

Programista zawodowy nigdy nie pracuje sam. Jego kod nie powinien więc wywoływać frustracji u innego programisty, który zacznie korzystać z tego co napisał. Jako, że problem jest powszechny postanowiono stworzyć całe nauki (Clean Code) na temat stosowania dobrych praktyk programistycznych, które powodują, że kod:

- jest czytelny i rozumiany przez wszystkich z zespołu,
- łatwy do utrzymania i modyfikowania,
- ryzyko wystąpienia błędu jest małe, a jeśli takowe posiada - łatwo go wykryć.

Ważną zasadą jest spójność kodu w kontekście nazewnictwa różnych elementów języka. Dzięki temu osoba, która pierwszy raz widzi dany kod jest w stanie szybko zrozumieć co jest w nim napisane. W Javie przyjęto następującą konwencję:

Element języka	Zasada tworzenia	Przykłady
klasy	Nazwy klas to rzeczowniki zapisane za pomocą UpperCamelCase – każde słowo rozpoczyna się wielką literą.	<code>class Raster;</code> <code>class ImageSprite;</code>
metody	Nazwy metod to czasowniki zapisane za pomocą lowerCamelCase, albo wyrażenia, które zaczynają się od czasownika zapisanego z małej litery, a każde kolejne słowa jest zaczyna się z wielką literą.	<code>run();</code> <code>runFast();</code> <code>getBackground();</code>
zmienne	Zmienne lokalne, zmienne obiektu, pola klasy, są zapisywane za lowerCamelCase. Nie powinny one zaczynać się od podkreślnika ( <code>_</code> ), albo znaku dolara ( <code>\$</code> ), być może znasz taki zapis z np. z C++, w Javie go nie stosujemy. Nazwy zmiennych powinny być krótkie, ale tłumaczące się. Powinny zdradzać intencje tzn. programista, który zapoznaje się z projektem po przeczytaniu nazwy powinien wiedzieć do czego ta zmienna służy. Nazwy zmiennych w postaci jednej litery nie powinny być stosowane, wyjątkowo, można ich używać do tymczasowych zmiennych (np. zakres iterowania w pętli <code>for</code> ). Zwyczajowo zmienne typu <code>int</code> nazywamy kolejno <code>i</code> , <code>j</code> , <code>k</code> , <code>m</code> ..., a zmienne typu <code>char</code> <code>c</code> , <code>d</code> , <code>e</code> ...	<code>int i;</code>  <code>String surname;</code>  <code>float myWidth;</code>
stałe	Nazwy stałych powinny być zapisane dużymi literami, a wyrazy w nice powinny być oddzielone podkreślikami. Jeśli w nazwie musisz użyć liczby, to pamiętaj, by nie była ona na początku nazwy.	<code>static final int MAX_PARTICIPANTS = 10;</code>
pakiety	Nazwy pakietów piszemy małymi literami, zwyczajowo nazwa pakietu jest nazwą domeny, ale odwracamy kolejność poszczególnych członów.	<code>pl.uksw.kursjavy</code>

Na koniec, przeczytaj ze zrozumieniem poniższe artykuły:

<https://javadeveloper.pl/solid/>

<https://www.samouczekprogramisty.pl/solid-czyli-dobre-praktyki-w-programowaniu-obiektow-ym/>

Jakie są zasady wchodzą w skład SOLID? Czym się charakteryzują?