

コーディング標準

目的	Java のプログラムの実装をガイドする
ヘッダ	すべてのプログラムは説明的なヘッダで開始する.
ヘッダフォーマット	<pre> /***** プログラム課題: プログラム番号 名前: 名前 日付: プログラム開発開始日 説明: プログラムの簡単な説明と、その機能 *****/ </pre>
コンテンツリスト	コンテンツリストの概要を提供すること.
コンテンツリスト例	<pre> /***** コンテンツリスト: 再利用の説明 修正の説明 コンパイルの説明 クラス宣言: -Program1 -Calculator 実装メソッド -main() -calcSum() *****/ </pre>
再利用の説明	<ul style="list-style-type: none"> プログラムの使用方法を記述する。宣言の形式、パラメータの値と型、パラメータの制限等。 不正な値、オーバーフロー状態、または不適切な操作につながる可能性のあるその他の状態に関する警告の提供
再利用の説明例	<pre> /***** 再利用の説明 int printLine(String line) 目的:String 型文字列 line を 1 行で印刷する 制約:文字列の長さが LINE_LENGTH 以下であること 戻り値:印刷可能であれば 1, 不可能であれば *****/ </pre>
識別子	<p>すべての変数,関数,定数,その他識別子には説明的な名前を使用すること。</p> <p>略語や 1 文字は避ける。</p>
識別子例	<pre> int numberOfStudents 良い例 int x4, j, ftave 悪い例 </pre>

コメント	<ul style="list-style-type: none"> ● その動作を理解できるように文書化すること ● コメントは、コードの目的と動作の両方を説明する必要がある ● 目的を明示するために、変数宣言にコメントを入れる
良いコメント	<code>If(recordCount > limit) //すべてのレコードが処理されたか？</code>
悪いコメント	<code>If(recordCount > limit) //recordCount が limit を超えているかチェックする</code>
主要なセクション	主要なセクションの前にそのセクションが行う処理を記述するコメントを示す
主要なセクション例	<pre> /***** このセクションでは、配列「grades」の内容を調べ、クラスの平均点を計算する。 *****/ </pre>
空白スペース	<ul style="list-style-type: none"> ● 爪件だように見えないように十分スペースを入れる ● 構成要素(変数,定数など)を少なくとも 1つのスペースで区切る.
インデント	<ul style="list-style-type: none"> ● すべての中括弧のレベル毎に 1段階インデントを入れる ● 中括弧の開始は前の行の末尾に、終了は単独の行に配置し開始の行の先頭と横軸をそろえる。
インデント例	<pre> while (missDistance > threshold) { successCode = moveRobot(targetLocation); if (successCode == MOVE_FAILED) { System.out.println("The robot move has failed."); } } </pre>
大文字化	<ul style="list-style-type: none"> ● 定義はアッパースネークケース. ● その他の識別子や予約語はキャメルケース ● 読みやすさのためにユーザーメッセージは混ぜていい
大文字化例	<code>public static final int MAX_SCORE = 100;</code>