



NAUKA PROGRAMOWANIA W JAVIE

WSKAZÓWKI

NAUCZ SIĘ ANGIELSKIEGO

- czytanie i pisanie dokumentacji technicznej
- czytanie dokumentacji API w JavaDoc
- czytanie specyfikacji (języka Java, a także systemów, które w przyszłości będziesz implementować)
- komunikacja z współpracownikami

CZYTAJ O PROGRAMOWANIU

- odwiedzaj blogi programistyczne, portale, czytaj magazyny
- wybierz najbardziej interesującą dziedzinę i skup się na niej
- sprawdzaj to, co czytasz, odkrywaj, badaj - zanurz się w temacie

NIE CZYTAJ / NIE OGLĄDAJ SAMOUCZKÓW

- dają złudne wrażenie posiadania wiedzy
- wprowadzają przekonanie, że już „umiesz” / „znasz” zagadnienie
- jeśli lubisz oglądać filmiki, to nie oglądaj „tutoriali”, lecz wykłady wyjaśniające trudne kwestie bądź uczące podstawowych zasad tworzenia oprogramowania

ZRÓB WŁASNY PROJEKT

- zaplanuj, co chcesz napisać (oto [lista inspiracji](#))
- wyszukuj, poznawaj technologie, rozwiązuj problemy w sposób ciągły, zawsze mając na uwadze ostateczny cel: wykonanie zaplanowanego programu
- nie porzucaj projektu, gdy zrobi się zbyt skomplikowany
- naucz się dzielić kod (metody, klasy, pakiety, moduły, biblioteki)
- poznawaj [wzorce projektowe oraz dobre praktyki](#)

ZNAJDŹ STRONĘ INTERNETOWĄ SWOJEGO JĘZYKA

Zaglądaj na nią regularnie.

- [API biblioteki standardowej Javy](#)
- [dokumentacja i narzędzia](#) do Javy
- [system modułów](#)
- [specyfikacje](#)

POZNAJ NAJWAŻNIEJSZE BIBLIOTEKI

- testowanie: [Mockito](#) (12.6K), [JUnit 4](#) (8.3K), lub [JUnit 5](#) (5K)
- rozszerzenie biblioteki standardowej: ogromny zbiór [Apache Commons](#), [Guava](#)
- logowanie ([Logback](#), [Log4j](#))
- praca z formatem JSON: [Jackson](#) or [Gson](#), [FastJson](#), [Moshi](#)
- serializacja / deserializacja: [Kyro](#), [XStream](#) (xml)

WYBIERZ SYSTEM BUDOWANIA

- mój ulubiony to [Maven](#)
- nowszy i bardzo popularny to [Gradle](#), choć są jeszcze inne, trochę bardziej egzotyczne, jak stary dobry [Ant](#)
- jest też nowszy i przeznaczony raczej do projektów pisanych w [Clojure Leiningen](#)
- a także [SBT](#) dla projektów w języku [Scala](#)
- w język [Groovy](#) wbudowany jest [Grape](#) jako system zarządzania zależnościami

WYBIERZ SYSTEM KONTROLI WERSJI – GIT

W zasadzie nie masz wyboru.

Używany dziś powszechnie [Git](#) jest bezkonkurencyjny. Po prostu naucz się go ([książka](#), [ściąga](#)) i załóż sobie konto na [GitHubie](#) albo [GitLabie](#), a później:

- spróbuj utworzyć nowy projekt
- sklonuj go lokalnie
- zrób kilka komitów, wypuszuj
- baw się przy tym dobrze

STWÓRZ SYSTEM NAUKI

Pamiętaj o sprzężeniu:

działanie -> rezultat -> motywacja

- określ czas nauki (codziennie od 20:00 do 21:00)
- określ miejsce nauki (stół w kuchni / biblioteka / salon)
- zdefiniuj kamienie milowe, np.
 - a. w tym tygodniu wybieram edytor do programowania w Javie
 - b. do końca przyszłego miesiąca napiszę kalkulator BMI

- c. za pół roku napiszę aplikację w Java Swing
- d. za rok zacznę naukę SpringBoot
- e. za dwa lata zdam certyfikat Oracle

PORADY KOŃCOWE

Wykształć u siebie takie cechy jak:

- ciekawość
- cierpliwość
- nieustępliwość w dążeniu do celu
- spokojne przyjmowanie porażek

a poza tym:

- wysypiaj się
- jedz zdrowo
- codziennie wyprowadź się na półgodzinny spacer