

Języki Programowania

dr inż. Tomasz Kubik

tomasz.kubik.staff.iiar.pwr.edu.pl

Program kursu

1. Wprowadzenie do języka Java poprzez porównanie z językami C/C++. Kod bajtowy i wirtualna maszyna. Kompilacja i uruchamianie aplikacji w zintegrowanym środowisku programowania.
2. Klasy, interfejsy, obiekty i cykl ich życia, typy podstawowe i referencje, strukturalna obsługa wyjątków, wzorce projektowe.
3. Technologie składowe i podstawowe pakiety klas platformy Java SE (w tym: ciągi znaków, tablice i kolekcje, strumienie).
4. Model obsługi zdarzeń. Budowa graficznego interfejsu użytkownika.
5. Realizacja wielowątkowości, sekcja krytyczna i wzajemne wykluczanie.
6. Pakiet klas do realizacji połączeń sieciowych.
7. Elementy programowania rozproszonego (architektura klient-serwer, mechanizmy zabezpieczeń), zdalne wywoływanie procedur.
8. Repetytorium.

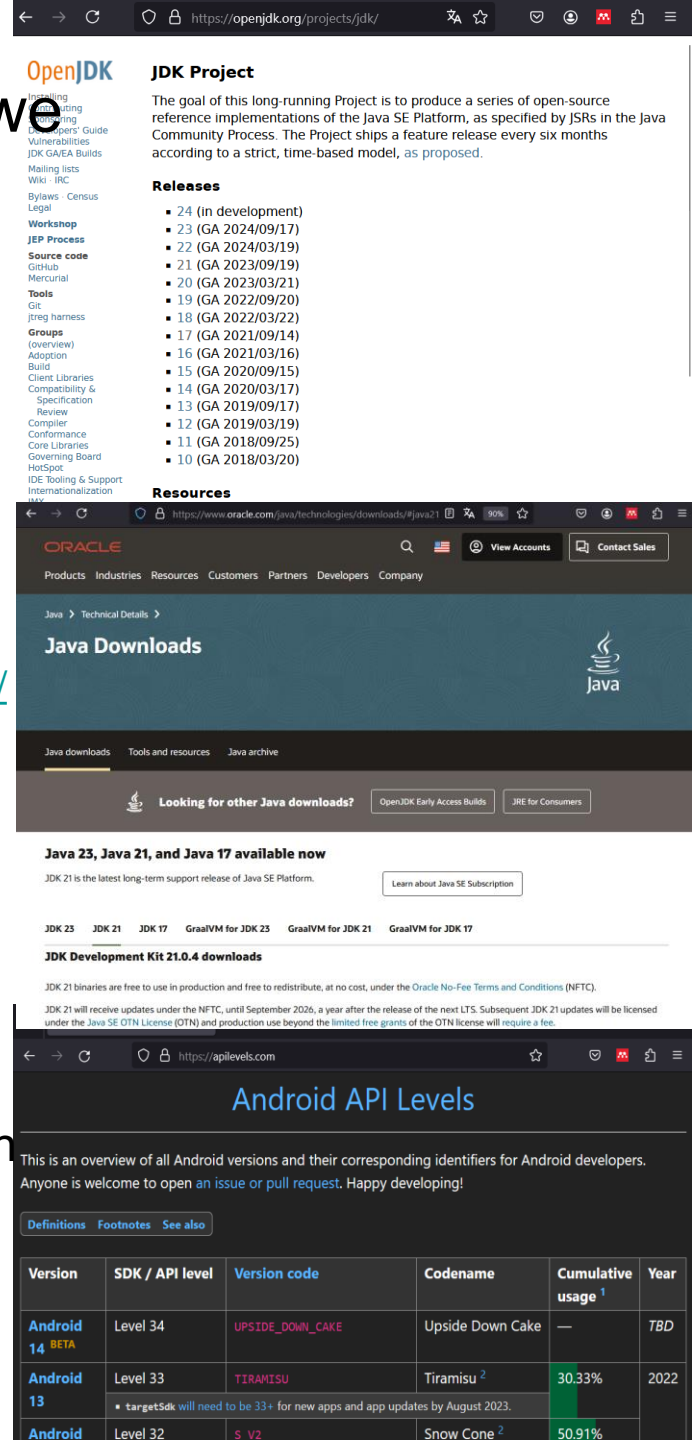
Zasady zaliczania

- Na kurs składają się
 - laboratoria (rozliczane na podstawie wykonanych zadań programistycznych)
 - wykład (rozliczany na podstawie wyników kolokwium przeprowadzonego w formie testu)
- Aby można było przystąpić do testu z wykładu należy pozytywnie zaliczyć laboratoria (tj. uzyskać pozytywną ocenę za tę część kursu)
- Ocena za kurs to średnia ważona ocen:
$$P = 0,6 * F1 + 0,4 * F2$$
gdzie F1 to ocena za część laboratoryjną,
a F2 to ocena za część wkładową

Narzędzia programowe

- JDK (ang. Java Development Kit)
 - dystrybucja środowiska uruchomieniowego oraz narzędzi
 - obowiązkowo JDK 17, preferowany JDK 21
 - dystrybucja
 - zalecane OpenJDK
 - <https://openjdk.org/projects/jdk/>
 - opcjonalnie JDK dostarczane przez Oracle
 - <https://www.oracle.com/java/technologies/downloads/>
- JRE (ang. Java Runtime Environment)
 - dystrybucja środowiska uruchomieniowego, zawiera JVM (Java Virtual Machine)
- Android API
 - Android OS oparto na Linuksie zawierającym wirtualną maszynę DVM (Dalvik Virtual Machine) pozwalającą na uruchamianie aplikacji napisanych w języku Java

Ciekawostka:
https://en.wikipedia.org/wiki/Google_LLC_v._Oracle_America,_Inc.



Licencje JDK

[JDK 23](#) [JDK 21](#) [JDK 17](#) [GraalVM for JDK 23](#) [GraalVM for JDK 21](#) [GraalVM for JDK 17](#)

JDK Development Kit 17.0.12 downloads

This is the last planned update of [JDK 17 under the NFTC](#). Updates after September 2024 will be licensed under the [Java SE OTN License](#) (OTN) and production use beyond the limited free grants of the OTN license will [require a fee](#).

JDK 17 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions](#) (NFTC).

[JDK 23](#) [JDK 21](#) [JDK 17](#) [GraalVM for JDK 23](#) [GraalVM for JDK 21](#) [GraalVM for JDK 17](#)

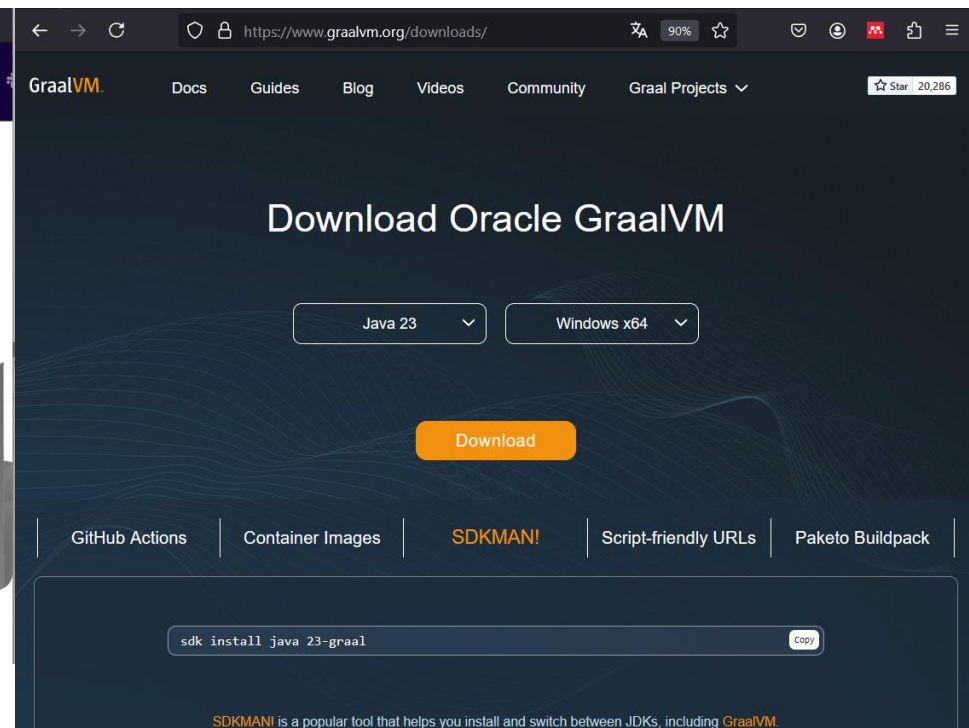
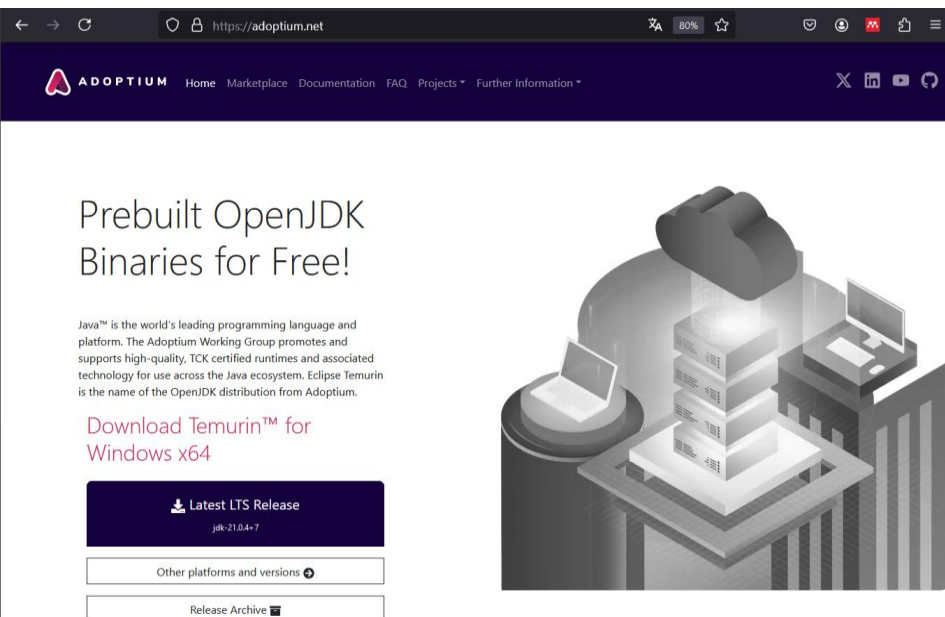
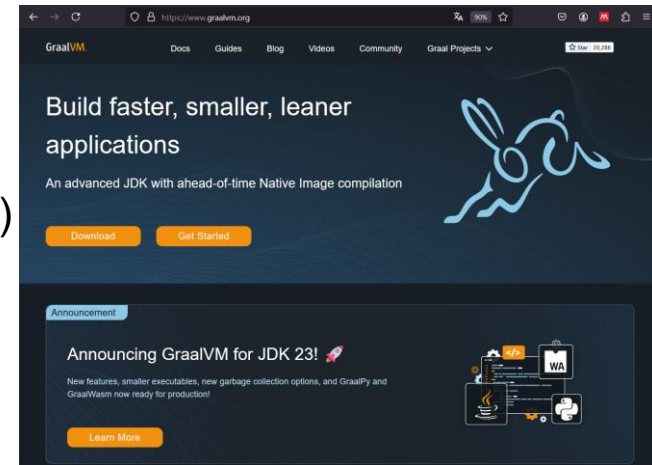
JDK Development Kit 21.0.4 downloads

JDK 21 binaries are free to use in production and free to redistribute, at no cost, under the [Oracle No-Fee Terms and Conditions](#) (NFTC).

JDK 21 will receive updates under the NFTC, until September 2026, a year after the release of the next LTS. Subsequent JDK 21 updates will be licensed under the [Java SE OTN License](#) (OTN) and production use beyond the [limited free grants](#) of the OTN license will [require a fee](#).

Narzędzia programowe

- JDK (ang. Java Development Kit)
 - dystrybucja
 - GraalVM (środowisko wspierające kilka języków)
 - <https://www.graalvm.org/>
 - <https://www.graalvm.org/downloads/>
 - AdoptOpenJDK -> Adoptium.net
 - <https://adoptopenjdk.net/> -> <https://adoptium.net/>



Narzędzia programowe

<https://www.oracle.com/java/technologies/java-se-support-roadmap.html>

Release	GA Date	Premier Support Until	Extended Support Until	Sustaining Support
8 (LTS)**	March 2014	March 2022	December 2030****	Indefinite
9 - 10 (non-LTS)	September 2017 - March 2018	March 2018 - September 2018	Not Available	Indefinite
11 (LTS)	September 2018	September 2023	January 2032****	Indefinite
12 - 16 (non-LTS)	March 2019 - March 2021	September 2019 - September 2021	Not Available	Indefinite
17 (LTS)	September 2021	September 2026****	September 2029****	Indefinite
18 - 20 (non-LTS)	March 2022 - March 2023	September 2022 - September 2023	Not Available	Indefinite
21 (LTS)	September 2023	September 2028****	September 2031****	Indefinite
22 (non-LTS)	March 2024	September 2024	Not Available	Indefinite
23 (non-LTS)***	September 2024	March 2025	Not Available	Indefinite
24 (non-LTS)***	March 2025	September 2025	Not Available	Indefinite
25 (LTS)***	September 2025	September 2030	September 2033	Indefinite

* Oracle Java SE product dates are provided as **examples** to illustrate the support policies. Customers should refer to the [Oracle Lifetime Support Policy](#) for the most up-to-date information. Timelines may differ for Oracle Products with a Java SE dependency ([My Oracle Support Note 1557737.1 - Support Entitlement for Java SE When Used As Part of Another Oracle Product](#) – Requires Support Login).

** These support timelines apply to Java client and server deployments of Java with the exception of Web Deployment Technology and JavaFX. For more information on those features, see below.

*** LTS/non-LTS designation and dates, as noted in the above example, are subject to change.

**** Or later.

***** The Extended Support uplift fee will be waived for the period October 2023 – January 2032 for Java SE 11. The Extended Support uplift fee will be waived for the period March 2022 - December 2030 for Java SE 8. During this period, you will receive Extended Support as described in the Oracle Technical Support Level sections of the Technical Support Policies..

† Excluding Deployment Technology and JavaFX, which is described in a separate section.

<https://www.oracle.com/java/technologies/java-se-support-roadmap.html>

<https://www.java.com/releases/>

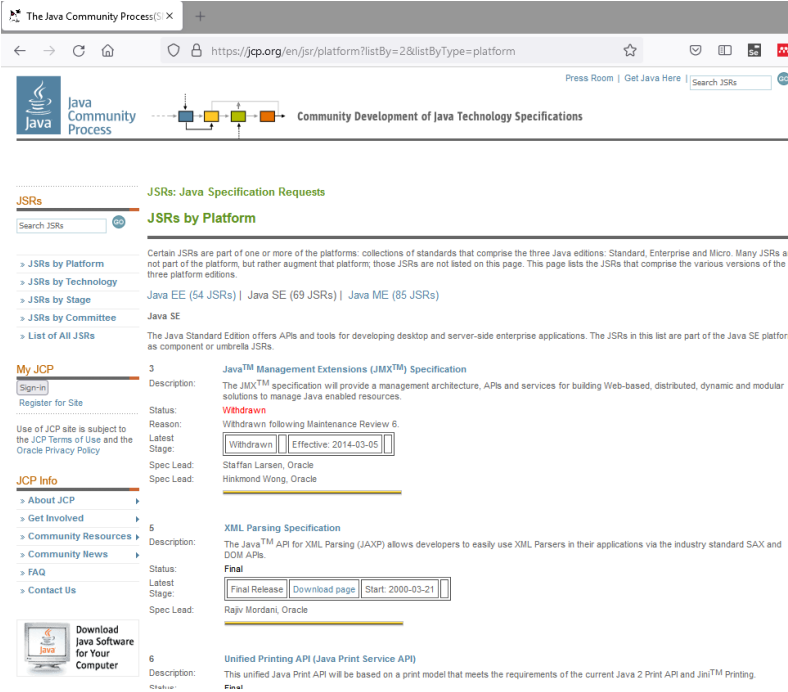
<https://www.codejava.net/java-se/java-se-versions-history>

<https://endoflife.date/java>

Źródła

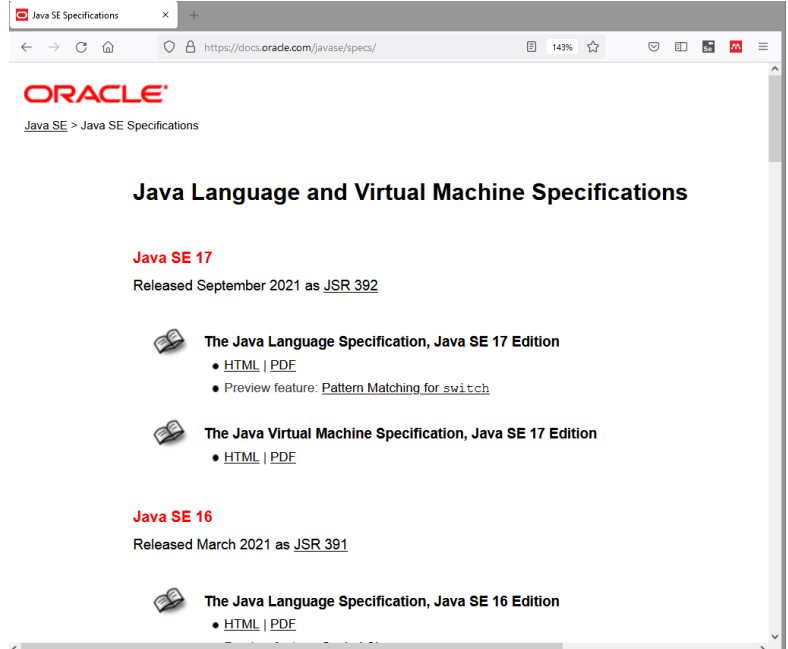
- Standaryzacja
 - Java Community Process
 - <https://jcp.org/en/home/index>
 - Specyfikacja języka
 - <https://docs.oracle.com/javase/specs/>
- Kod źródłowy JDK
 - <https://github.com/openjdk/jdk>
 - <https://github.com/openjdk/jdk/tree/jdk-11%2B28/src/hotspot/share/gc>
 - <https://openjdk.java.net/groups/build/doc/building.html>

Aby skompilować nową wersję JDK trzeba mieć JDK w wersji równej lub mniejszej o 1 (!)



The screenshot shows the 'The Java Community Process' website. The main navigation includes 'Press Room', 'Get Java Here', and a search bar. The page is titled 'Community Development of Java Technology Specifications'. The 'JSRs' section is active, displaying 'JSRs by Platform' and 'JCP Info'. The 'JCP Info' section lists several specifications:

- Java™ Management Extensions (JMX™) Specification**: Status: Withdrawn following Maintenance Review 6. Latest Stage: Withdrawn. Effective: 2014-03-05. Spec Lead: Staffen Larsen, Oracle; Hinkmond Wong, Oracle.
- XML Parsing Specification**: Status: Final. Latest Stage: Final Release. Download page. Start: 2000-03-21. Spec Lead: Rajiv Mordani, Oracle.
- Unified Printing API (Java Print Service API)**: Status: Final. Description: This unified Java Print API will be based on a print model that meets the requirements of the current Java 2 Print API and Jini™ Printing.



The screenshot shows the Oracle Java SE Specifications page. The page is titled 'Java Language and Virtual Machine Specifications'. It lists the following specifications:

- Java SE 17**: Released September 2021 as JSR 392.
 - The Java Language Specification, Java SE 17 Edition
 - [HTML](#) | [PDF](#)
 - Preview feature: [Pattern Matching for switch](#)
 - The Java Virtual Machine Specification, Java SE 17 Edition
 - [HTML](#) | [PDF](#)
- Java SE 16**: Released March 2021 as JSR 391.
 - The Java Language Specification, Java SE 16 Edition
 - [HTML](#) | [PDF](#)

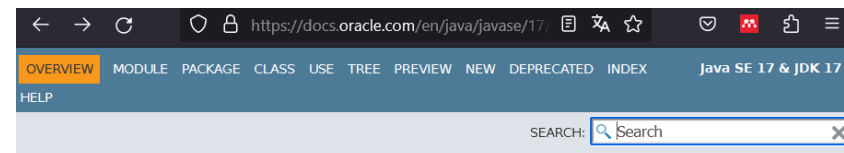
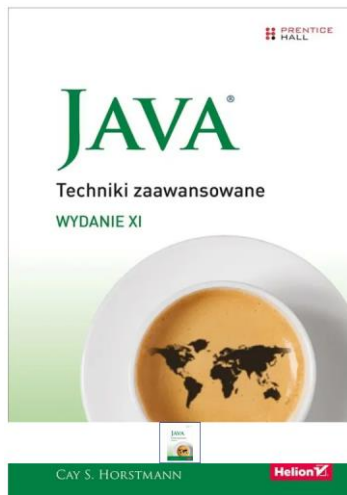
Źródła

- Dokumentacja

- <https://docs.oracle.com/en/java/>
- <https://docs.oracle.com/en/java/javase/17/docs/api/index.html>
- <https://docs.oracle.com/en/java/javase/17/>

- Książki

- Cay S. Horstmann: Java. Techniki zaawansowane. Wydanie 11



Java® Platform, Standard Edition & Java Development Kit Version 17 API Specification

This document is divided into two sections:

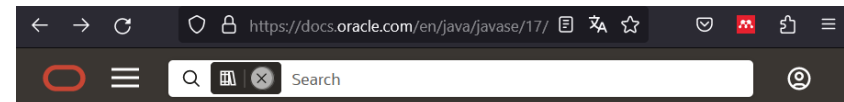
Java SE

The Java Platform, Standard Edition (Java SE) APIs define the core Java platform for general-purpose computing. These APIs are in modules whose names start with `java`.

JDK

The Java Development Kit (JDK) APIs are specific to the JDK and will not necessarily be available in all implementations of the Java SE Platform. These APIs are in modules whose names start with `jdk`.

All Modules	Java SE	JDK	Other Modules
Module	Description		
<code>java.base</code>	Defines the foundational APIs of the Java SE Platform.		
<code>java.compiler</code>	Defines the Language Model, Annotation Processing, and Java Compiler APIs.		
<code>java.datatransfer</code>	Defines the API for transferring data between and within applications.		



← Java SE

More ▾

JDK 17 Documentation

Home

[Java Components page](#)

Looking for a different release? [Other releases](#)



Overview

[Read Me](#)

[Release Notes](#)

Narzędzia programowe

- Eclipse IDE (obecnie 2023-09 R)
 - <https://www.eclipse.org/downloads/>
 - <https://www.eclipse.org/downloads/packages/>

The image shows a screenshot of the Eclipse Foundation website, specifically the page for Eclipse IDE 2021-09 R packages. The page is displayed in a browser window with the URL <https://www.eclipse.org/downloads/packages/>. The page features a dark header with the Eclipse Foundation logo and navigation links for Projects, Working Groups, Members, and More-. Below the header, there is a main content area with a large blue banner that reads "Try the Eclipse Installer 2021-09 R" and "The easiest way to install and update your Eclipse Development Environment." To the right of this banner, there is a "Download" button and a list of operating systems and architectures: macOS x86_64, Windows x86_64, Linux x86_64, and AArch64. Below the banner, there is a section titled "Eclipse IDE 2021-09 R Packages" which lists two packages: "Eclipse IDE for Java Developers" (322 MB, 387,100 DOWNLOADS) and "Eclipse IDE for Enterprise Java and Web Developers" (517 MB, 255,805 DOWNLOADS). Each package has a download icon and a list of supported operating systems and architectures. At the bottom of the page, there is a "RELATED LINKS" section with links to "Compare & Combine Packages" and "New and Noteworthy".

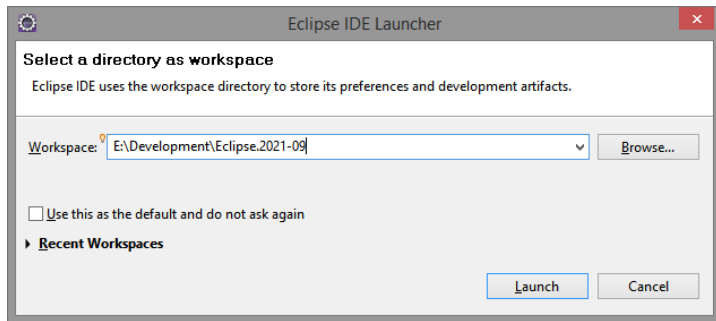
The screenshot also shows a sidebar on the left with a "Tool Platforms" section, featuring Eclipse Che and Orion. The Eclipse Che section includes the text "Eclipse Che is a developer workspace server and cloud IDE." and the Orion section includes "A modern, open source software development environment that runs in the cloud." There is also a "Sponsored Ad" for the "IoT & Edge Developer Survey 2021" with a "PARTICIPATE TODAY!" banner.

Eclipse IDE - launch

Uruchomienie

- przez klik na `eclipse.exe` w menadżerze plików lub
- z linii komend w katalogu `eclipse`:
> `eclipse.exe`

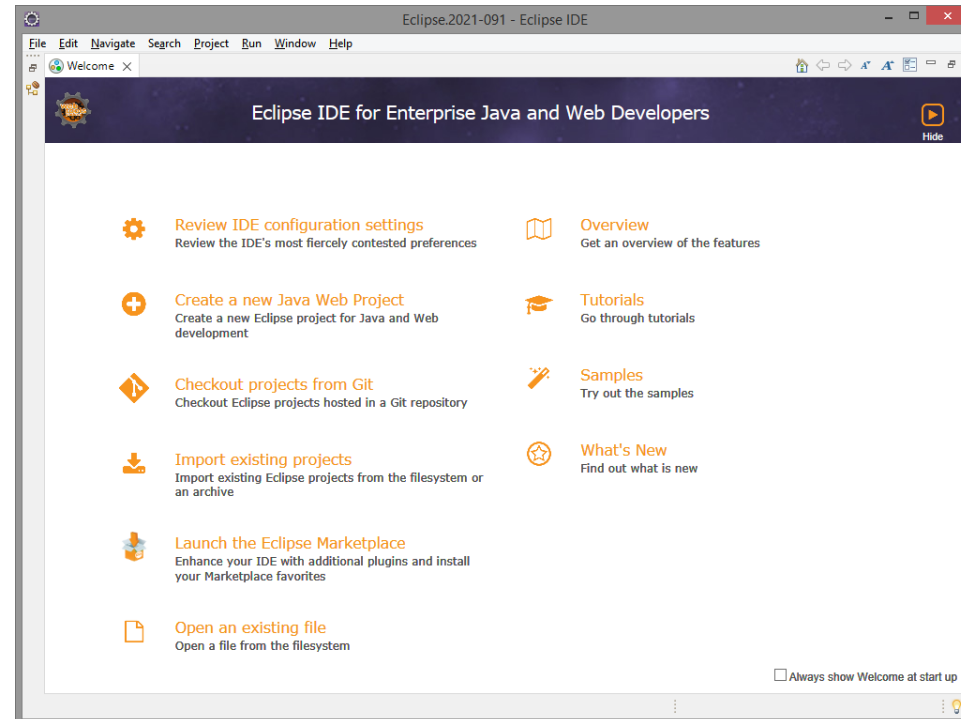
Podczas uruchomienia pojawia się monit o wybranie przestrzeni roboczej



The screenshot shows a Windows File Explorer window titled 'eclipse-jee-2021-09-R-win32-x86_64 > eclipse'. The search bar contains 'Przeszukaj: eclipse'. The main area displays a list of files and folders with columns for 'Nazwa', 'Data modyfikacji', 'Typ', and 'Rozmiar'. The file 'eclipse.exe' is selected and highlighted in blue.

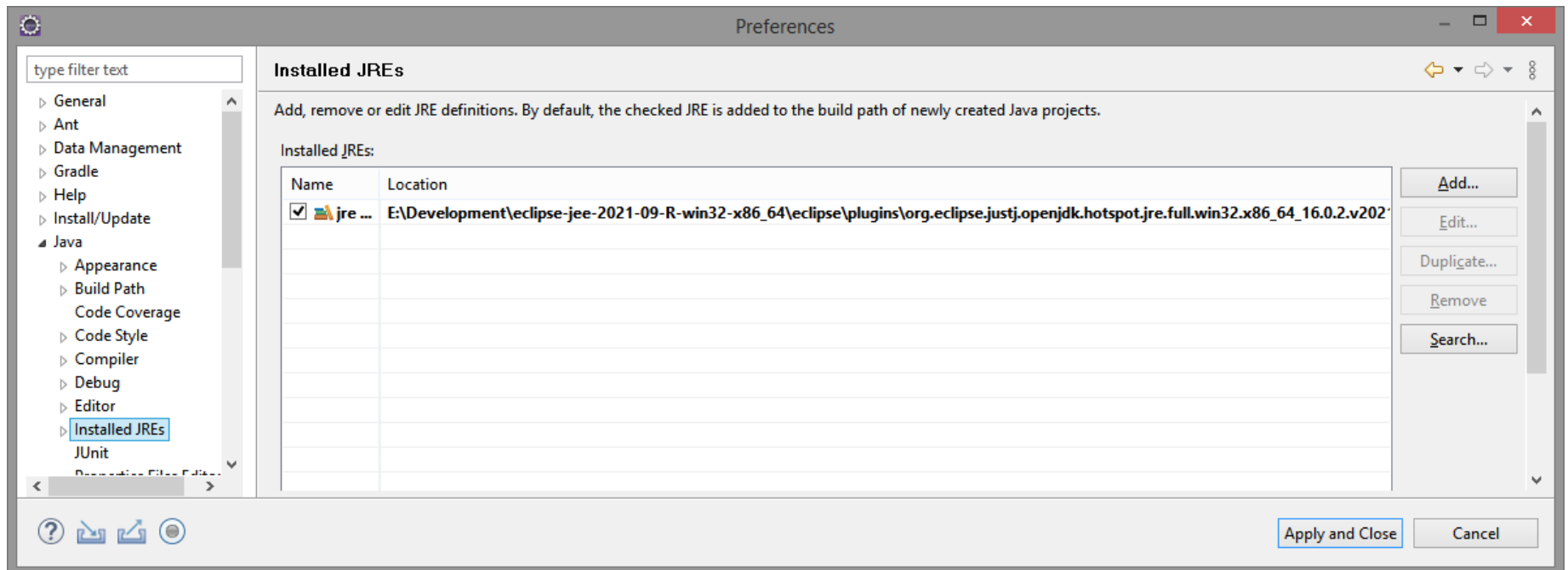
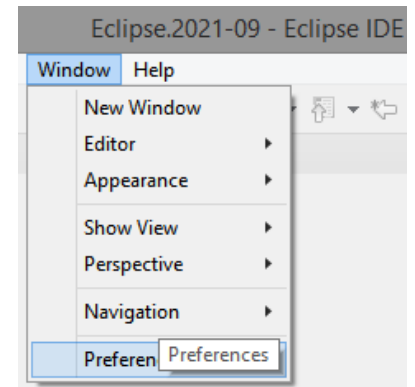
Nazwa	Data modyfikacji	Typ	Rozmiar
configuration	2021-10-03 20:30	Folder plików	
dropins	2021-09-10 14:46	Folder plików	
features	2021-10-03 20:37	Folder plików	
p2	2021-10-03 20:44	Folder plików	
plugins	2021-10-03 20:37	Folder plików	
readme	2021-10-03 18:59	Folder plików	
.eclipseproduct	2021-09-06 09:06	Plik ECLIPSEPROD...	1 KB
artifacts.xml	2021-10-03 20:37	Plik XML	287 KB
eclipse.exe	2021-09-10 14:48	Aplikacja	519 KB
eclipse.ini	2021-10-03 20:37	Ustawienia konfig...	1 KB
eclipsesec.exe	2021-09-10 14:49	Aplikacja	231 KB

Po uruchomieniu pojawia się okno powitalne



Eclipse IDE - Java

- Środowisko Eclipse 2021-09 (4.21) dostarczane jest razem z JRE w wersji 16
- Można się o tym przekonać sprawdzając preferencje



Eclipse IDE – Java 17 Support

- Eclipse 2021-09 (4.21) wspiera składnię Java do wersji 16.
- Aby uzyskać wsparcie wersji 17 należy zainstalować odpowiednie rozszerzenie

The image consists of three screenshots from the Eclipse IDE interface:

- Left Screenshot:** Shows the Eclipse IDE window titled "Eclipse.2021-09 - Eclipse IDE". The "Help" menu is open, and the "Eclipse Marketplace..." option is selected. A tooltip over the "Eclipse Marketplace..." option reads "Open the Eclipse Marketplace wizard".
- Middle Screenshot:** Shows the Eclipse Marketplace window. The search bar contains "17". The search results list several solutions, with "Java 17 Support for Eclipse 2021-09 (4.21)" selected. The description for this solution states: "This marketplace solution provides Java 17 support for Eclipse 2021-09 (4.21). To install the feature, please ensure you have the most recent version of Eclipse... more info". It is by "The Eclipse Foundation, EPL 2.0" and has "2,27K (2 183 last month)" installs. An "Install" button is visible.
- Right Screenshot:** Shows the "Confirm Selected Features" dialog box. It lists the selected features for installation:
 - Java 17 Support for Eclipse 2021-09 (4.21) <https://download.eclipse.org/eclipse/upda>
 - Eclipse JDT (Java Development Tools) Patch with Java 17 support for 2021-09 deve
 - Eclipse PDE Patch with Java 17 support for 2021-09 development stream (required
 - Eclipse JDT (Java Development Tools) Source Patch with Java 17 support for 2021-
 - Eclipse PDE Source Patch with Java 17 support for 2021-09 development streamThe dialog includes a progress bar at the bottom labeled "Calculating requirements and dependencies." and buttons for "< Install More", "Confirm >", "Finish", and "Cancel".

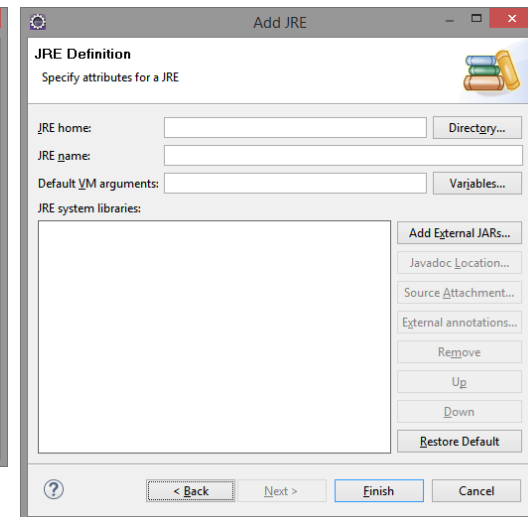
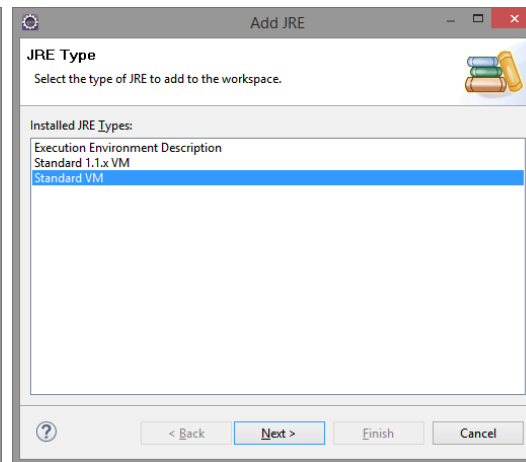
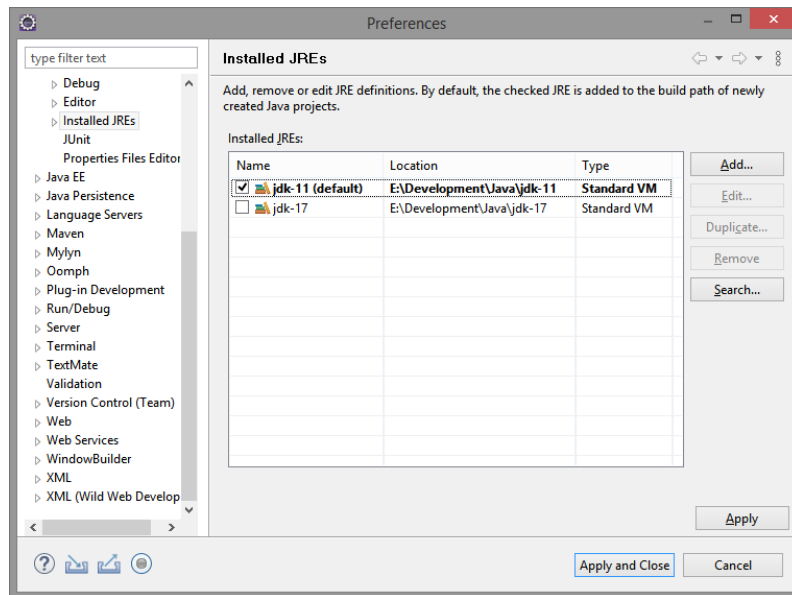
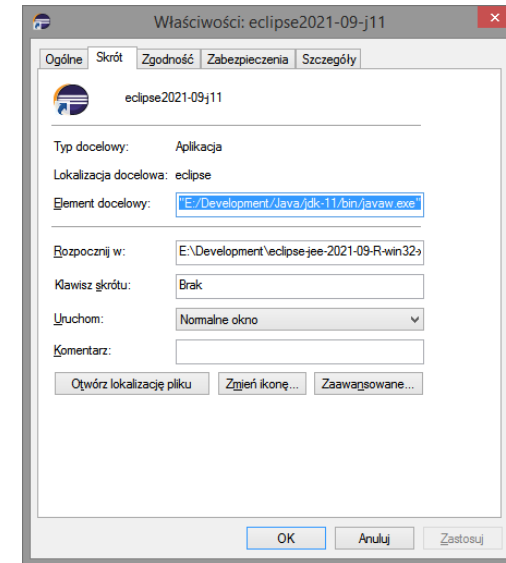
Eclipse IDE – Java

Środowisku można wskazać JRE, z jakim ma pracować:

- poprzez użycie opcji `-vm` (ustawiającej domyślne JRE) podczas uruchomienia `eclipse` z linii komend lub przez skrót:

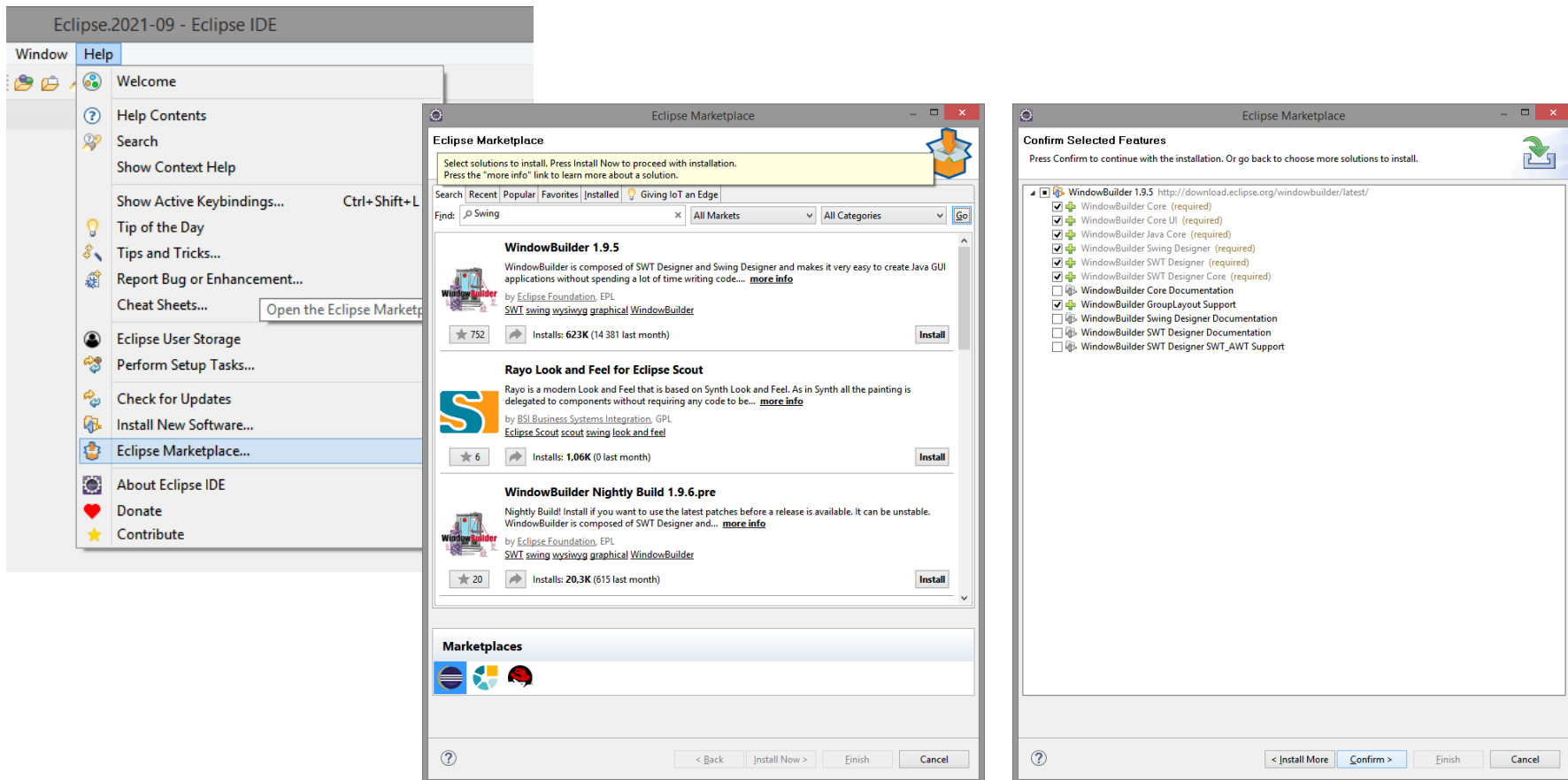
> `eclipse.exe -vm "E:/Development/Java/jdk-11/bin/javaw.exe"`

- poprzez ustawienie preferencji (trzeba wskazać typ JRE oraz ścieżkę)



Eclipse IDE – WindowBuilder

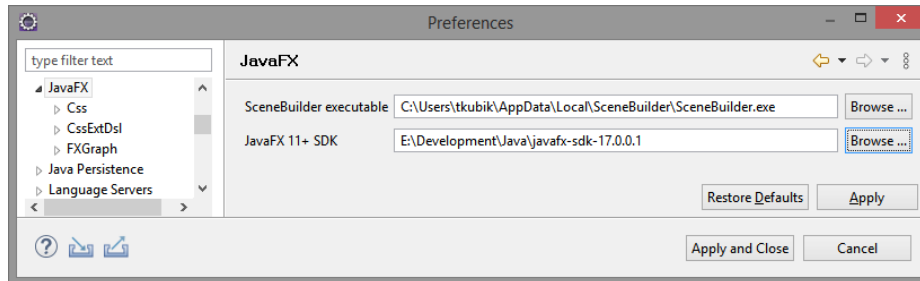
- Eclipse domyślnie nie oferuje narzędzia do projektowania GUI dla programów Java. Takie narzędzia należy sobie doinstalować.
- WindowBuider pozwala na projektowanie GUI za pomocą klas Swing, AWT, SWT



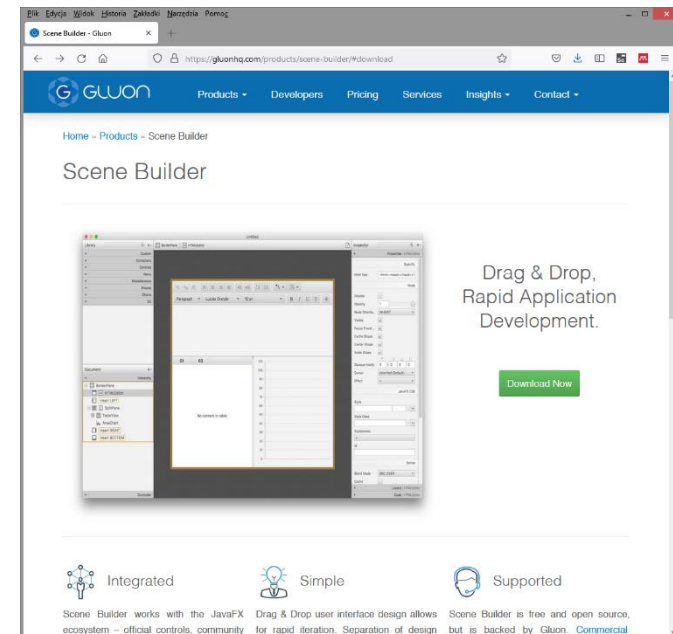
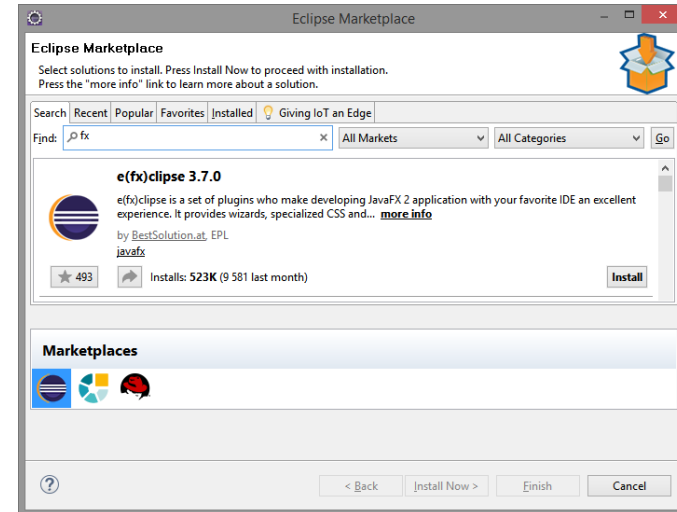
Eclipse IDE – e(fx)clipse

W środowisku można również uzyskać wsparcie projektowania aplikacji JavaFX, ale wymaga to

- obecności JavaFX na komputerze
 - do pobrania ze strony:
<https://gluonhq.com/products/javafx/>
- zainstalowania rozszerzenia e(fx)clipse z MarketPlace
- zainstalowania SceneBuilder do projektowania GUI (wersja 17.0.0 działa z JDK11 i wyżej)
 - domyślnie instalowane jest w katalogu:
C:\Users\\AppData\Local\Scene Builder\
– ścieżkę tę należy ustawić w preferencjach eclipse



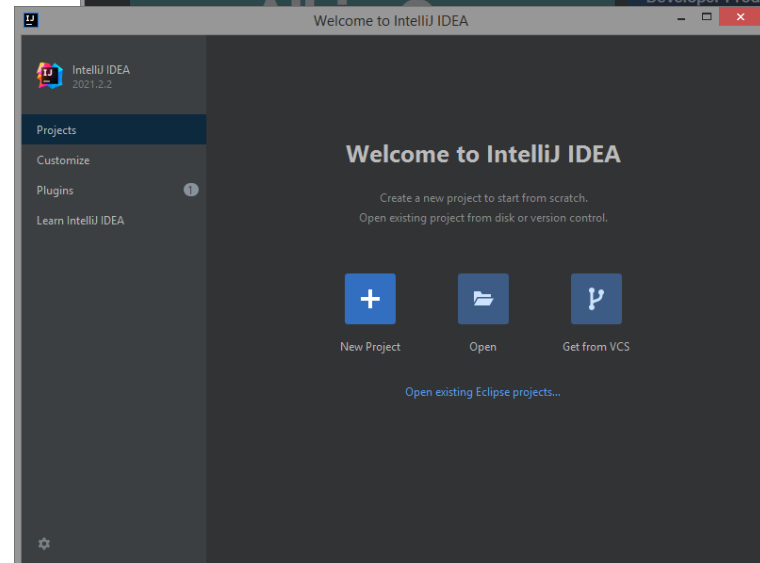
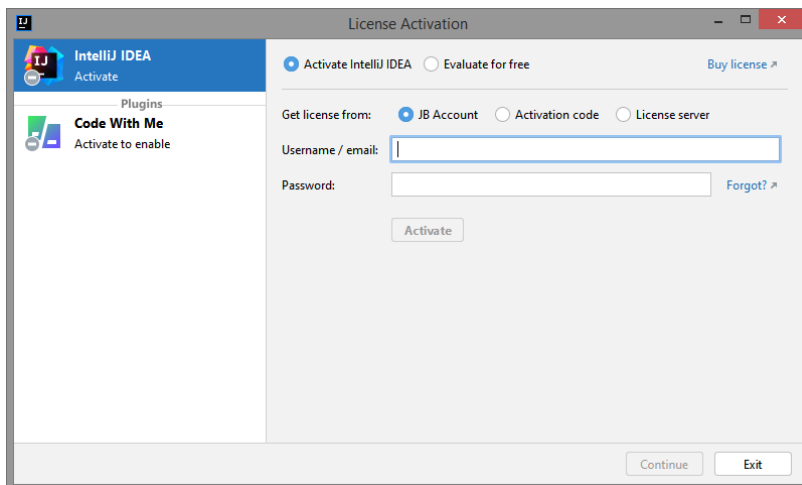
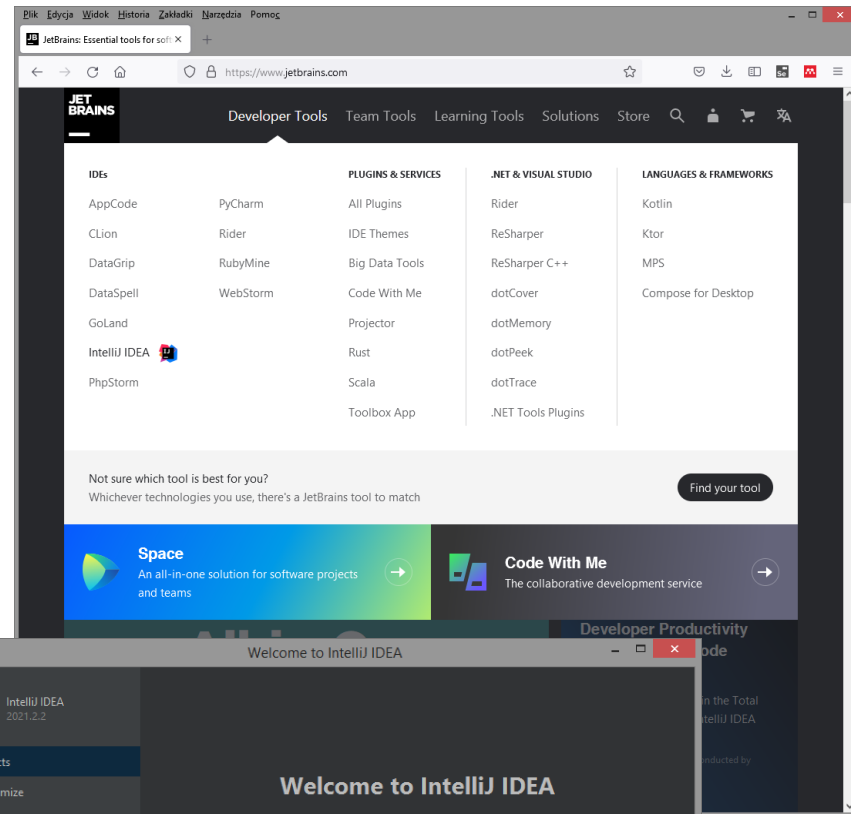
<https://gluonhq.com/products/scene-builder/#download>



IntelliJ IDEA

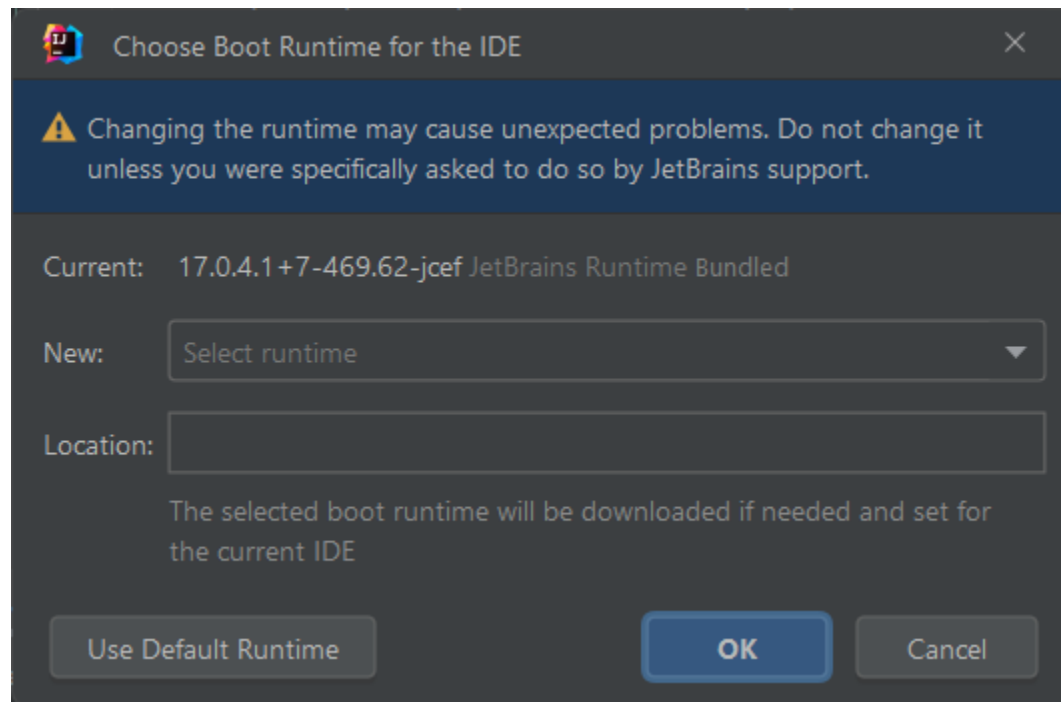
<https://www.jetbrains.com/idea/download/>

- Oprogramowanie w wersji
 - Ultimate (komercyjna, ale z dostępną licencją edukacyjną)
 - Community (darmowa)
- Pobrać można
 - instalator (exe) albo
 - archiwum (zip)
- Przy pierwszym uruchomieniu wersji Ultimate pojawi się
 - monit o aktywację licencji, a potem (po wpisaniu parametrów uwierzytelnienia użytkownika zarejestrowanego na stronie JetBrains)
 - okno powitalne



IntelliJ IDEA

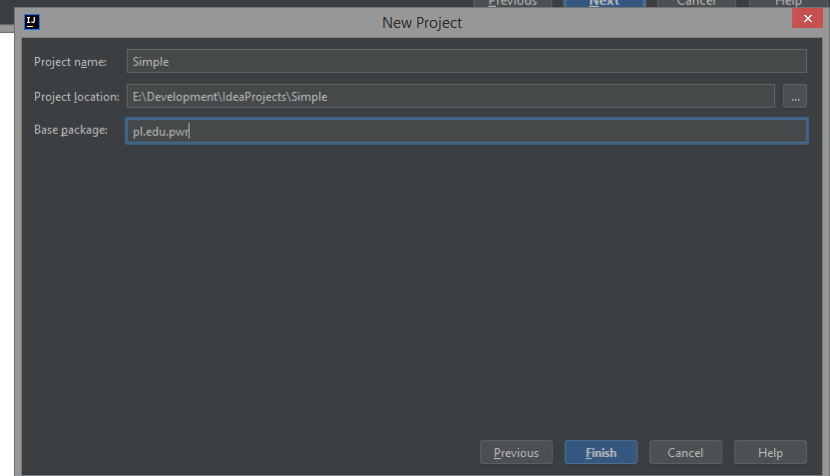
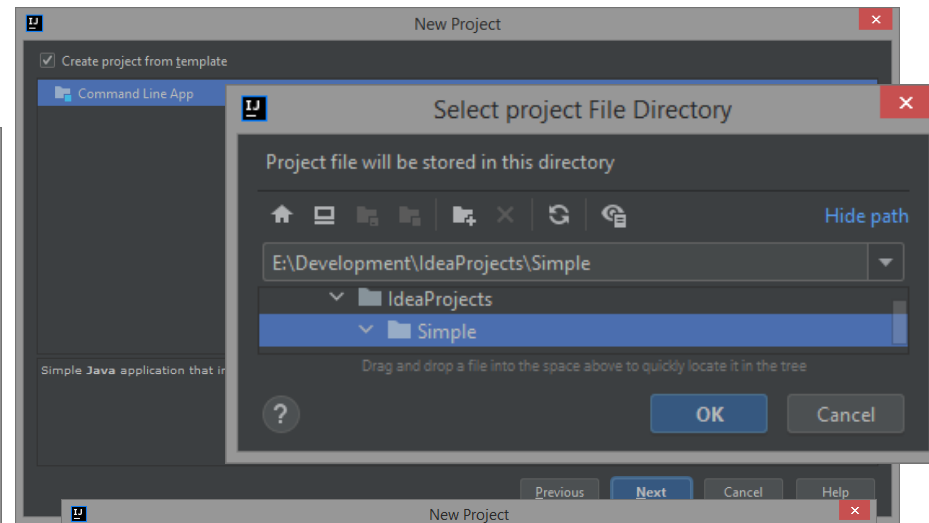
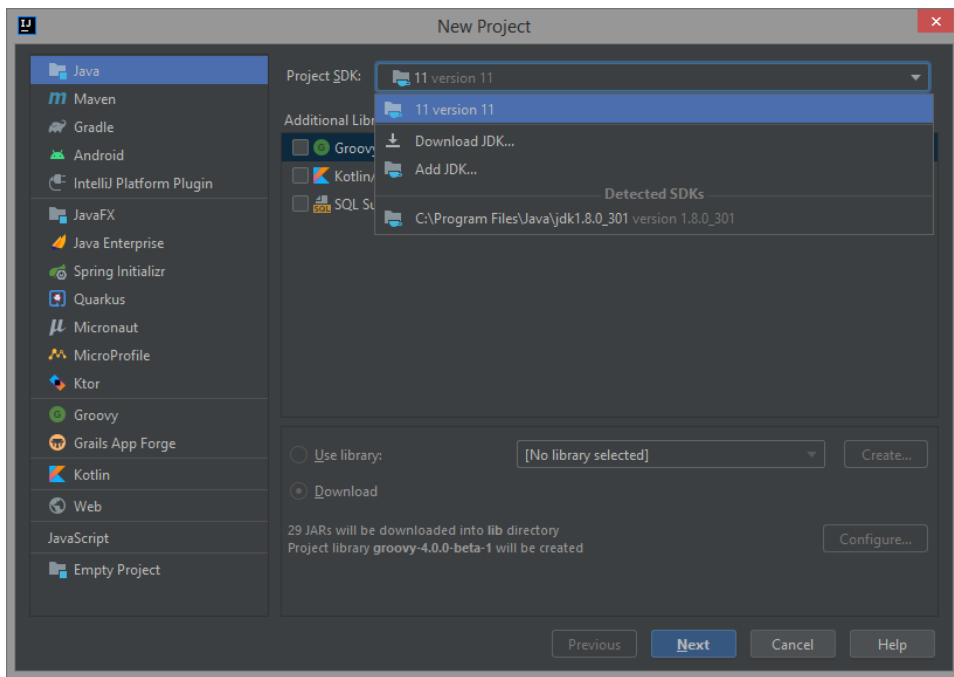
- Domyślnie środowisko IntelliJ IDEA 2022.2.3 (Ultimate Edition) działa w JRE 17.0.4.1+7-469.62-jcef
 - można się o tym przekonać próbując zmienić Boot Runtime



<https://www.jetbrains.com/help/idea/switching-boot-jdk.html>

IntelliJ IDEA - praca ze środowiskiem

- Podczas tworzenia projektów można zadeklarować, z którym JDK mają być one kompilowane i uruchamiane

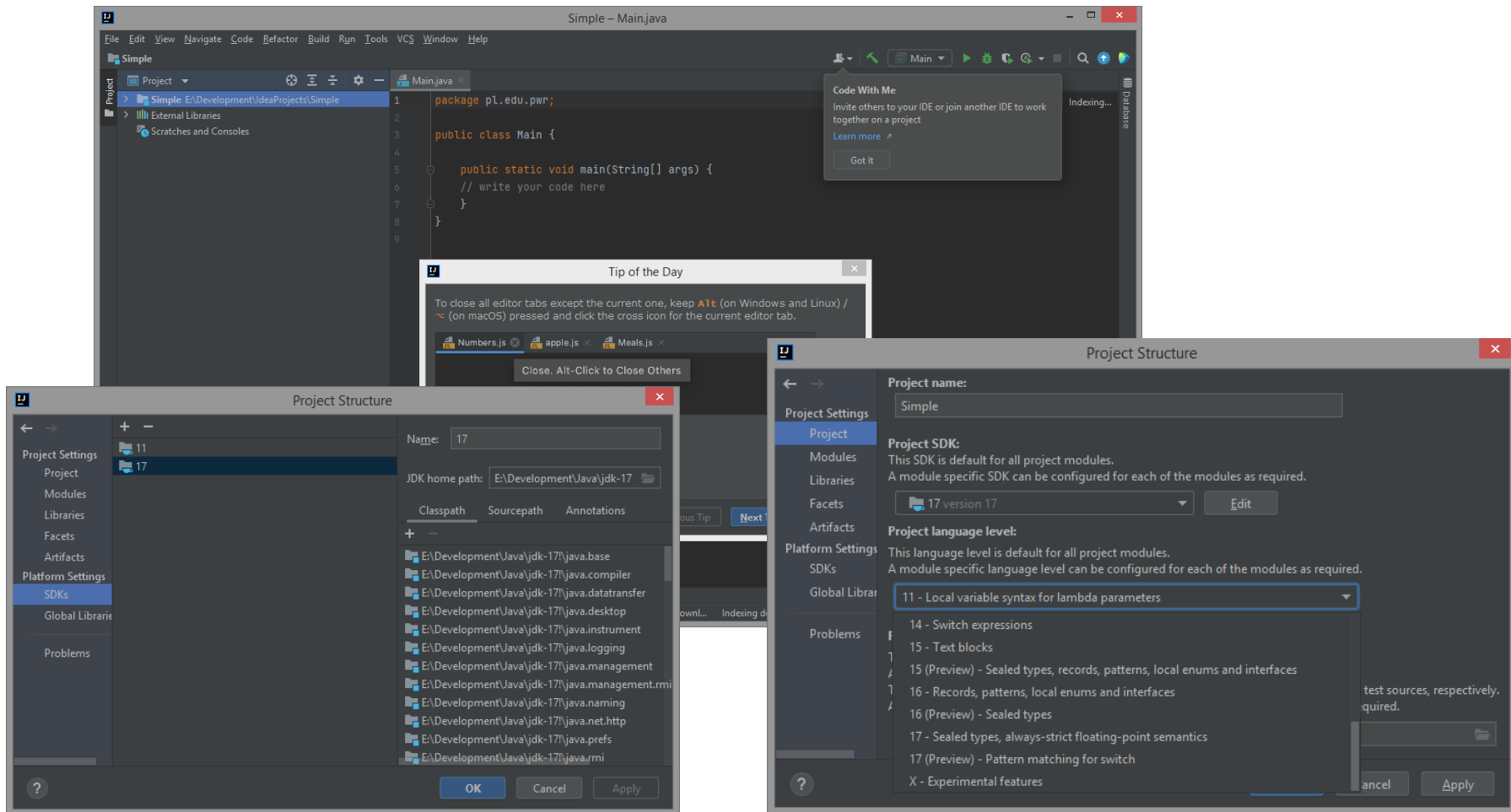


Do poczytania:

<https://blog.jetbrains.com/idea/2021/09/java-17-and-intellij-idea/>

IntelliJ IDEA – praca ze środowiskiem

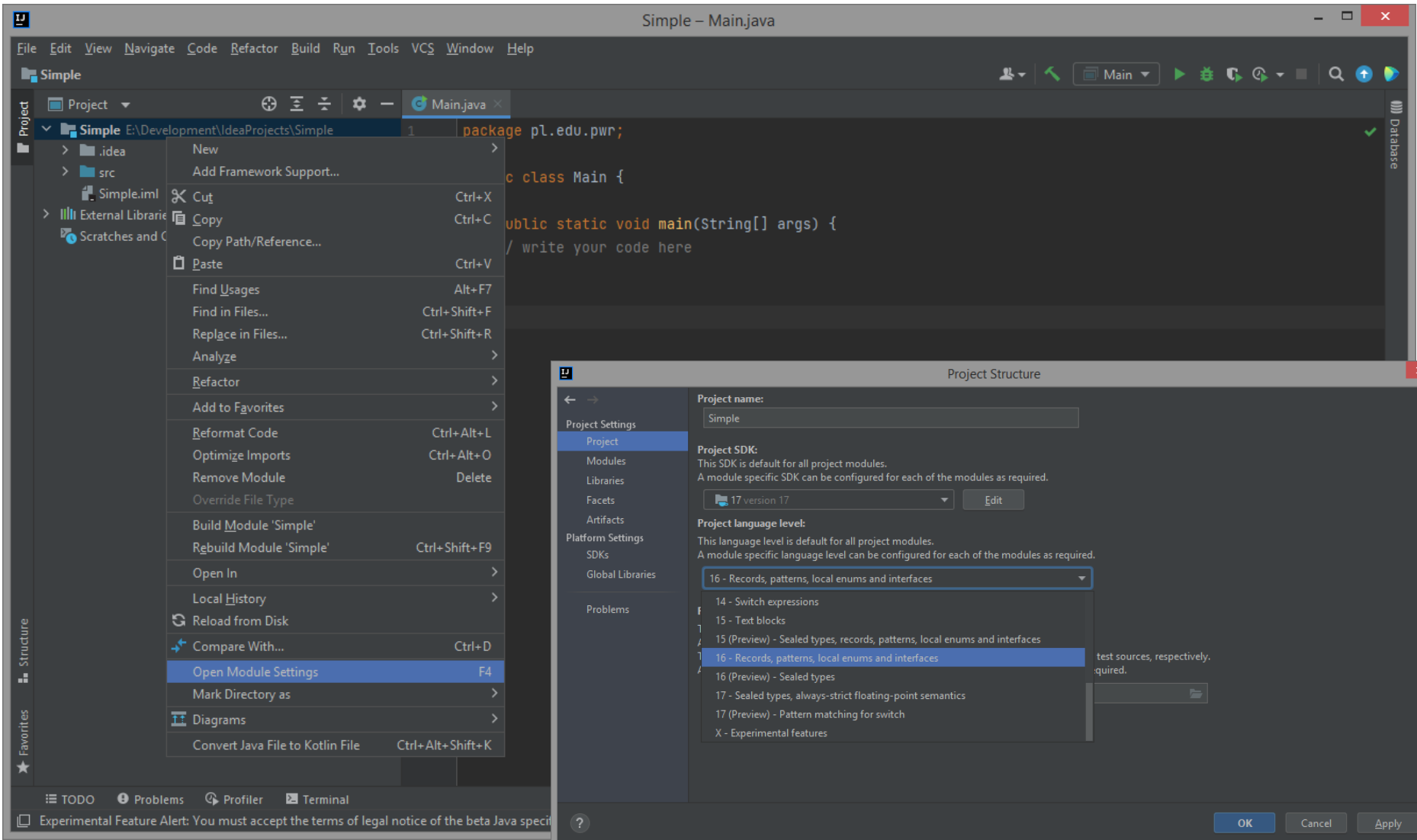
- Po uruchomieniu projektu można zmienić JDK w dowolnym momencie w ustawieniach Project Structure (Ctrl-Alt-Shift-S)



<https://mkyong.com/intellij/how-to-change-the-intellij-idea-jdk-version/>

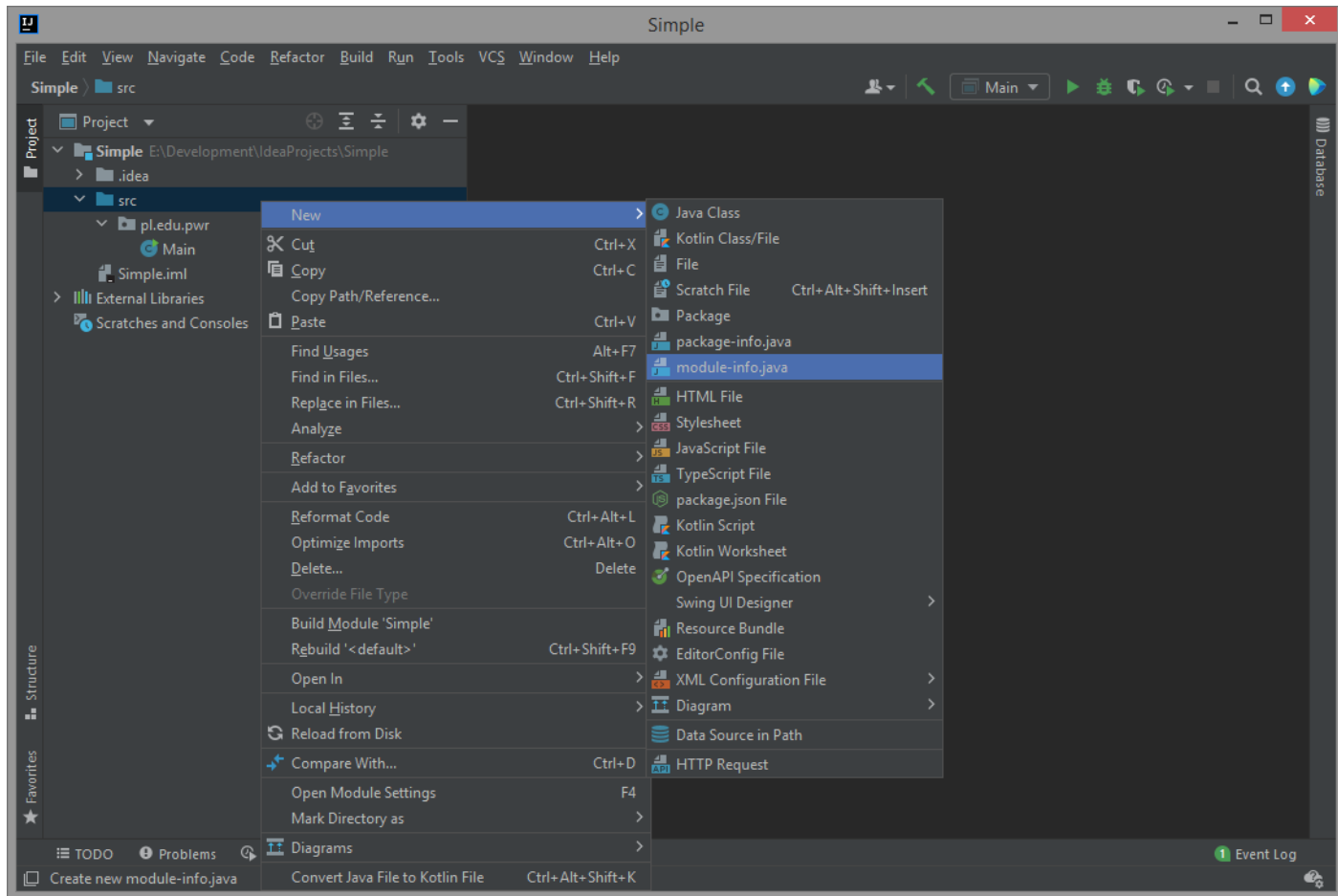
IntelliJ IDEA – praca ze środowiskiem

- Do ustawień można dojść również poprzez Open Module Settings



IntelliJ IDEA – praca ze środowiskiem

- Projekty Java domyślnie tworzone są bez `module-info.java`
- Plik ten jednak można w każdej chwili dodać

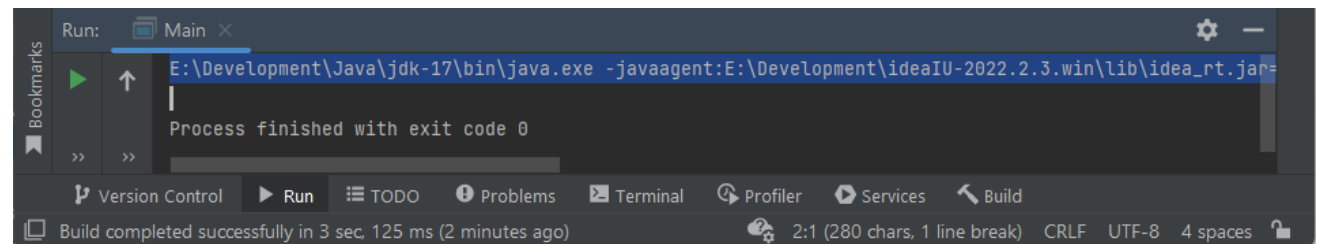


IntelliJ IDEA – praca ze środowiskiem

- Po uruchomieniu można zobaczyć, jaką komendę rzeczywiście odpaliło środowisko:
 - wersja projektu bez `module-info.java` (środowisko automatycznie ustawia ścieżkę klas)

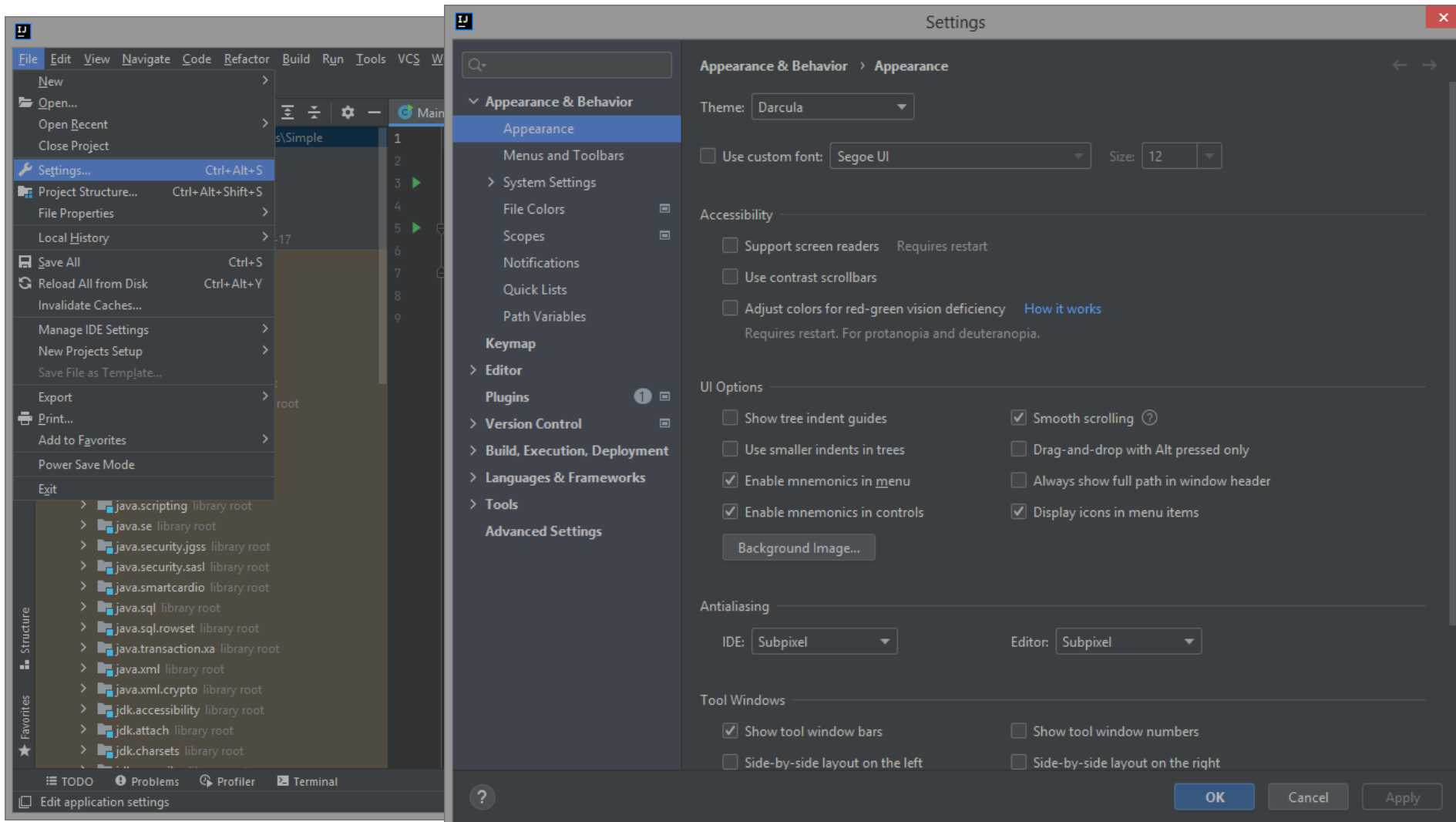
```
E:\Development\Java\jdk-17\bin\java.exe -javaagent:  
E:\Development\ideaIU-  
2022.2.3.win\lib\idea_rt.jar=51005:E:\Development\ideaIU-  
2022.2.3.win\bin -Dfile.encoding=UTF-8 -classpath  
E:\Dydaktyka\JęzykiProgramowania\2022\projektyIntelliJ\Simple\ou  
t\production\Simple pl.edu.pwr.Main
```
 - wersja z `module-info.java` (środowisko automatycznie ustawia ścieżkę modułową)

```
E:\Development\Java\jdk-17\bin\java.exe -javaagent:  
E:\Development\ideaIU-  
2022.2.3.win\lib\idea_rt.jar=51005:E:\Development\ideaIU-  
2022.2.3.win\bin -Dfile.encoding=UTF-8 -p  
E:\Dydaktyka\JęzykiProgramowania\2022\projektyIntelliJ\Simple\ou  
t\production\Simple -m Simple/pl.edu.pwr.Main
```
- Widać, że w opcjach wirtualnej maszyny pojawił się `javaagent` – dzięki niemu środowisko ma możliwość zajrzenia do środka wirtualnej maszyny poprzez instrumentację



IntelliJ IDEA – opcje interfejsu

- Środowisko pozwala dopasować wygląd interfejsu zgodnie z preferencjami



IntelliJ IDEA - JavaFX

- Projekt typu JavaFX to projekt mavenowy (konfiguracja w pliku `pom.xml`)

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>pl.edu.pwr</groupId>
  <artifactId>SimpleFX</artifactId>
  <version>1.0-SNAPSHOT</version>
  <name>SimpleFX</name>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <junit.version>5.7.1</junit.version>
  </properties>

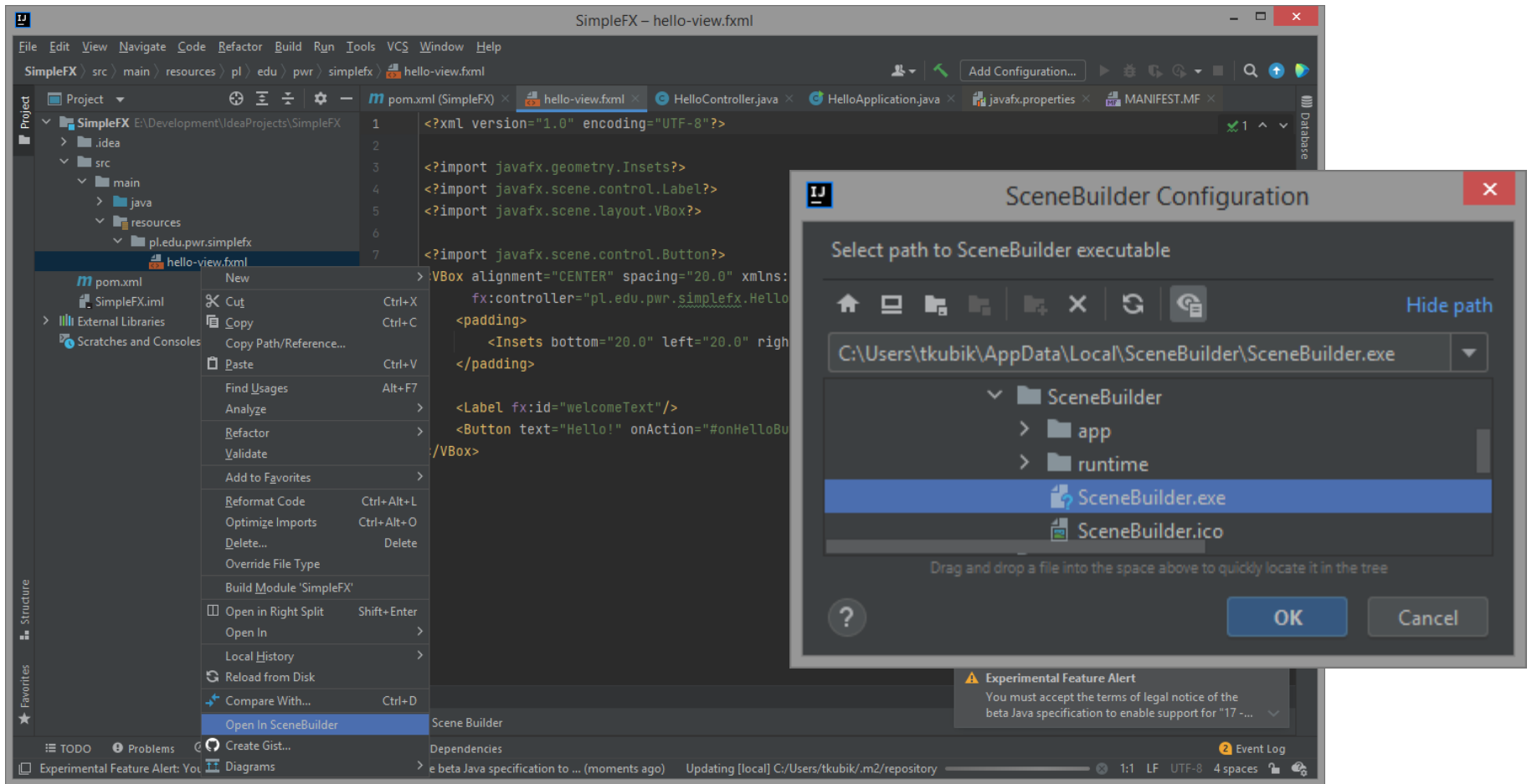
  <dependencies>
    <dependency>
      <groupId>org.openjfx</groupId>
      <artifactId>javafx-controls</artifactId>
      <version>11</version>
    </dependency>
    <dependency>
      <groupId>org.openjfx</groupId>
      <artifactId>javafx-fxml</artifactId>
      <version>11</version>
    </dependency>
    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter-api</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.junit.jupiter</groupId>
      <artifactId>junit-jupiter-engine</artifactId>
      <version>${junit.version}</version>
      <scope>test</scope>
    </dependency>
  </dependencies>

  <build>
    <plugins>
      <plugin>
        <groupId>org.apache.maven.plugins</groupId>
        <artifactId>maven-compiler-plugin</artifactId>
        <version>3.8.1</version>
        <configuration>
          <source>11</source>
          <target>11</target>
        </configuration>
      </plugin>
      <plugin>
        <groupId>org.openjfx</groupId>
        <artifactId>javafx-maven-plugin</artifactId>
        <version>0.0.6</version>
        <configuration>
          <stripDebug>true</stripDebug>
          <compress>2</compress>
          <noHeaderFiles>true</noHeaderFiles>
          <noManPages>true</noManPages>
          <jlinkImageName>simplefx</jlinkImageName>
          <launcher>launcher</launcher>
          <jlinkZipName>simplefxzip</jlinkZipName>
        </configuration>
      </plugin>
    </plugins>
  </build>

  <mainClass>pl.edu.pwr.simplefx/pl.edu.pwr.simplefx.HelloApplication</mainClass>
</project>
```

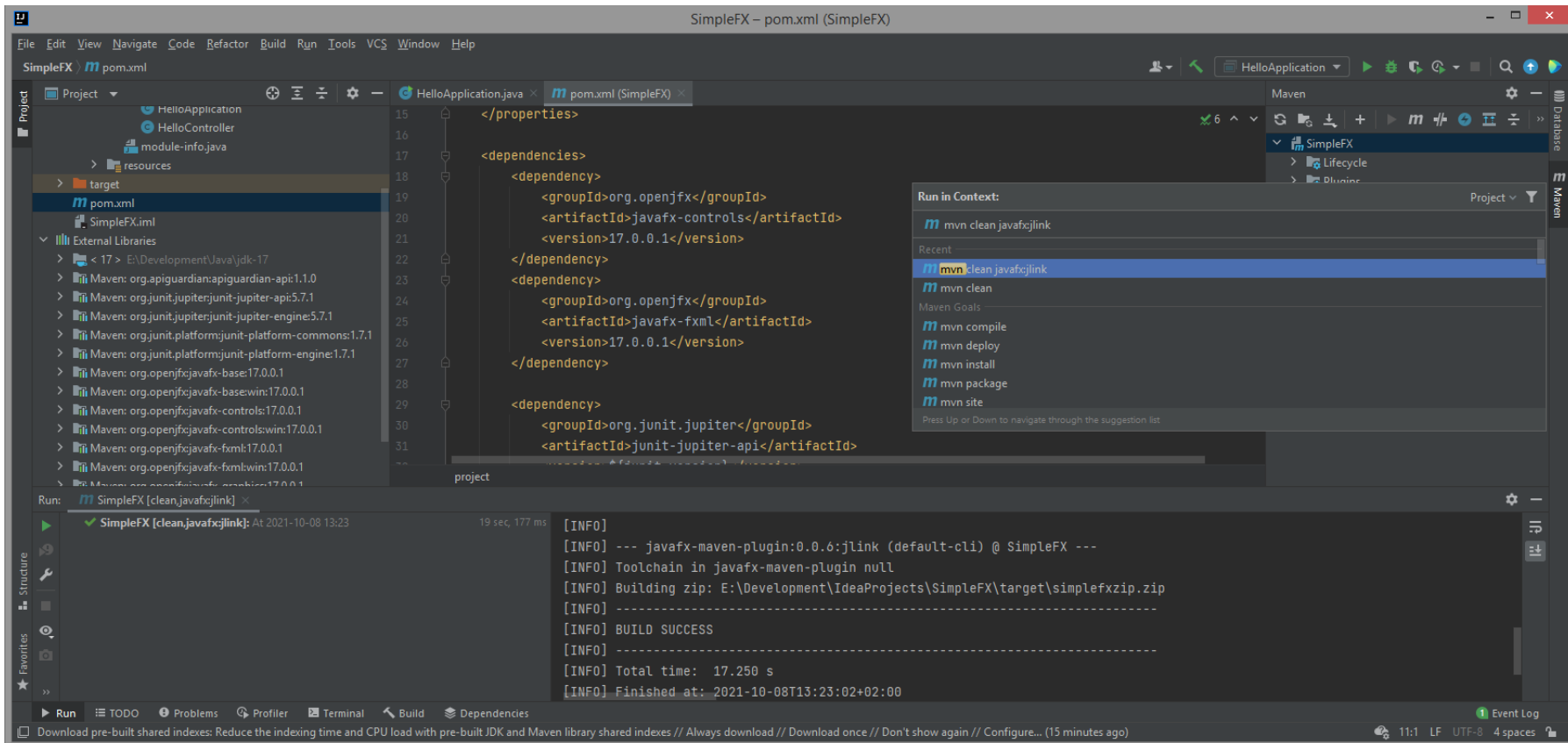
IntelliJ IDEA - JavaFX

- Aby dało się uruchomić SceneBuilder należy ustawić odpowiednią ścieżkę (SceneBuilder dostępny jest też jako plugin)



IntelliJ IDEA – JavaFX

- Pakowanie aplikacji z użyciem plugina mavenowego i jlink działa dla
 - JavaFX 17.0.0.1
 - JavaFX 11



<https://www.jetbrains.com/help/idea/packaging-javafx-applications.html>

<https://openjfx.io/openjfx-docs/#modular>

Porada

- Narzędzia deweloperskie takie jak eclipse czy IntelliJ zbudowane są w strukturach zawierających wiele katalogów i plików. Zwykle uruchamianie tych środowisk zajmuje sporo czasu, zwłaszcza gdy uruchamia się je po raz pierwszy po restarcie. Można nieco skrócić czas uruchomienia definiując wykluczenia dla programu antywirusowego (chodzi o to, by program nie skanował przy każdym uruchomieniu katalogu, w którym znajdują się pliki danego narzędzia).

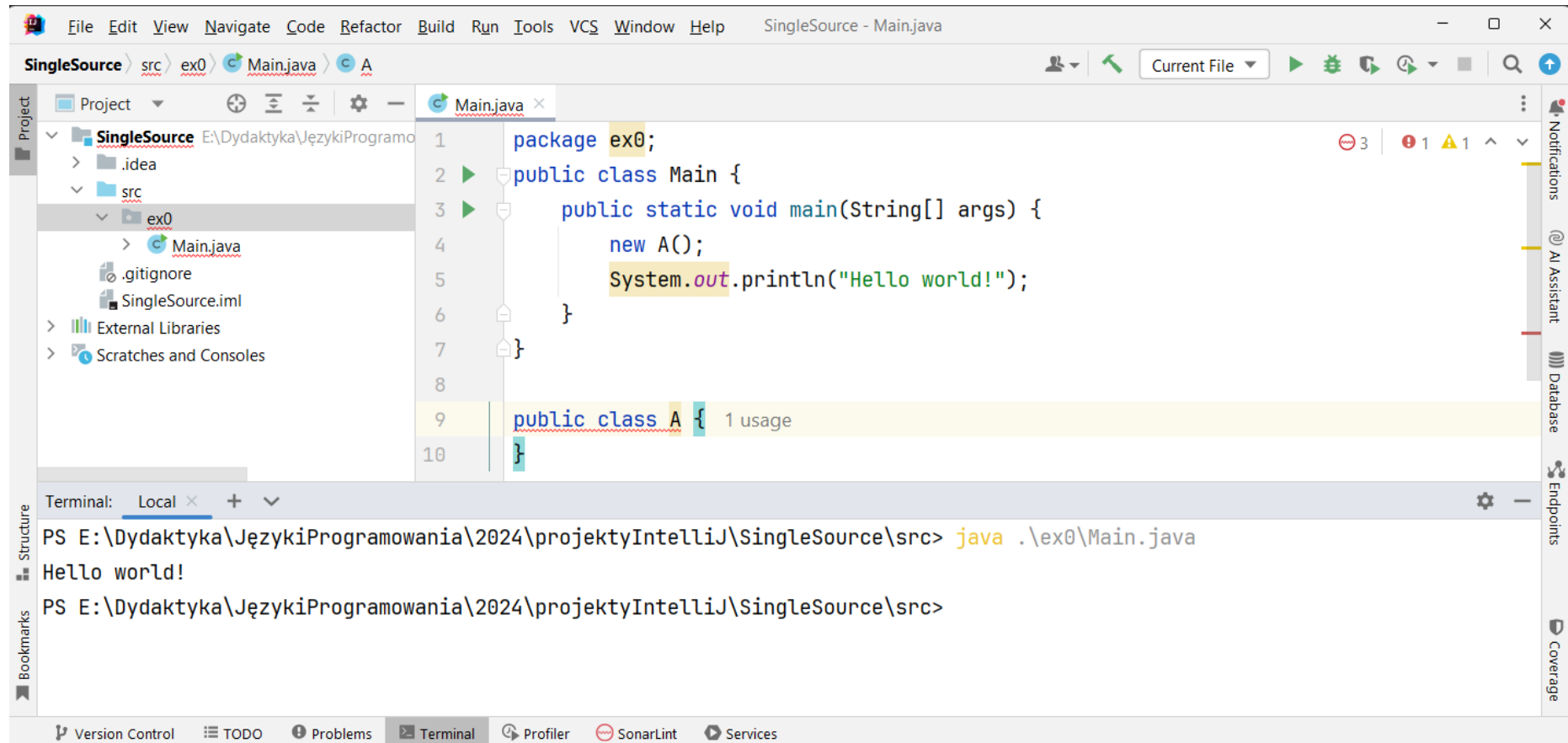
Win 11: Aby dodać wykluczenie

1. Przejdź do obszaru **ustawienia > aktualizacja i zabezpieczenia > wirusem & ochrona przed zagrożeniami**.
2. W obszarze **Ochrona przed zagrożeniami & zagrożeń** wybierz pozycję **Zarządzaj ustawieniami**.
3. W obszarze **wykluczenia** wybierz pozycję **Dodaj lub Usuń wykluczenia**.
4. Wybierz pozycję **Dodaj wykluczenie**
5. Wybierz jedną z czterech opcji w zależności od typu wykluczenia, które chcesz dodać:
 - **Plik** — wyklucza określony plik
 - **Folder** -wykluczanie określonego folderu (i wszystkich plików w tym folderze)
 - **Typ pliku** — wyklucza wszystkie pliki określonego typu, na przykład. DOCX lub. Formatach.
 - **Process** — wyklucza wszystkie pliki otwarte przez określony proces. Jeśli na przykład wyłączysz proces test.exe, wszelkie pliki otwarte przez proces test.exe będą wykluczone.

Uruchamianie programów napisanych w języku JAVA

- Standardowe sposoby:
 - przez wskazanie klasy znajdującej się w ścieżce klas (pliku `.class`)
`java -cp . ex.A`
 - przez wskazanie pliku `jar`, w manifeście którego określono główną klasę
`java -jar app.jar`
 - przez wskazanie klasy z modułu znajdującego się w ścieżce modułów
`java -p . -m module/ex.A`
 - przez uruchomienie klasy znajdującej się w **źródle kodu**
`java ex.A.java`
- Ostatni ze sposobów opisano w specyfikacji *JEP 330: Launch Single-File Source-Code Programs*. Zgodnie tą specyfikacją polecenie `java` może być wykorzystane do bezpośredniego uruchomienia kodu źródłowego jakiejś klasy bez etapu jawnej jego kompilacji. Jednak kod tej klasy powinien pojawić się jako pierwszy w źródle, klasa powinna być publiczna i zawierać metodę `main()`, od której zacząć się ma wykonywanie programu (pozostałe klasy w tym kodzie źródłowym też mogą być publiczne).

JEP 330 – problem zaburzenia reguł edycji źródeł kodu



The screenshot displays an IDE window titled "SingleSource - Main.java". The main editor shows the following Java code:

```
1 package ex0;
2 public class Main {
3     public static void main(String[] args) {
4         new A();
5         System.out.println("Hello world!");
6     }
7 }
8
9 public class A { 1 usage
10 }
```

The code for `public class A` is highlighted in yellow. The IDE interface includes a project view on the left showing the file structure, a terminal at the bottom, and various toolbars and panels on the right.

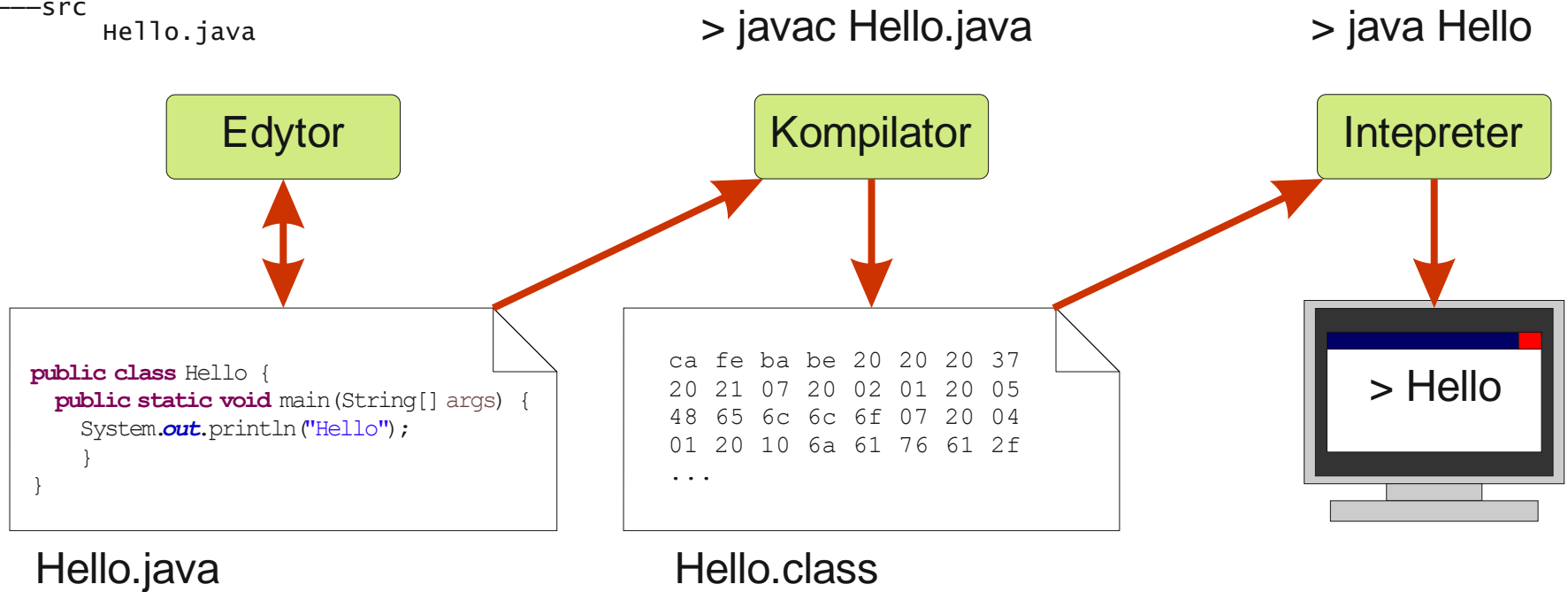
The terminal window shows the following output:

```
PS E:\Dydaktyka\JęzykiProgramowania\2024\projektyIntelliJ\SingleSource\src> java .\ex0\Main.java
Hello world!
PS E:\Dydaktyka\JęzykiProgramowania\2024\projektyIntelliJ\SingleSource\src>
```

Wytwarzanie oprogramowania w języku JAVA

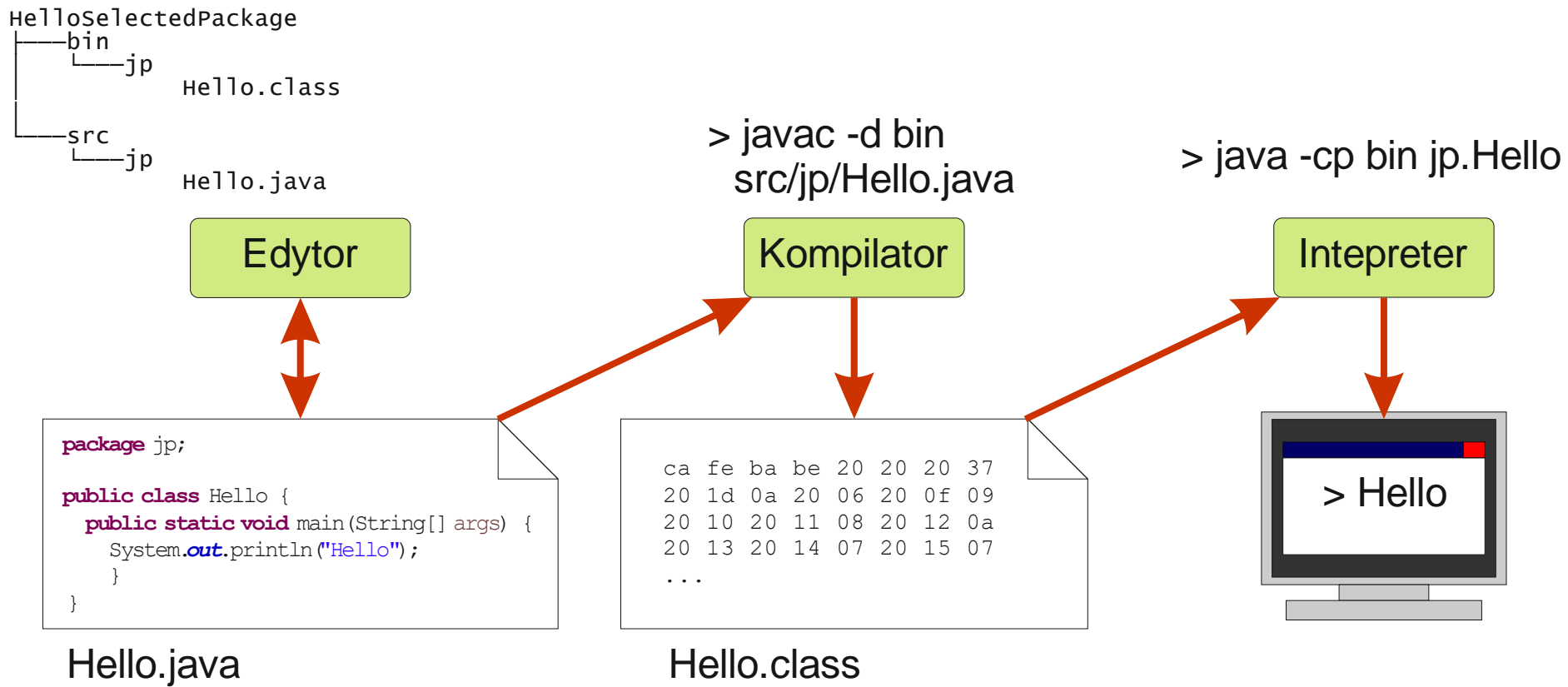
- podejście standardowe (klasa w pakiecie domyślnym)

```
HelloDefaultPackage  
├── bin  
│   └── Hello.class  
└── src  
    └── Hello.java
```



Wytwarzanie oprogramowania w języku JAVA

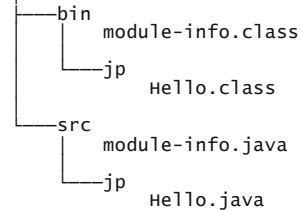
- podejście standardowe (klasa w pakiecie zdefiniowanym)



Wytwarzanie oprogramowania w języku JAVA

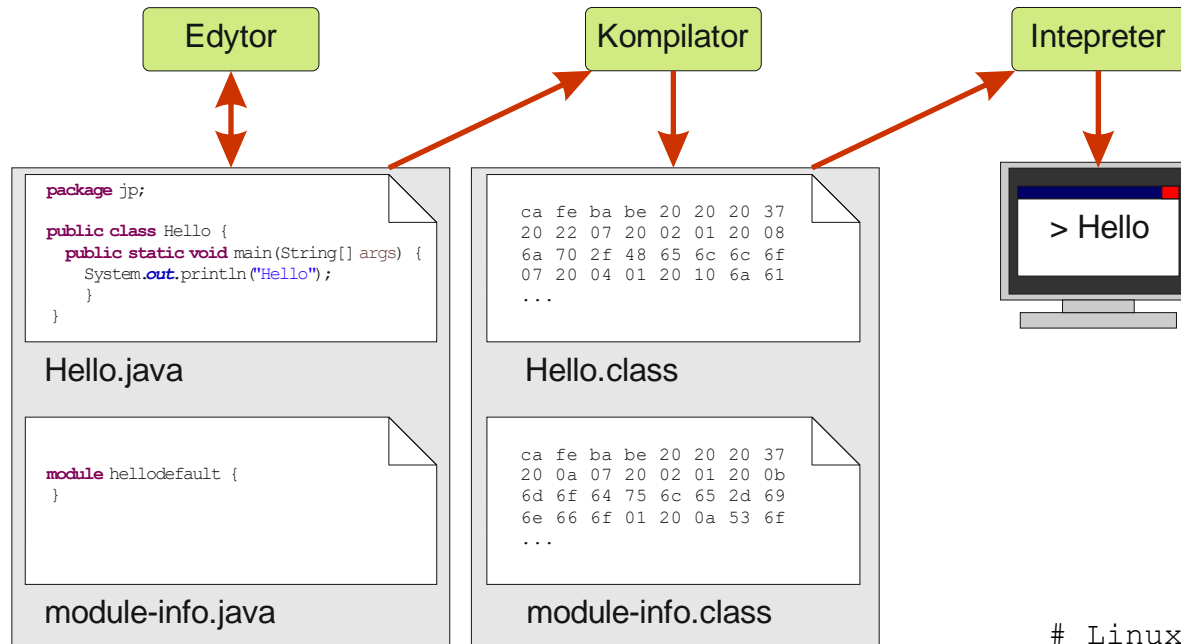
- podejście z jednym, domyślnym modułem

HelloSingleJPMS:



```
> javac -d bin
src\jp\Hello.java
src\module-info.java
```

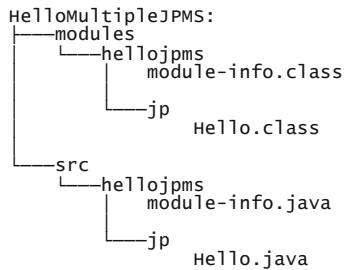
```
> java --module-path bin
--module hellodefault/jp.Hello
```



```
# Linux / MacOS
$ find -name "*.java" > sources.txt
$ javac @sources.txt
```

```
:: Windows
> dir /s /B *.java > sources.txt
> javac @sources.txt
```

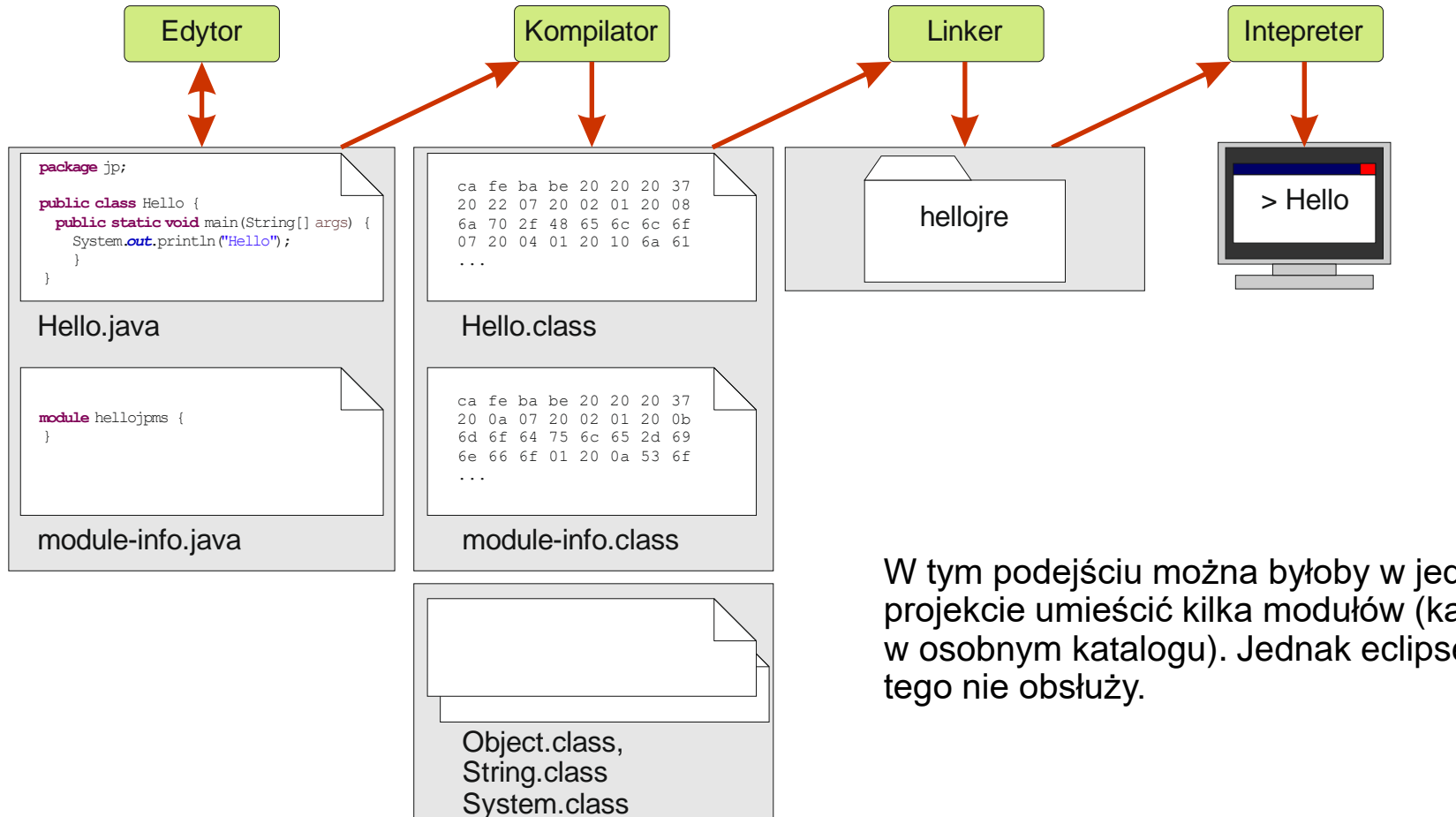
Wytwarzanie oprogramowania w języku JAVA - podejście z osobnym katalogiem dla modułu



```
> javac --module-source-path src  
-d modules -m hellojpm
```

```
> jlink --module-path modules  
--add-modules  
hellojpm,java.base  
--output hellojre
```

```
> hellojre/bin/java  
-m hellojpm/jp.Hello
```



W tym podejściu można byłoby w jednym projekcie umieścić kilka modułów (każdy w osobnym katalogu). Jednak eclipse tego nie obsługuje.

Pliki metadanych

- `module-info.java`
 - zawiera informacje o zależnościach modułowych, może zawierać dokumentację modułu

```
module SimpleFX {  
  requires javafx.controls;  
  requires javafx.fxml;  
  
  opens application to javafx.graphics, javafx.fxml;  
}
```

```
{@author}, {@deprecated}, {@provides},  
{@see}, {@since}, {@serialField}, {@uses},  
{@version} {@code}, {@docRoot}, {@index},  
{@link}, {@linkplain}, {@literal}, {@summary}
```

<https://stackoverflow.com/questions/53957486/javadoc-comments-in-module-info>

- `package-info.java`
 - zwykle zawiera dokumentację pakietu oraz adnotacje wykorzystywane przez różne frameworki,

```
/**  
 * @author tkubik  
 *  
 */  
package application;
```

```
@NonNullApi  
@NonNullFields  
package com.baeldung.nullibility;  
  
import org.springframework.lang.NonNullApi;  
import org.springframework.lang.NonNullFields;
```

<https://www.baeldung.com/java-package-info>

Przydatne linki

- Tworzenie i kompilacja projektów modułowych
 - <https://www.logicbig.com/tutorials/core-java-tutorial/modules/modes.html>
 - <https://www.baeldung.com/java-9-modularity>
 - <https://www.developer.com/java/ent/using-a-java-9-module-as-a-jar-file.html>
 - <https://www.geeksforgeeks.org/jlink-java-linker/>
- Java + JavaFX modularnie
 - <https://openjfx.io/openjfx-docs/modular>
- Uruchamianie projektów jednoplukowych poleceniem `java` bez konieczności jawnej kompilacji
 - <https://www.infoq.com/articles/single-file-execution-java11/>
- Specyfikacja Java
 - <https://docs.oracle.com/javase/specs/>
- Specyfikacja linkera
 - <https://openjdk.java.net/jeps/282>