

# Spis treści

1. JAVA.....	1
2. SQL i NoSQL.....	1
3. Frameworks (SPRING, HIBERNATE).....	1
3. OTHERS.....	2

## 1. JAVA

### Kurs Java dla początkujących

<https://nullpointerexception.pl/kurs-java-dla-poczatkujacych-1-wprowadzenie/>

### Zastosowanie Stream API z Java 8

<https://justjoin.it/blog/zastosowanie-stream-api-z-java-8-przyklady>

### Wielowątkowość - Future i CompletableFuture

<https://www.youtube.com/watch?v=xlkSa44Mdz8>

### Wątki w języku Java (Thread and Runnable)

<https://www.samouczekprogramisty.pl/watki-w-jezyku-java/>

### Algorytm fizzbuzz

<https://stormit.pl/fizzbuzz/>

### Try-with-resources

<https://www.samouczekprogramisty.pl/konstrukcja-try-with-resources-w-jezyku-java/>

### Jakie znamy wzorce do współbieżności w Javie

<https://howtointerview.pl/pytania-rekrutacyjne/jakie-znamy-wzorce-do-wspolbieznosci-w-javie/10488/>

### Interfejsy funkcyjne w Javie

<https://javappa.com/kurs-java/java8/przeglad-interfejsow-funkcyjnych>

### Garbage Collector – zarządzanie pamięcią

<https://softwareskill.pl/jak-dziala-garbage-collector-zarządzanie-pamiecia-jvm>

<https://softwareskill.pl/obszary-pamieci-maszyny-wirtualnej-javy-jvm>

### Co to jest interfejs mapy Java (Map<Key,Value>)

<https://codegym.cc/pl/groups/posts/pl.221.interfejs-mapy-w-javie>

### TreeMap - drzewa czerwono - czarne

<https://codegym.cc/pl/quests/lectures/pl.questcollections.level06.lecture07>

<https://codegym.cc/pl/groups/posts/pl.297.treemap-w-javie>

### HashMap – bardzo dokładnie

<https://cupofcodes.pl/co-powinienes-wiedziec-o-hash-map/>

### Zmiany Java 8 do Java 11

<https://nullpointerexception.pl/zmiany-w-javie-od-wersji-8-do-java-11/>

### JAVA 12

<https://nullpointerexception.pl/java-12-nowosci/>

## 2. SQL i NoSQL

### SQL Bez Tajemnic - Pytania i Odpowiedzi

<https://datacraze.pl/sql-bez-tajemnic-pytania-rekrutacyjne>

### SQL vs NoSQL

<https://www.ovhcloud.com/pl/learn/sql-vs-nosql/>

### Indeks w bazie danych co to jest i jak działa

<https://www.youtube.com/watch?v=PKVvn5oP4p8>

### Dlaczego indeks przyspiesza wykonywanie zapytań SQL?

<https://www.youtube.com/watch?v=U0Gxv9GU46o>

### Rodzaje indeksów w bazie danych Oracle

<https://www.youtube.com/watch?v=MEI3uBPTjJk>

### Jak przyspieszyć zapytanie SQL, czyli ścieżki dostępu do danych

<https://www.youtube.com/watch?v=Pz9vFdnSsKk>

### Zarządzanie Dostępem: Pessimistic vs Optimistic Locking

<https://www.youtube.com/watch?v=SD7vLt-v-hY>

## 3. Frameworks (SPRING, HIBERNATE)

### Pessimistic vs Optimistic Locking in Spring Boot

<https://beratyesbek.medium.com/pessimistic-and-optimistic-locking-in-jpa-spring-boot-aa59f2b906fa>

### Spring Framework

<https://javappa.com/kurs-spring/spring-framework>

### Spring Boot | ResponseEntity vs Object -> When to Use ResponseEntity<?> ?

<https://medium.com/@rahul302619/spring-boot-responseentity-vs-object-d530695a8481>

### TransactionManager Spring Boot

<https://medium.com/@Gotit.GotIT/mastering-transactions-in-spring-boot-with-platformtransactionmanager-a-practical-guide-4f51bb9e8e42>

<https://www.youtube.com/watch?v=wDMMD2pYNmg>

### Hibernate – najpopularniejsze pytania z rozmów kwalifikacyjnych

<https://codebykris.com/hibernate-najpopularniejsze-pytania/>

### Zakres Beanów w Spring Boot

<https://javappa.com/kurs-spring/zakresy-beanow>

### Jak działa rest controller w springu - głębiej

<https://medium.com/@viranthrocky/how-the-restcontrollers-are-working-internally-in-spring-boot-487c3ed370b0>

### Cała Prawda o Spring IoC Containers

<https://www.youtube.com/watch?v=pYoKGbf8fm8>

<https://bykowski.pl/spring-context-jak-dziala-kontener-ioc/>

### **Hibernate: Sequence vs Identity**

<https://www.geeksforgeeks.org/sql-server/difference-between-sequence-and-identity-in-sql-server/>

### **Kurs Spring Boot Mateusz Dąbrowski**

<https://www.youtube.com/playlist?list=PLLIGVI2WVN6ugud2cc3OShwWoTt65jzSL>

### **Kurs Spring Boot #08 - Spring Cache**

<https://www.youtube.com/watch?v=IWv3uBLO2LU&list=PLLIGVI2WVN6ugud2cc3OShwWoTt65jzSL&index=10>

### **Spring AOP - Implementacja aspektów**

<https://www.youtube.com/watch?v=vVwtKYdpSec&t=1539s>

<https://nullpointerexception.pl/jak-korzystac-ze-spring-aop/>

<https://reflectoring.io/aop-spring/>

### **Hibernate i JPA różnica:**

<https://clockworkjava.pl/2020/11/jpa-orm-jpa-i-hibernate/>

### **Spring Data JPA - projekcja danych w zapytaniach dynamicznych**

<https://walczak.it/pl/blog/spring-data-jpa-projekcja-danych-zapytaniach-dynamicznych>

### **Spring Boot: N+1 Problem**

<https://marcinszewczyk.pl/problem-n-plus-1/>

<https://medium.com/@ayoubseddiki132/what-is-the-n-1-query-problem-a56e39774cf7>

<https://nullpointerexception.pl/hibernate-i-problem-n-plus-1-zapytan/>

### **The best way to use JOIN FETCH and Pagination with Spring**

<https://vladmihalcea.com/join-fetch-pagination-spring/>

### **How Projections Affect Pagination in Spring Data JPA**

<http://medium.com/@neharathig/how-projections-affect-pagination-in-spring-data-jpa-c1ecd8475e53>

### **Spring, Micronaut czy Quarkus?**

<https://bulldogjob.pl/readme/spring-micronaut-czy-quarkus>

### **JWT token**

<https://nofluffjobs.com/pl/log/praca-w-it/poczatki-w-it/jwt-token-co-to-jest-jak-go-uzywac-w-autoryzacji/>

## **3. OTHERS**

### **Spring Boot Microservices with RabbitMQ Example:**

<https://rameshfadatare.medium.com/spring-boot-microservices-with-rabbitmq-example-92a38cbe08fc>

<https://medium.com/@sachinmewar800/rabbitmq-with-spring-amqp-using-spring-boot-4a62e68a6f1b>

<https://github.com/mzerek/RabbitMqExample/tree/main>

### **Jak przygotować się do rekrutacji na programistę:**

<https://www.youtube.com/watch?v=6Se64Wg1qQQ>

### **Co powinien umieć senior?**

<https://www.youtube.com/watch?v=tdgWeF7XVyw>

### **Servlety w aplikacjach webowych**

<https://www.samouczekprogramisty.pl/servlety-w-aplikacjach-webowych/>

### **Wzorce Projektowe - Observer**

[https://www.youtube.com/watch?v=gFevma4a\\_qQ](https://www.youtube.com/watch?v=gFevma4a_qQ)

### **Przemysław Bykowski**

<https://github.com/bykowski>

## **Wzorce projektowe**

<https://refactoring.guru/pl/design-patterns>

## **10 wzorców projektowych, które musisz znać**

<https://www.youtube.com/watch?v=pWP7Vm6OHpc>

## **Wzorzec dekorator**

<https://www.samouczekprogramisty.pl/wzorzec-projektowy-dekorator/>

## **[Kurs Dockera #03] Docker Networks – Jak działają sieci w Dockerze?**

<https://www.youtube.com/watch?v=-XvpLnWFMaI>

## **[Kurs Dockera #04] Docker Ports – Mapowanie i konfiguracja portów**

<https://www.youtube.com/watch?v=j5Lj6XkEk70>

## **[Kurs Dockera #05] Docker Volumes – Jak zarządzać danymi w kontenerach?**

<https://www.youtube.com/watch?v=SqMyER2GN5s>

## **[Kurs Dockera #06] Docker Compose – Sieci i konfiguracja krok po kroku**

<https://www.youtube.com/watch?v=5iPq4YtkLkM>

## **[Kurs Dockera #07] Docker Compose – Ports i użycie w developmentcie**

<https://www.youtube.com/watch?v=thQ8hWvYa00>

## **[Kurs Dockera #08] Jak tworzyć własne obrazy Dockerowe - Podstawy pracy z Dockerfile**

<https://www.youtube.com/watch?v=ji8nC2z3wpg>

## **[Kurs Dockera #09] Dockerfile Multistage Build – Optymalizacja obrazów**

<https://www.youtube.com/watch?v=La9BN8MAFRk>

## **Bykowski blog**

<https://bykowski.pl/blog/>

## **Microservice Orchestration vs Choreography – what's the difference?**

<https://camunda.com/blog/2023/02/orchestration-vs-choreography/>

## **Circuit Breaker Pattern (Bezpiecznik)**

<https://softwareskill.pl/circuit-breaker-pattern>

## **Jak nie zgubić zdarzenia, czyli wzorce Outbox i Inbox**

[https://oskar-dudycz.netlify.app/en/jak\\_nie\\_zgubic\\_zdarzenia\\_czyli\\_outbox\\_i\\_inbox\\_w\\_praktyce/](https://oskar-dudycz.netlify.app/en/jak_nie_zgubic_zdarzenia_czyli_outbox_i_inbox_w_praktyce/)

## **Komunikacja między mikroserwisami**

<https://grzegorzdziedzic.pl/2020/07/19/komunikacja-miedzy-mikroserwisami/>

## **Apache Kafka - opis działania i zastosowania**

<https://bulldogjob.pl/readme/apache-kafka-opis-dzialania-i-zastosowania>

example: <https://github.com/dsu/kafka-streams-redis-lambda-poc/tree/main>

example: <https://github.com/dsu/kafka-docker-subscription/tree/master>

kurs: <https://www.youtube.com/watch?v=pOh9I8ecFt0&list=PLEQXPnKOvPGQplxSFmpPy7-whZiw13Yox>

[https://www.youtube.com/watch?v=nk11\\_2dnvqc&list=PLEQXPnKOvPGQplxSFmpPy7-](https://www.youtube.com/watch?v=nk11_2dnvqc&list=PLEQXPnKOvPGQplxSFmpPy7-whZiw13Yox&index=10)

[whZiw13Yox&index=10](https://www.youtube.com/watch?v=nk11_2dnvqc&list=PLEQXPnKOvPGQplxSFmpPy7-whZiw13Yox&index=10)

**Oskar Polak:** <https://www.youtube.com/@oskarpolak4766>

## **Apache Kafka + Avro + Schema Registry**

<https://softwareskill.pl/apache-kafka-avro-schema-registry>

<https://www.baeldung.com/spring-cloud-stream-kafka-avro-confluent>

<https://medium.com/ing-tech-romania/implementing-a-basic-kafka-producer-and-consumer-using-spring-boot-spring-kafka-and-avro-schema-2b6d06e6c4cf>

**OWASP – co to jest?**

<https://semcore.pl/co-to-jest-owasp-wyjasnienie-i-omowienie-owasp-top-10/>  
<https://owasp.org/www-project-top-ten/>

**SonarQube**

<https://stormit.pl/sonarqube/>  
<https://www.baeldung.com/sonar-qube>

**Nexus**

<https://bykowski.pl/nexus-menager-repozytoriow-ktory-przyniosl-ulge-niejednej-organizacji/>

**DevEnd – różne info**

<https://www.youtube.com/@DevEnv-pl>

**DDD -> Model Domenowy vs ORM (Aggregate vs ORM)**

<https://www.youtube.com/watch?v=DLLVvkOG0OE&list=PLO5muTI694A7BKlnShyeAi58kFTna9awT&index=3>

**DDD**

<https://boringowl.io/blog/domain-driven-design-wprowadzenie-i-praktyczne-zastosowanie>

**Idempotency**

<https://medium.com/@connectmadhukar/idempotency-patterns-when-stream-processing-messages-3df44637b6af>

**Architektura wzorca porty i adaptery**

<https://nearshore-it.eu/pl/artykuly/porty-i-adaptery-w-praktyce/>