TESTOWANIE OPROGRAMOWANIA
I
TEST DRIVEN DEVELOPMENT

Autor: Michał Bojanowski

JUnit - executables

Czym jest executable?

```
import org.junit.jupiter.api.function.Executable;
```

```
@FunctionalInterface
@API(status = STABLE, since = "5.0")
public interface Executable {
   void execute() throws Throwable;
}
```

```
public class ExecutableExample implements Executable {
    @Override
    public void execute() {
        assertEquals(2, 3);
    }
}
```

Autor: Michał Bojanowski

JUnit – inny sposób użycia

```
@Test
void testThatDoesNothingButShowsHowToUseExecutables() {
    final Executable checkOneExecutable = () -> assertEquals(1, 1);
    final Executable checkTwoExecutable = () -> assertTrue(4 % 3 == 1);

    assertAll(checkOneExecutable,
              checkTwoExecutable,
              new ExecutableExample());
}
```

Autor: Michał Bojanowski

Java – bloki statyczne

```java
public class StaticBlocks {

    private static List<String> staticList;
    private List<String> notStaticList;

    static {
        // to mogę zrobić bo staticList jest statyczne
        staticList = Arrays.asList("STR1", "STR2", "STR3");

        // tego nie mogę zrobić. Tu mogę zmieniać jedynie zmienne statyczne
        // notStaticList = new ArrayList<>();
        System.out.println("BLOK STATYCZNY");
    }

    public StaticBlocks() {
        // tu mogę edytować zmienne statyczne i niestatyczne
        notStaticList = new ArrayList<>();
        staticList.contains("STR1");
        System.out.println("KONSTRUKTOR");
    }
}
```

```
BLOK STATYCZNY
KONSTRUKTOR
```

Autor: Michał Bojanowski

Java – mapy

Czym jest mapa?

- Kontener przechowujący pary. Wszystkie pary to tzw. EntrySet
- Każda para to tzw. Entry
- Każda para, tzn. Entry, składa się z klucza (key) i wartości (value)
- Kazdy klucz może pojawić się w mapie tylko raz

Autor: Michał Bojanowski

Java – mapy, klucze i wartości

```
final Map<Integer, String> indexToMonthName = new HashMap<>();
indexToMonthName.put(1, "STYCZEN");
indexToMonthName.put(2, "LUTY");
indexToMonthName.put(3, "GRUDZIEN");
```

```
indexToMonthName.get(1);
```

```
boolean containsKeyTwo = indexToMonthName.containsKey(2);
```

```
boolean containsStyczen = indexToMonthName.containsValue("STYCZEN");
```

Autor: Michał Bojanowski

Java – mapy, cd.

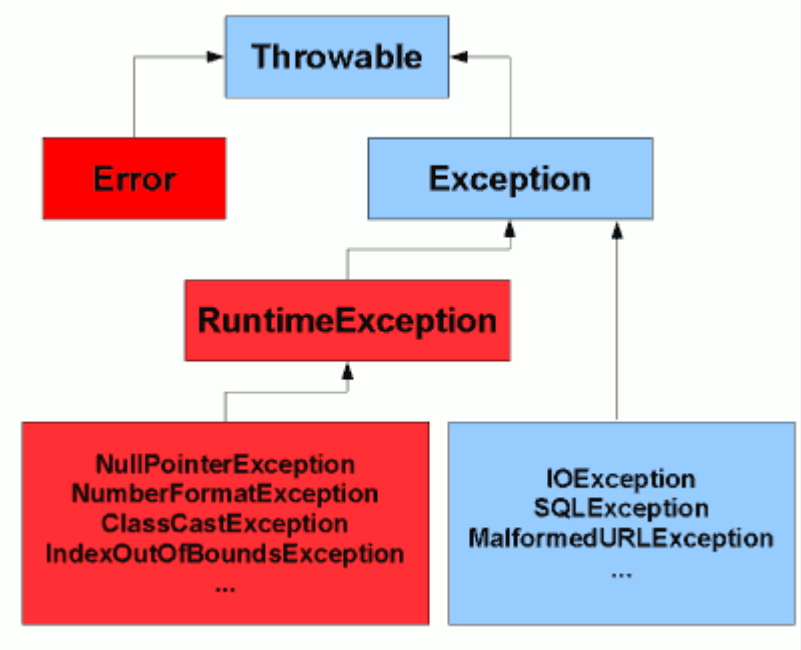
```
indexToMonthName.entrySet().forEach(entry ->
System.out.println(entry.getKey() + " " + entry.getValue()));
```

```
indexToMonthName.forEach((key, value) -> System.out.println(key + " " + value));
```

```
1 STYCZEN
2 LUTY
3 GRUDZIEN
```

Autor: Michał Bojanowski

Java – wyjątki



Autor: Michał Bojanowski

Java – wyjątki

Każdy wyjątek posiada:

- Wiadomość wyjątku
- Przyczynę wyjątku (tzn. poprzedni wyjątek, który go wyrzucił
- StackTrace: stos wywołań metod, który przyczynił się do wyrzucenia wyjątku

```
new Throwable().getMessage();
new Throwable().getCause();
new Throwable().getStackTrace();
```

Autor: Michał Bojanowski

Java – wyjątki

```
public static void showingExceptionWithCause() {
    try {
        new FibonacciSeries().compute(-1);
    } catch (final IllegalArgumentException exp) {
        System.err.println("Failed to compute fibonacci serie value");
        throw new FibonacciSeriesException(exp.getMessage(), exp);
    }
}
```

Oryginalny exp (IllegalArgumentException) jest POWODEM (tzw. cause) wyrzucenia wyjątku FibonacciSeriesException

Autor: Michał Bojanowski

Java – cause

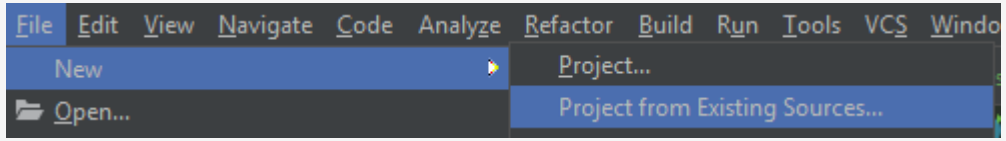
```java
public static void showingExceptionWithCause() {
    try {
        new FibonacciSeries().compute(-1);
    } catch (final IllegalArgumentException exp) {
        System.err.println("Failed to compute fibonacci serie value");
        throw new FibonacciSeriesException(exp.getMessage(), exp);
    }
}
```

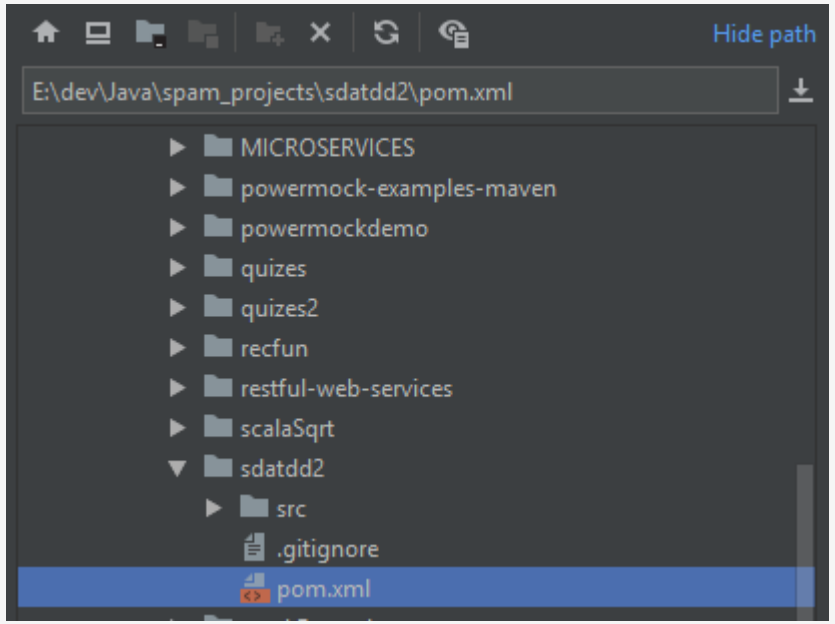
```
"C:\Program Files (x86)\Java\jdk1.8.0_131\bin\java.exe" ...
Failed to compute fibonacci serie value

pl.sdacademy.exceptions.FibonacciSeriesException: Index has to be positive

    at pl.sdacademy.ExceptionsExample.showingExceptionWithCause(ExceptionsExample.java:13)
    at pl.sdacademy.ExceptionsExampleTest.test(ExceptionsExampleTest.java:11) <15 internal calls>
    at java.util.ArrayList.forEach(ArrayList.java:1249) <5 internal calls>
    at java.util.ArrayList.forEach(ArrayList.java:1249) <17 internal calls>
Caused by: java.lang.IllegalArgumentException: Index has to be positive
    at pl.sdacademy.calculations.FibonacciSeries.compute(FibonacciSeries.java:7)
    at pl.sdacademy.ExceptionsExample.showingExceptionWithCause(ExceptionsExample.java:10)
    ... 40 more
```

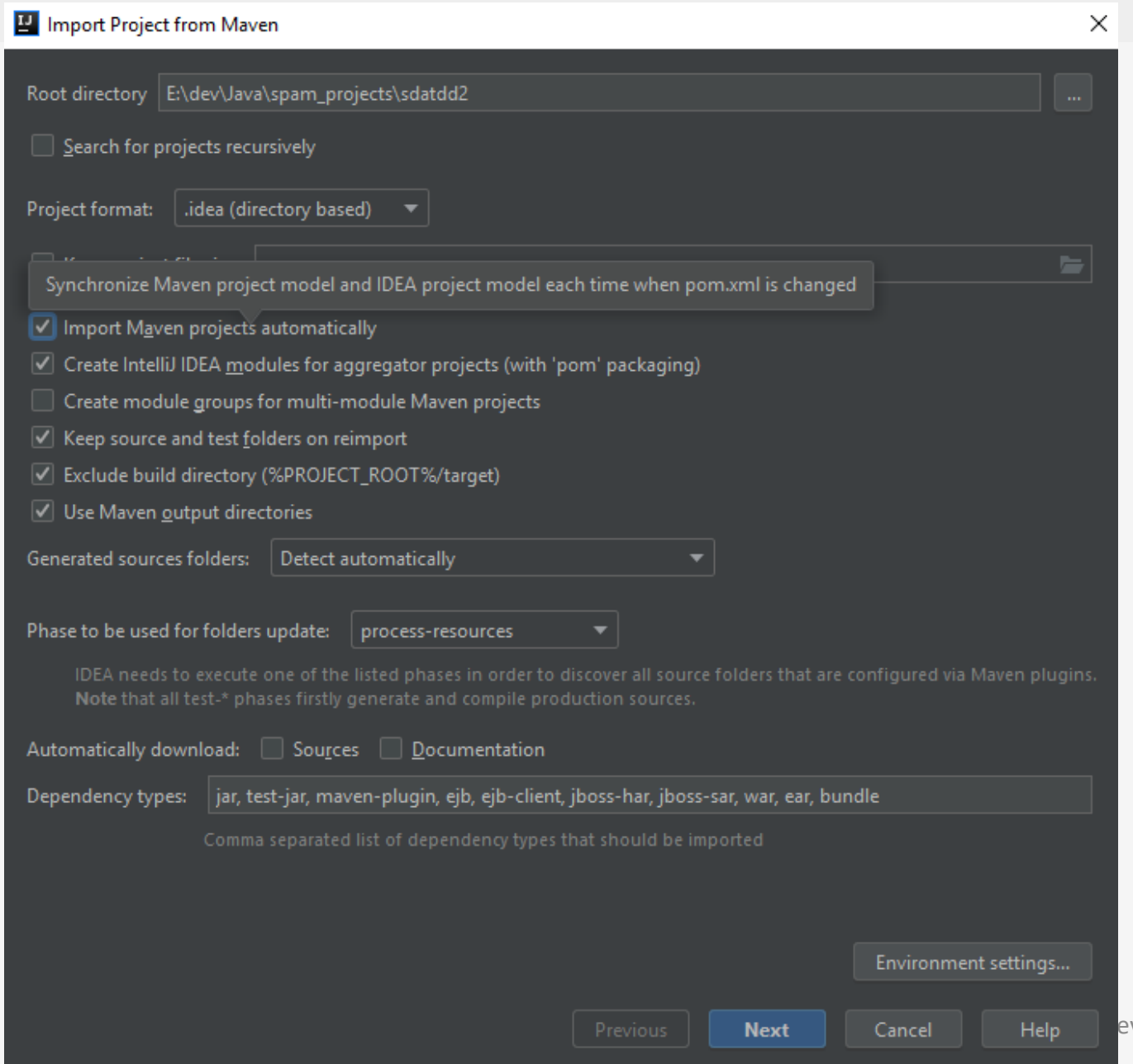
IDEA – poprawne importowanie projektów



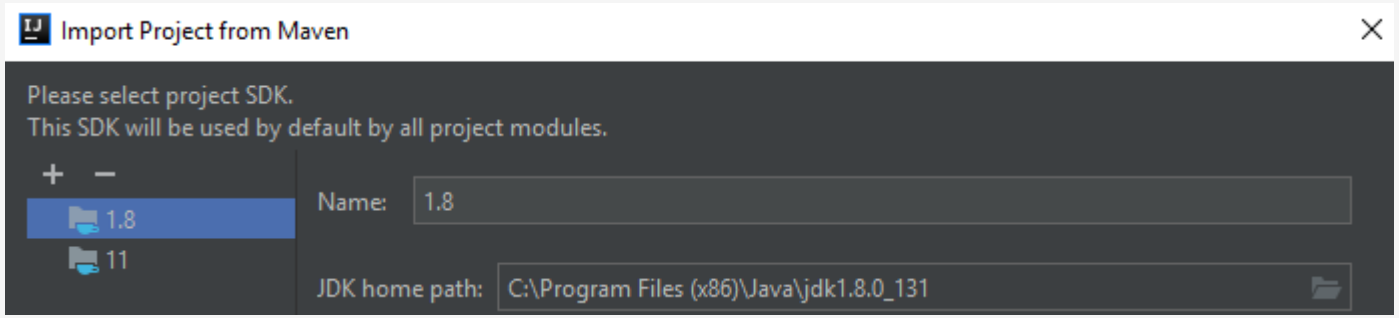


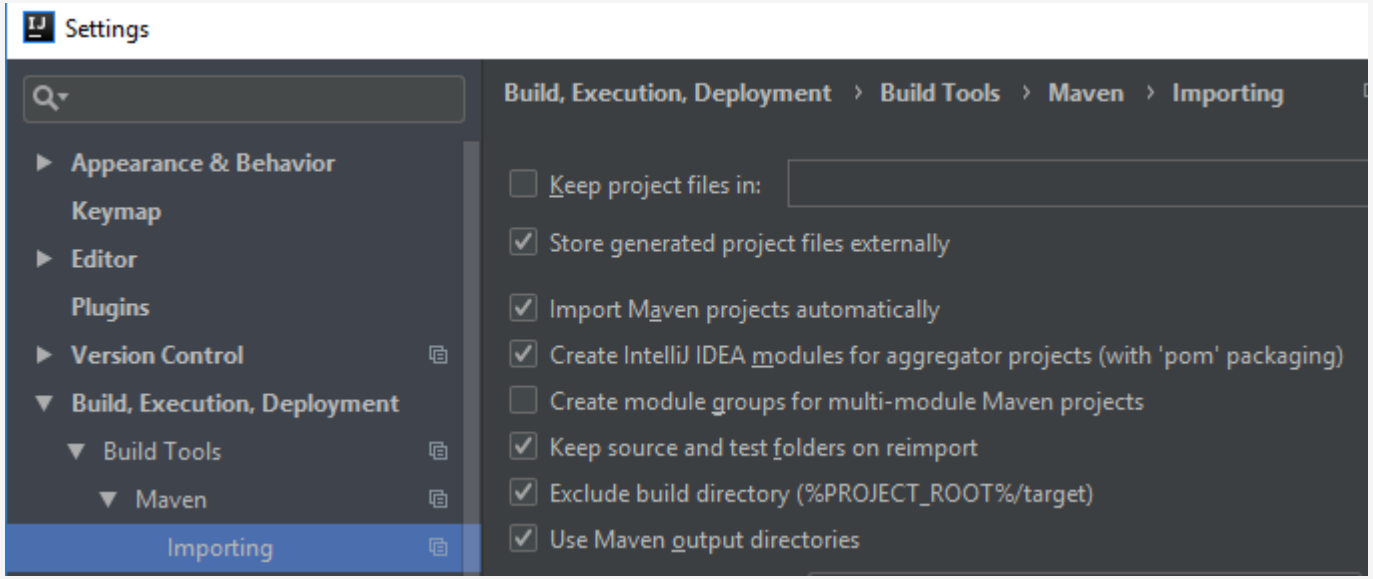
Autor: Michał Bojanowski

IDEA – poprawne importowanie projektów



IDEA – poprawne importowanie projektów





Autor: Michał Bojanowski

IDEA – wymuszenie reimportu

CTRL + SHIFT + A

Autor: Michał Bojanowski