

JAVA programavimo kalba

Java srautai (streams)

Naujos sąsajos

- `Functions<T, R>` - operacija (funkcija), kuri turi vieną parametą tipo `T` ir grąžina rezultatą tipo `R`.
- `Predicate<T>` - operacija (funkcija), kuri turi vieną parametą tipo `T` ir grąžina boolean tipo rezultatą.
- `Supplier<T>` - operacija (funkcija), kuri neturi parametų ir grąžina `T` tipo rezultatą.
- `Consumer<T>` - operacija (funkcija), kuri turi vieną parametą tipo `T` ir nieko negrąžina.
- `Optional<T>` - operacija kuri grąžina `T` tipo objektą, jei jis ne null

Srautai (Streams)

- Kolekcijų karkasas (framework), pradedant java 8 versija, turi naują “dalykėlį”. Kuris puikiai dera su lambda išraiškomis ir yra vadinamas kolekcijų srautu (stream).
- Srautai skirti aprašyti ir vykdyti veiksmus su kolekcijos elementais.
- Srautus galima įsivaizduoti kaip konvejerį arba vamzdyną (pipeline), per kurį juda mūsų kolekcijos elementai ir tam tikrose vietose su tais elementais atliekamos tam tikros operacijos - elementai keičiami (map), atrenkami (filter), rūšiuojami (sort) ir vėliau, jei reikia, surenkami į konteinerį (collect).
- Srauto operaciją galima įsivaizduoti kaip veiksmą, kuris įėjime gauna vieną elementų rinkinį ir pagal kažkokią taisyklę sugeneruoja išėjime kitų ar tų pačių elementų kitą rinkinį.
- Veiksmai gali būti vykdomi nuosekliai (vienas elementas po kito) arba lygiagrečiai (keli elementai vienu metu).

Srautai (Streams)

Paprasčiausi būdai gauti srautą:

- `Arrays.stream(...)`
- *kolekcija*.stream()..., *kolekcija*.parallelStream()...
- `Stream.of(...)`
- `IntStream`, `DoubleStream`....
- `new Random().ins(...)`

Srautų operacijos

Yra dviejų rūšių srautų operacijos:

- **tarpinė** - jos rezultatas yra kitas srautas (filter, sort, map, reduce, ...)
- **galutinė** - ji arba neturi rezultato (void) arba rezultatas nėra srautas (forEach, sum, ifPresent, ...).

Reikalavimai srautų operacijoms

Srautų operacijoms keliami šie reikalavimai, jos turi būti:

- **nejsimenamos (stateless)** - jos rezultatas nepriklauso nuo ankstesnių rezultatų.
- **nesikišanti (non interference)** - ji nemodifikuoja, t.y. nesikiša į pradinį srautą.
- **be šalutinių poveikių (side-effects)** – rezultatas nepriklauso nuo jokių kitų pašalinių aplinkybių.
- Pastaba: operacijos gali ir nesilaikyti šių reikalavimų, bet tada rezultatas gali būti nenusakomas.

Lygiuotieji/Nelygiuotieji srautai

- Srautai gali būti sulygiuoti (ordered) arba ne (unordered)
- Jei srautas nelygiuotas, tai rezultato tvarka negarantuota, t.y. gali būti bet kokia.

Tarpinės operacijos

- filter - filtravimas - tarpinė operacija, kurios parametras Predicate funkcijos tipo. Praleidžiami tik tie srauto elementai, kuriems funkcija grąžina true.
- map - tarpinė operacija, kurios parametras Function tipo. Funkcija vykdoma kiekvienam srauto elementui ir grąžinamas naujas srautas.
- sorted - tarpinė operacija kuri lygiuoja srauto elementus.

Galutinės operacijos

- `forEach` - kiekvienam elementui atliekama funkcija
- `reduce` - mažinimo galutinė operacija grąžinanti `Optional`
 - `reduce(BinaryOperator accumulator)`
 - `reduce(identity, BinaryOperator accumulator)`
 - `reduce(identity, BiFunction accumulator, BinaryOperator combiner)`
- Pastaba: trečia forma veikia lygiagretaus srauto atveju. Kitu atveju ji reikia kaip antra.

Galutinės operacijos

collect - iš srauto sukuria kolekciją:

- `Collectors.toList()`
- `Collectors.toSet()`
- `Collectors.toMap()`
- `Collectors.groupingBy()`
- `Collectors.reducing()`
- `Collectors.mapping()`

Nuorodos

- <https://docs.oracle.com/javase/8/docs/api/java/util/stream/package-summary.html>
- <https://docs.oracle.com/javase/tutorial/collections/streams/>
- <http://www.oracle.com/technetwork/articles/java/ma14-javase-8-streams-2177646.html>
- <http://www.oracle.com/technetwork/articles/java/architectstreams-pt2-2227132.html>
- <http://winterbe.com/posts/2014/07/31/java8-stream-tutorialexamples/>

Uždaviniai

Tarkime turime objektą Employee ir objektą Salary, kuriame yra Employee tipo laukas ir Double tipo laukas reiškiantis išmokėtą atlyginimą.

Sukurkite sąrašą darbuotojų ir jų atlyginimų (gali būti keli to pačio darbuotojo įrašai) ir aprašykite tokius veiksmus naudojant duomenų srautus/vamzdynus:

1. Išrinkite kiek ir kokiam darbuotojui yra išmokėta ir išveskite mažėjimo tvarka pagal išmokėtą sumą: vardas - išmokėta suma.
2. Suskaičiuokite kiek išmokėjimų (ne suma bet kiek kartų buvo išmokėta) yra kiekvienam darbuotojui ir atspausdinkite: vardas - kiek kartų išmokėta.