão entre eles. A linha de continuação deve ter uma identação de 2 tabs (8 espaços).

**Comentários**

Comentários de bloco, quando necessários, devem seguir o modelo proposto na seção cabeçalho.

Comentários de uma linha devem ser feitos no seguinte modelo, espaço entre a segunda barra e a letra inicial maiúscula e ponto final.:

// Comentário de uma linha.

**Declarações**

Declarações de variáveis devem ser feitas em linhas separadas, ou seja, cada variável numa linha.

Declarações de funções, classes ou interfaces devem ser feitas com o parentese junto da última letra do nome (da função ou classe), sem espaço entre a próxima letra. Log em seguida vem tipo da variável, espaço, nome da variável, vírgula, espaço e assim sucessivamente. Chaves devem ser abertas na mesma linha e fechadas na mesma identação da função.

Ex: class Exemplo extends Object {

public Exemplo(int argumento1, int argumento2) {

}

}

**Estruturas de controle e/ou repetição**

Devem ter a declaração da palavra reservada, um espaço, o parentese, nome da variável, espaço, operador, espaço e assim sucessivamente, como no exemplo. Quando essas estruturas executarem apenas um comando, ou nenhum, deve deixar explícito abrindo as chaves e deixando o ‘interior’ com a linha de código específica ou comentário de uma linha:

Ex: for (int i = 0; i <= 10; i++) {

// Esse *for* não executa nada.

}

Estrutura de controle *if,* quando for composta de ter o próximo *if* dentro do else correspondente, como no exemplo:

Ex: if ( expressãoBooleana ) {

} else {

if ( outraExpressao ) {

} else {

// E assim sucessivamente.

}

}

**Espaços em branco**

Deixar uma linha em branco entre métodos e variáveis (ou bloco de variáveis).

Deixar linha em branco no começo de funções e métodos.

Ex: funcaoTal(){

// A função começa aqui.

}

//Alterações propostas por Parley:

Todas as variáveis ou atributos globais de uma classe devem ser acessadas utilizando-se o modificador *this.*

Ex: (this.variable = variable)

this.globalField = newValue.

Cada função deverá começar com um comentário, com uma breve descrição da função.

Ex: // This function modifies a client.

public void modifyClient(){

...

}