

Funkcje

Kilka pojęć

- **Funkcja** = reużywalna porcja kodu
- **Metoda** = w praktyce to samo, co funkcja (tyle, że zwykle mówi się o metodach klasy – czyli reużywalnych porcjach kodu mających dostęp do zmiennych klasy)
- **Biblioteka** = zestaw funkcji udostępniony publicznie przez mądrzejszych programistów
- **Framework** = biblioteka + zestaw konwencji i dobrych praktyk

Zadanie

Generator imion i nazwisk

Czyli dokładnie co?

- Imię to krótszy String, nazwisko – dłuższy
- Określ długość każdego z nich
- Korzystaj tylko z liter
- Wykorzystaj istniejącą bibliotekę do generowania
- Nadaj Stringom właściwy format – Imię Nazwisko
- Wydrukuj wygenerowane dane na ekran

Czyli dokładnie co?

- **Imię to krótszy String, nazwisko - dłuższy**
- Określ długość każdego z nich
- Korzystaj tylko z liter
- **Wykorzystaj istniejącą bibliotekę do generowania**
- Nadaj Stringom właściwy format – Imię Nazwisko
- Wydrukuj wygenerowane dane na ekran

Istniejąca biblioteka?

show 3 more comments

467

If you're happy to use Apache classes, you could use `org.apache.commons.text.RandomStringGenerator` (commons-text).

Example:

```
RandomStringGenerator randomStringGenerator =  
    new RandomStringGenerator.Builder()  
        .withinRange('0', 'z')  
        .filteredBy(CharacterPredicates.LETTERS, CharacterPredicates.DIGITS)  
        .build();  
randomStringGenerator.generate(12); // toUpperCase() if you want
```

Since commons-lang 3.6, `RandomStringUtils` is deprecated.

share improve this answer

edited Oct 18 '17 at 7:58

community wiki

10 revs, 9 users 42%

numéro6

21 Has just looked through **mentioned class** of `Apache Commons Lang 3.3.1` library - and it is using only `java.util.Random` to provide random sequences, so it is producing **insecure sequences**. – Yura Apr 3 '14 at 14:51

15 Make sure you use `SecureRandom` when using `RandomStringUtils`: `public static java.lang.String random(int count, int start, int end, boolean letters, boolean numbers, @Nullable char[] chars, java.util.Random random)` – Ruslans Uralovs Mar 3 '15 at 13:28

add a comment

https://mvnrepository.com

[Home](#) » [org.apache.commons](#) » [commons-text](#) » 1.6



Apache Commons Text » 1.6

Apache Commons Text is a library focused on algorithms working on strings.

License	Apache 2.0
Categories	String Utilities
HomePage	http://commons.apache.org/proper/commons-text
Date	(Oct 13, 2018)
Files	jar (192 KB) View All
Repositories	Central
Used By	828 artifacts

[Maven](#)

[Gradle](#)

[SBT](#)

[Ivy](#)

[Grape](#)

[Leiningen](#)

[Buildr](#)

```
<!-- https://mvnrepository.com/artifact/org.apache.commons/commons-text -->
<dependency>
  <groupId>org.apache.commons</groupId>
  <artifactId>commons-text</artifactId>
  <version>1.6</version>
</dependency>
```

☒ Include comment with link to declaration

I co z tym zrobić?

- Znajdź plik pom.xml
- Otwórz go
- Przed tagiem `</project>` wpisz:
 `<dependencies>`
 `</dependencies>`
- A między te znaczniki wklej tekst skopiowany z przeglądarki
- Maven projects need to be imported → Import changes
 - Lub Maven Project → Reimport All Maven Projects

Suuuuper, to jak z tego skorzystać?

- Dokumentacja to potęga!
- <http://commons.apache.org/proper/commons-text/apidocs/index.html>
- <http://commons.apache.org/proper/commons-text/apidocs/org/apache/commons/text/RandomStringGenerator.html>

public vs private

- Public – czyli funkcja, którą będzie widać z każdego miejsca po zaimportowaniu
- Private – czyli funkcja, która jest widoczna tylko wewnątrz danej klasy

void

- Funkcja void nie zwraca nic – robi rzeczy i tyle. Efekty jej działania widzimy, ale nie możemy ich w zasadzie użyć
- Bez słowa kluczowego return

Funkcje zwracające wartość

- Funkcja najczęściej jednak zwracają wartość – w przypadku naszego zadania, chcemy wytworzyć String i użyć go potem – np. do wydrukowania na ekran albo do zapisania do pliku, przekazania do testu itd.
- Zawierają słowo kluczowe return

Deklaracja

- **public void generateFirstnameAndLastnameAndDisplayIt**
 - **public String generateFirstnameAndLastname**
 - **public String generateFirstname(int length)**
-
- **Dostęp**
 - **Zwracany typ**
 - **Nazwa**
 - **Typ przyjmowanego parametru**
 - **Nazwa przyjmowanego parametru**

static

- Klasy i funkcje statyczne nie potrzebują instancji klasy, by można je było wywołać.
- W praktyce – funkcje i klasy statyczne są wieczne i zawsze dostępne
- W praktyce – często używane do funkcji pomocniczych tzw. helperów