

# Programowanie w języku JAVA

## Zajęcia laboratoryjne 3

### 1. Zadania do wykonania

**Zadanie 0.** Pobierz JavaFx SDK z adresu: <https://gluonhq.com/products/javafx/>. Wybierz wersję 12. (Strona z dokumentacją <https://openjfx.io/openjfx-docs/>). Rozpakuj zawartość. Przykładowa ścieżka do wykorzystania: C:\JAVAFX\

Pobierz również program do wizualizacji i szybkiej modyfikacji layoutu aplikacji. <https://gluonhq.com/products/scene-builder/#download>

**Zadanie 1.** Stwórz nowy projekt w IntelliJ Idea. (File -> New -> Project). Z listy po lewej stronie wybierz drugą opcję - **JavaFx**.

**Zadanie 2.** Dodaj do projektu pliki biblioteki JavaFx. File->Project structure->Libraries->Znak "+". Przykładowa ścieżka: C:\JAVAFX\javafx-sdk-12.0.1\lib. Wybierz wszystkie pliki z rozszerzeniem .jar

**Zadanie 3.** Dodaj opcje wirtualnej maszyny. Run->Edit configurations. W polu VM options wpisz: `--module-path "C:\JAVAFX\javafx-sdk-12.0.1\lib" --add-modules javafx.controls,javafx.fxml` Podmieniając ścieżkę na własną.

**Zadanie 4.** Zmodyfikuj plik sample.fxml tak aby posiadał:

- 3 pola tekstowe (**TextField**)
- 3 pola opisujące nazwy pól tekstowych (**Label**)
- Przycisk "wyślij"

(Skorzystaj z SceneBuildera. Kliknij prawym przyciskiem myszy na pliku sample.fxml i wybierz *Open in Scene builder*)

**Zadanie 5.** Do projektu dodaj nową paczkę. Nazwij ją **com.utils**.

**Zadanie 6.** Do nowej paczki dodaj nową klasę **User**. Dodaj do niej pola: username, email oraz phone. Dopisz odpowiednie gettery, settery oraz konstruktor.

**Zadanie 7.** Reakcję na naciśnięcie przycisku dodaj jako metody w klasie Controller. Przykład:

```
public void Clicked(ActionEvent event) {  
  
    (...)  
  
}
```

**Zadanie 8.** Dodaj kod, który po naciśnięciu przycisku stworzy nową instancję klasy User z zawartości pól tekstowych.

Żeby móc pobrać zawartość pól konieczne jest dodanie ich jako pola w klasie Controller z tagiem @FXML.

Przykład:

@FXML

```
private Label label1;
```

**Zadanie 9.** Dodaj walidację pól przed stworzeniem użytkownika. Pole email powinno zawierać znak '@'. Pole username powinno zezwalać jedynie na małe litery. Pole phone powinno przyjmować wyłącznie 9 znaków numerycznych.

**Zadanie 10.** Pobierz biblioteki z linków poniżej, a następnie dodaj je do projektu w identyczny sposób jak w poprzednim przypadku.

<https://search.maven.org/artifact/com.google.code.gson/gson/2.8.5/jar>

<https://hc.apache.org/downloads.cgi>

**Zadanie 11.** Dopisz kod tak aby po stworzeniu użytkownika został on zamieniony na obiekt JSON. Wykorzystaj bibliotekę Gson.

**Zadanie 12.** Dopisz kod, który wysle żądanie HTTP POST na adres: <https://jsonplaceholder.typicode.com/users>. W treści żądania umieść nowego użytkownika. Sprawdź czy odpowiedź zawiera kod statusu 201 - created.