

PROGRAMOWANIE – WYTYCZNE CZYLI JAK PISAĆ ZAJEBISTY KOD ;)

Spis treści

NAZWY.....	3
NAZWY ZMIENNYCH.....	3
KOMENTARZE	3

Nazwy

Profesjoniści piszą kod zrozumiały dla innych

Naszym celem powinno być uczynienie kodu tak łatwego do zrozumienia jak to tylko możliwe

1. Twórz nazwy, które można wymówić
2. Używaj nazw przedstawiających intencje (nazywaj zmienną, funkcję, klasę etc. wg. jej przeznaczenia i tego co ma robić) - **Jeśli nazwa wymaga komentarza, znaczy, że nie ilustruje intencji**
3. Unikaj dezinformacji:
 - a. nazw, które coś znaczą np. accountList – jeśli nie jest to obiekt listy to czemu nie nazwać tego accounts
 - b. nazw, które się mało od siebie różnią – np. zmienne account i accounts w obrębie jednego obiektu/zasięgu
4. Nie używaj typów w nazwach zmiennych np. nameString, userObject czy usersTable jest błędne – name, user, users – poprawne
5. Korzystaj z nazw łatwych do wyszukiwania
6. Nie używaj przedrostków
- ~~7. Nie używaj tego samego słowa dla dwóch różnych celów np. jeśli nazwa add metody tworzy nową wartość przez połączenie dwóch innych to nie powinniśmy jej używać do wstawienia parametru do kolekcji, zamiast tego możemy nazwać ją insert~~
- ~~8. Korzystaj z nazw dziedziny rozwiązania np. jeśli korzystasz z wzorca Factory do stworzenia obiektu użytkownika to powinieneś zdefiniować nazwę klasy UserFactory~~
9. **Krótsze nazwy są lepsze niż dłuższe o ile są one jasne**
10. **Długa opisowa nazwa jest lepsza niż krótka enigmatyczna**
11. **Czas poświęcony na wybór odpowiedniej nazwy jest dobrą inwestycją**

Nazwy zmiennych

1. Używaj jednoliterowych nazw zmiennych tylko tam gdzie ich zakres pozostaje bardzo mały i jest zrozumiały dla kontekstu np. liczniki w pętlach takie jak i,j,k

Komentarze

Czytelny kod nie wymaga komentarzy!

Dobre komentarze:

1. Komentarze dla dokumentacji API np. @param, @return
2. Komentarze prawne – Copyright na początku dokumentu
3. Wyjaśnienie zamierzeń fragmentu kodu
4. Ostrzeżenie o konsekwencjach
5. Komentarze TODO