



**TRANSITION**  
TECHNOLOGIES

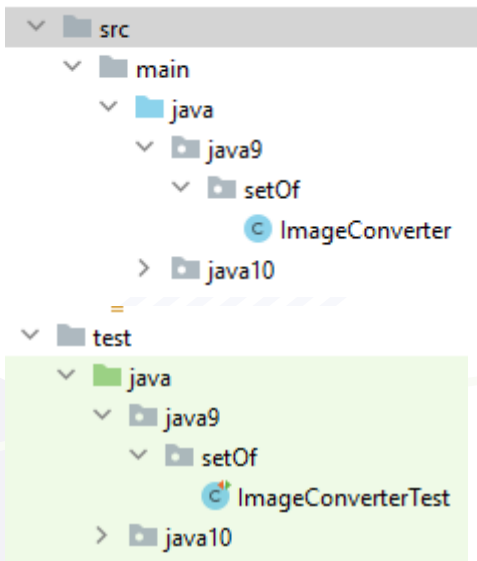
# Java 9-17 Features

**Michał Krysiuk**

Transition Technologies PSC

# Repo

- Exercises to each Java version
- Useful comments
- Written tests for exercises










```
String getUsername(int id) {
    // implement here
    // get username from DataProvider.
    // Return value or throw exception if there is no value

    return "";
}
```

```
@Test
void shouldReturnUsernameForIdGreaterThan3() {
    String username = dataService.getUsername(id: 4);

    assertEquals(expected: "user-4", username);
}
```

PROCEDURE						
						
Drain the washer fluid tank		Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 9

# Java 9

- **Modular system**
- **New HTTP Client**
- **Interface Private Methods**
- **JShell**
- **Collection factory methods**



# Modular system

## java.base

```
// module-info.java
Module java.base {
    exports java.lang;
    exports java.io;
    exports java.net;
    exports java.util;
}
```

### Exported Packages

java.io  
java.time  
java.util  
java.net  
java.lang

### Concealed Packages

sun.security.provider  
sun.nio.ch  
sun.reflect.annotation

# New HTTP Client

```
// using HttpURLConnection class
URL url = new URL("https://jsonplaceholder.typicode.com/todos/1");
HttpURLConnection con = (HttpURLConnection) url.openConnection();
con.setRequestMethod("GET");

con.setRequestProperty("header-name", "header-value");

BufferedReader in = new BufferedReader(
    new InputStreamReader(con.getInputStream()));
String inputLine;
StringBuffer content = new StringBuffer();
while ((inputLine = in.readLine()) != null) {
    content.append(inputLine);
}
in.close();

String response = content.toString();
```

```
// using HttpClient API
HttpRequest request = HttpRequest.newBuilder()
    .uri(new URI("https://jsonplaceholder.typicode.com/todos/1"))
    .GET()
    .header("header-name", "header-value")
    .build();

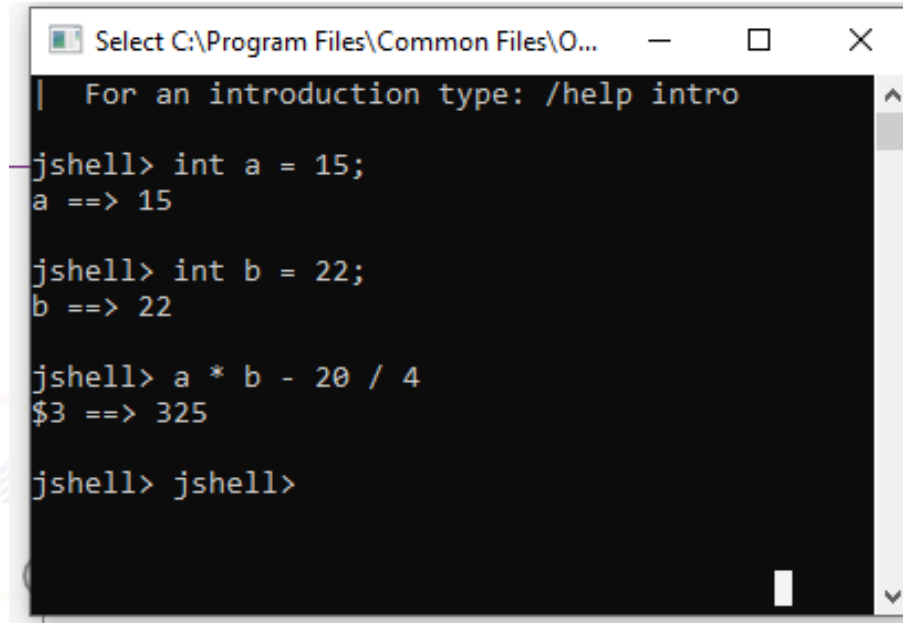
HttpResponse<String> response = HttpClient.newHttpClient()
    .send(request, HttpResponse.BodyHandler.asString());
```

# Interface Private Methods

```
public interface InterfaceWithPrivateMethod {  
    default void processMethod(Integer input) {  
        // many code lines  
        // many code lines  
        // many code lines  
        // many code lines  
        // many code lines  
        Integer someConstant = input * 9;  
        // many code lines  
        // many code lines  
        // many code lines  
        // many code lines  
        if (someConstant < 1) {  
            throw new RuntimeException("Value lower than 1!");  
        } else if (someConstant.equals(81)) {  
            throw new RuntimeException("Value equals to 81!");  
        } else if (someConstant > 99) {  
            throw new RuntimeException("Value greater than 99!");  
        }  
    }  
}
```

```
public interface InterfaceWithPrivateMethod {  
    private static Integer getSomeNamedConstant() {  
        return 9;  
    }  
  
    default void processMethod(Integer input) {  
        // many code lines  
        // many code lines  
        // many code lines  
        // many code lines  
        // many code lines  
        Integer someConstant = input * getSomeNamedConstant();  
        // many code lines  
        // many code lines  
        // many code lines  
        // many code lines  
        verify(someConstant);  
    }  
  
    private void verify(Integer value) {  
        if (value < 1) {  
            throw new RuntimeException("Value lower than 1!");  
        } else if (value.equals(81)) {  
            throw new RuntimeException("Value equals to 81!");  
        } else if (value > 99) {  
            throw new RuntimeException("Value greater than 99!");  
        }  
    }  
}
```

## Java REPL



A screenshot of a JShell terminal window. The window title bar shows the path "Select C:\Program Files\Common Files\O...". The terminal content is as follows:

```
| For an introduction type: /help intro  
jshell> int a = 15;  
a ==> 15  
  
jshell> int b = 22;  
b ==> 22  
  
jshell> a * b - 20 / 4  
$3 ==> 325  
  
jshell> jshell>
```

The terminal has a black background with white text. A vertical scrollbar is visible on the right side of the terminal area.









# Collection factory methods

## List::of, Set::of

```
// immutable list  
List<Integer> list = List.of(1, 2, 3);
```

```
// immutable set  
Set<Integer> set = Set.of(1, 2, 3);
```

```
// when try to add element  
java.lang.UnsupportedOperationException  
at java.base/java.util.ImmutableCollections.uoe(ImmutableCollections.java:1  
at java.base/java.util.ImmutableCollections$AbstractImmutableCollection.add  
at java9.setOf.ImageConverter.getAvailableFileKeys(ImageConverter.java:14)  
at java9.setOf.ImageConverterTest.name(ImageConverterTest.java:16)  
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native M
```








PROCEDURE						
						
Drain the washer fluid tank		Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 9 Excercises





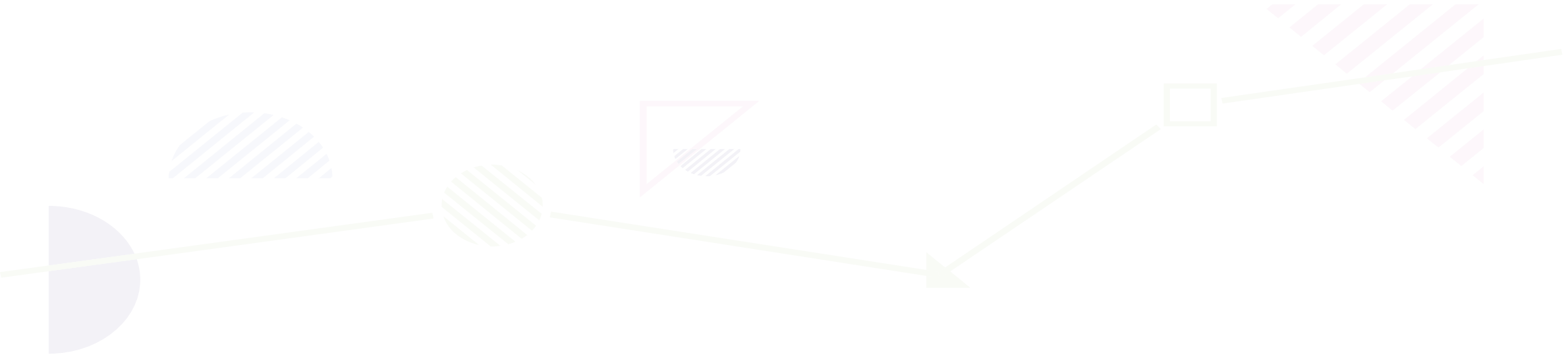
PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 10

# Java 10

- Local Variable Type Inference
- Unmodifiable Collections
- New Optional.orElseThrow() Method



# Local Variable Type Inference

```
// without var
String text = "example-text";
// text = 5;
User myUser = new User();
Map<String, Integer> myMap = new HashMap<>();
```

```
// with var
var text = "example-text";
// text = 5;
var myUser = new User();
var myMap = new HashMap<String, Integer>();
```

```
// illegal use
var x;
var x = null;

class User {
    var myField;
}
```

# Unmodifiable Collections

**Collectors::toUnmodifiableList**  
**Collectors::toUnmodifiableSet**  
**Collectors::toUnmodifiableMap**

**List::copyOf**  
**Set::copyOf**  
**Map::copyOf**

```
@Test
void shouldToUnmodifiableCollectorReturnUnmodifiableList() {
    List<Integer> integers = Arrays.asList(1, 2, 3);

    List<Integer> result = integers.stream()
        .collect(Collectors.toUnmodifiableList());

    assertThrows(UnsupportedOperationException.class, () -> {
        result.add(4);
    });
}

@Test
void shouldCopyOfReturnUnmodifiableList() {
    List<Integer> integers = Arrays.asList(1, 2, 3);

    List<Integer> result = List.copyOf(integers);

    assertThrows(UnsupportedOperationException.class, () -> {
        result.add(4);
    });
}
```

# New Optional.orElseThrow() Method

## Preferred alternative to get() method








```
Optional<Integer> integerOptional = Optional.of(1);  
Integer integer = integerOptional.orElseThrow();
```

java.util.NoSuchElementException: No value present














PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 10 Exercises





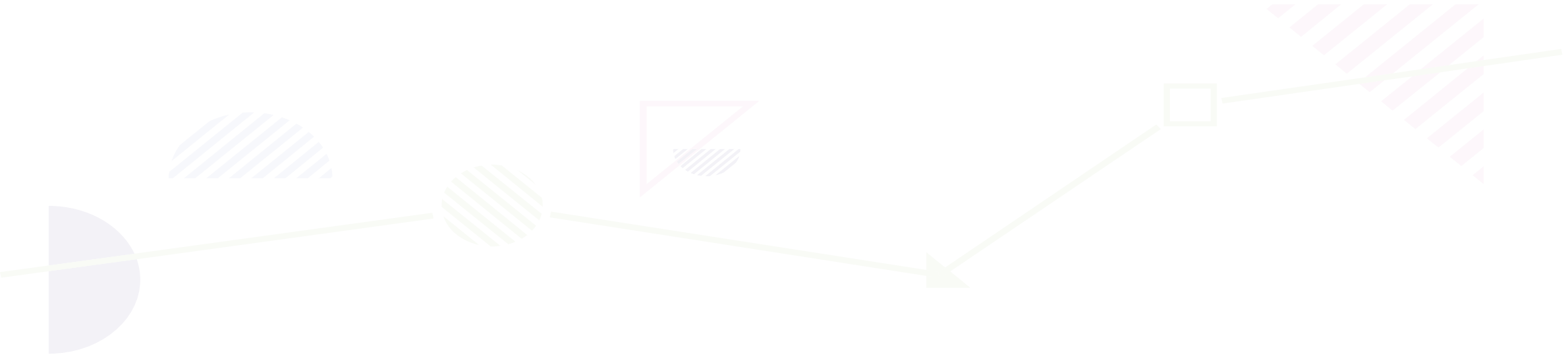
PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 11

# Java 11

- Many new String methods
- New Files methods
- Script running



# Many new String methods

## isBlank

```
"ABC".isBlank();    //false
" ABC ".isBlank();  //false
"  ".isBlank();     //true
"".isBlank();       //true
"\n".isBlank();     //true
"\r".isBlank();     //true
```

## strip stripLeading stripTrailing

```
" ABC ".strip();    // "ABC"
" ABC ".stripLeading(); // "ABC "
" ABC ".stripTrailing(); // "  ABC"
```

## lines

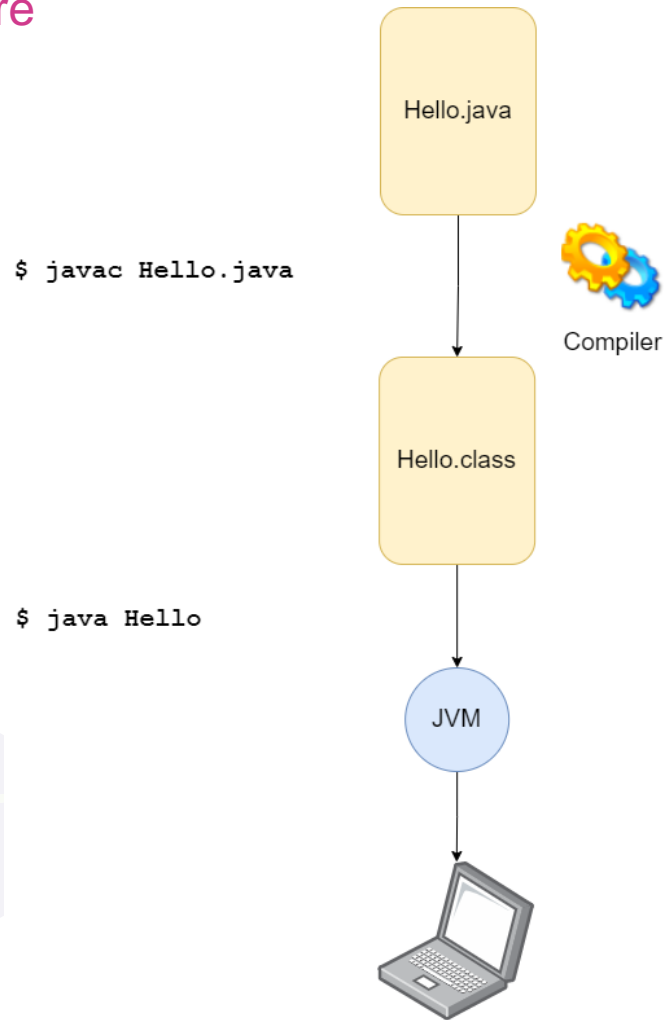
```
"String operations \r in Java \nare soooo \r\neasy".lines()
// "String operations", " in Java", " are soooo", "easy"
```

## repeat

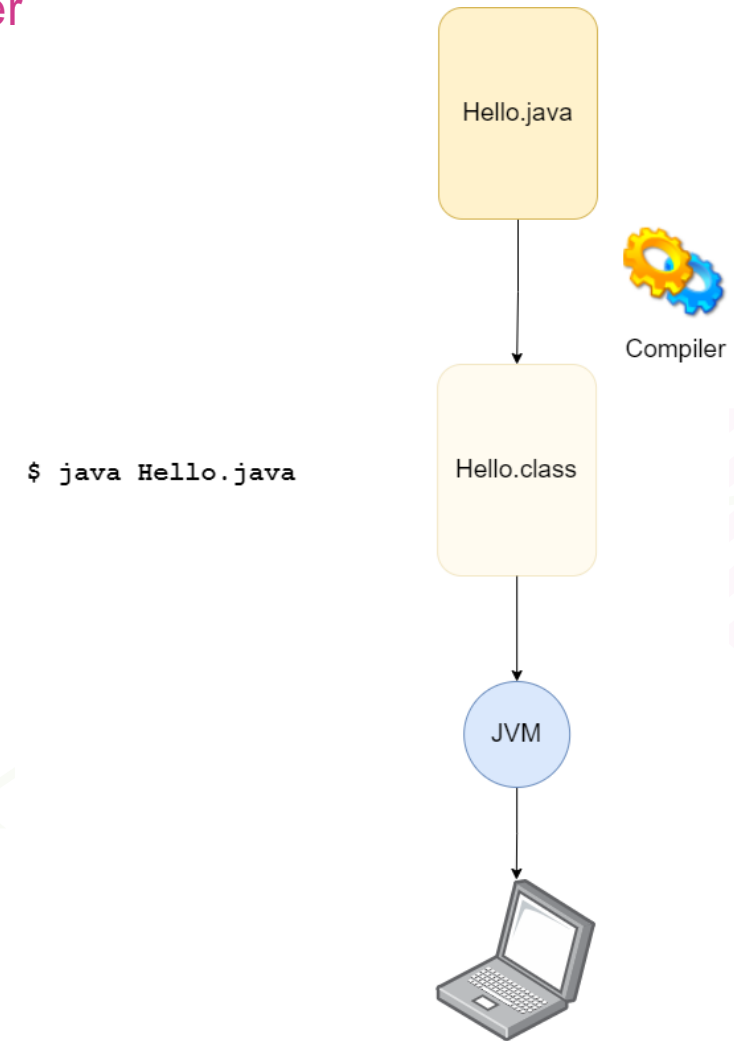
```
"ha".repeat(10); // "hahahahahahahahaha"
```

# Script running

Before



After



# New Files methods

Files::writeString

Files::readString








```
Path file = Files.createTempFile( prefix: "filename", suffix: ".txt");
```

```
Files.writeString(file, csq: "Some text");
```

```
String fileContent = Files.readString(file);
```












PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 11 Exercises



PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 12

# Java 12

- New String methods
- Teeing Collector
- `File::mismatch` method





# New String methods

## indent(int n)

```
String text = "Hello\nCode!";  
System.out.println(text);  
  
String textWithIndent = text.indent(10);  
System.out.println(textWithIndent);
```

Hello  
Code!

Hello  
Code!

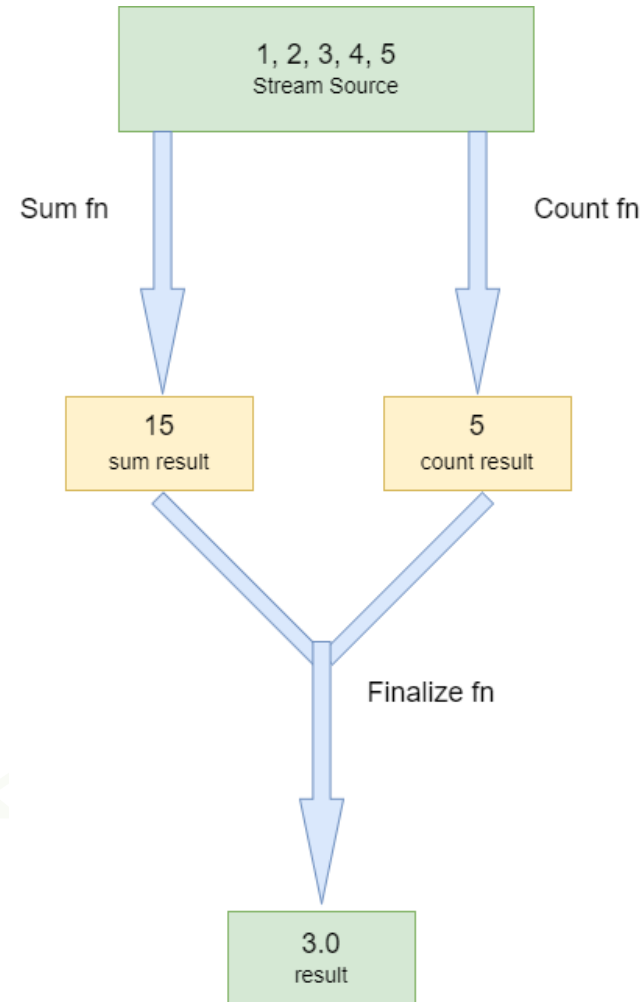
## transform

```
String text = "Some text";  
  
String transformed = text  
    .transform(s -> s.replace(target: "Some", replacement: "Any"));  
System.out.println(transformed);
```

Some text  
Any text

# Teeing Collector

```
double mean = Stream.of(1, 2, 3, 4)
    .collect(Collectors.teeing(
        Collectors.summingDouble(i -> i),
        Collectors.counting(),
        (sum, count) -> sum / count
    ));
```










# File::mismatch method

**mismatch(Path path, Path path2) throws IOException**

```
Path filePath1 = Files.createTempFile( prefix: "file1", suffix: ".txt");  
Files.writeString(filePath1, csq: "Java 12 is awesome");  
  
Path filePath2 = Files.createTempFile( prefix: "file2", suffix: ".txt");  
Files.writeString(filePath2, csq: "Java 12 is amazing");  
  
long mismatch = Files.mismatch(filePath1, filePath2);  
// 12
```










PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 12 Exercises



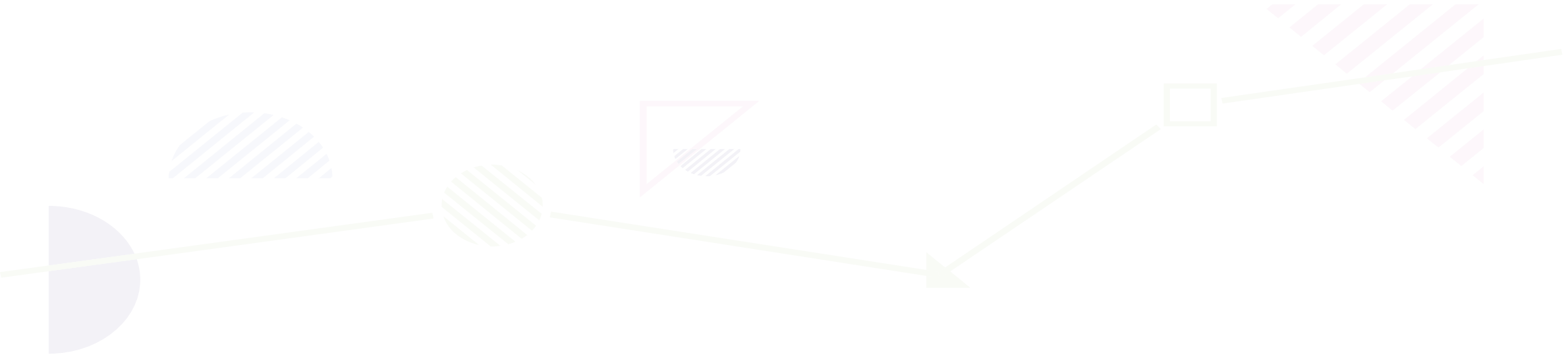
PROCEDURE						
						
Drain the washer fluid tank		Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274










# Java 13

# Java 13

- Switch Expressions update - Preview Feature
- Text Blocks – Preview Feature
- Z Garbage Collector improvements
- Minor Java API changes





PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 14

# Java 14

- Switch Expressions
- NullPointerExceptions update





# Switch Expressions

## Before

```
String season = "Winter";

String translation;
switch (season) {
    case "Spring": {
        translation = "wiosna";
        break;
    }
    case "Summer": {
        translation = "lato";
        break;
    }
    case "Autumn": {
        translation = "jesień";
        break;
    }
    case "Winter": {
        translation = "zima";
        break;
    }
    default: {
        translation = "nieznany";
    }
}
```

- Arrow case
- Eliminate the need of break statement

## After

```
String season = "Winter";

String translation = switch (season) {
    case "Spring" -> "wiosna";
    case "Summer" -> "lato";
    case "Autumn" -> "jesień";
    case "Winter" -> "zima";
    default -> "nieznany";
};
```

# NullPointerException update

## Code

```
List<Student> warsawStudents = students.stream()  
    .filter(s -> s.getAddress().getCity().equals("Warszawa"))  
    .collect(Collectors.toList());
```



# NullPointerException update








## Before

```
Exception in thread "main" java.lang.NullPointerException
  at Test.lambda$main$0(scratch_2.java:114)
  at java.util.stream.ReferencePipeline$2$1.accept(ReferencePipeline.java:174)
  at java.util.ArrayList$ArrayListSpliterator.forEachRemaining(ArrayList.java:1382)
  at java.util.stream.AbstractPipeline.copyInto(AbstractPipeline.java:482)
  at java.util.stream.AbstractPipeline.wrapAndCopyInto(AbstractPipeline.java:472)
  at java.util.stream.ReduceOps$ReduceOp.evaluateSequential(ReduceOps.java:708)
  at java.util.stream.AbstractPipeline.evaluate(AbstractPipeline.java:234)
  at java.util.stream.ReferencePipeline.collect(ReferencePipeline.java:499)
  at Test.main(scratch_2.java:115)
```

## After

```
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "Address.getCity()" because the return value
of "Student.getAddress()" is null
  at Test.lambda$main$0(scratch_2.java:114)
  at java.base/java.util.stream.ReferencePipeline$2$1.accept(ReferencePipeline.java:178)
  at java.base/java.util.ArrayList$ArrayListSpliterator.forEachRemaining(ArrayList.java:1625)
  at java.base/java.util.stream.AbstractPipeline.copyInto(AbstractPipeline.java:509)
  at java.base/java.util.stream.AbstractPipeline.wrapAndCopyInto(AbstractPipeline.java:499)
  at java.base/java.util.stream.ReduceOps$ReduceOp.evaluateSequential(ReduceOps.java:921)
  at java.base/java.util.stream.AbstractPipeline.evaluate(AbstractPipeline.java:234)
  at java.base/java.util.stream.ReferencePipeline.collect(ReferencePipeline.java:682)
  at Test.main(scratch_2.java:115)
```










PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 14 Exercises





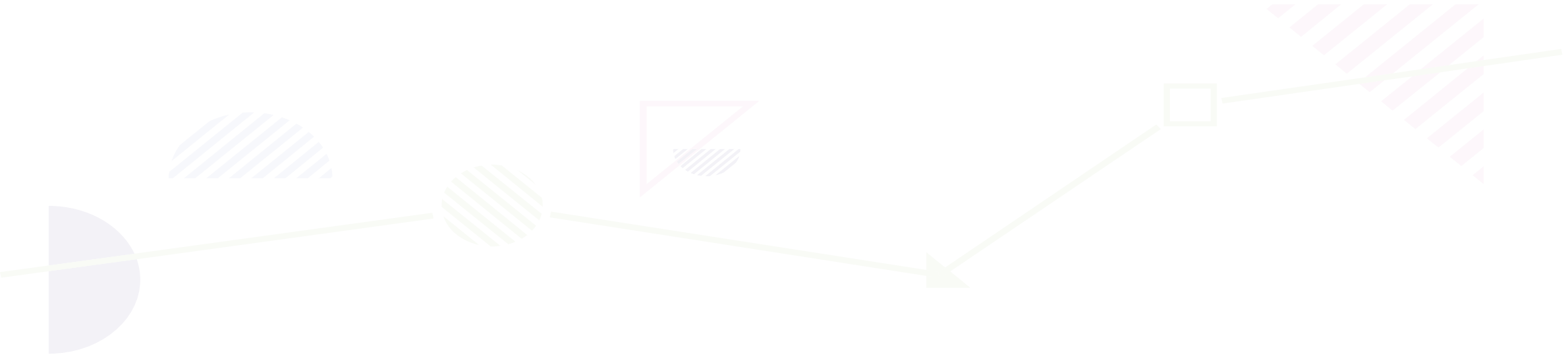
PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 15

# Java 15

- Records
- Text blocks



# Records

## Before

```
class User {  
  
    private final String name;  
    private final String address;  
  
    public User(String name, String address) {  
        this.name = name;  
        this.address = address;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public String getAddress() {  
        return address;  
    }  
  
    @Override  
    public int hashCode() {  
        return Objects.hash(name, address);  
    }  
  
    @Override  
    public boolean equals(Object obj) {  
        if (this == obj) {  
            return true;  
        } else if (!(obj instanceof User)) {  
            return false;  
        } else {  
            User other = (User) obj;  
            return Objects.equals(name, other.name)  
                && Objects.equals(address, other.address);  
        }  
    }  
  
    @Override  
    public String toString() {  
        return "Person [name=" + name + ", address=" + address + "];"  
    }  
}
```

## After

```
record User(String name, String address) {  
}
```

### Custom constructor

```
record User(String name, String address) {  
    public User {  
        Objects.requireNonNull(name);  
    }  
}
```

# Text blocks

## Before








```
String text = "Lorem ipsum dolor sit\n" +  
    "amet, consectetur '\n" +  
    "adipiscing elit. Aenean sit \" amet enim odio. \n" +  
    "Suspendisse quis dui finibus,\n" +  
    "\n" +  
    "\n" +  
    "\n" +  
    "tempor mi et,\n" +  
    "vehicula ex. Suspendisse commodo tris\n";
```

## After

```
String textBlock = """  
    Lorem ipsum dolor sit  
    amet, consectetur '  
    adipiscing elit. Aenean sit " amet enim odio.  
    Suspendisse quis dui finibus,  
  
    tempor mi et,  
    vehicula ex. Suspendisse commodo tris  
    """;
```












PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 15 Exercises

PROCEDURE						
						
Drain the washer fluid tank		Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 16

# Java 16

- Pattern Matching For Instanceof
- Day Period Support
- New Stream.toList method





# Pattern Matching For instanceof

```
// old
if (shape instanceof Circle) {
    Circle circle = (Circle) shape;
    return circle.getDisplayName();
}
```

```
//new
if (shape instanceof Circle circle) {
    return circle.getDisplayName();
}
```



# Day Period Support

## New DateTimeFormatter symbol

### Alternative to am/pm

```
LocalTime date = LocalTime.parse("18:25:08.690791");  
DateTimeFormatter formatter = DateTimeFormatter.ofPattern("h B");  
  
date.format(formatter)  
// 6 in the afternoon
```





# New Stream.toList method








## New toList() method

```
//old  
List<Integer> list = Stream.of("1", "2", "3", "4")  
    .map(Integer::parseInt)  
    .collect(Collectors.toList());
```

```
//new  
List<Integer> list = Stream.of("1", "2", "3", "4")  
    .map(Integer::parseInt)  
    .toList();
```












PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 16 Exercises



PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 17



# Java 17

- Always Strict Floating-Point
- Sealed Classes



# Always Strict Floating-Point

Floating-point operations always in strict mode  
Result is independent of platform

```
public strictfp class SomeMathClass {}  
public strictfp interface Calulcator {}  
public strictfp void sum() {}
```



# Sealed Classes

```
sealed class Bike permits Motorbike, Bicycle {  
}
```

```
non-sealed class Motorbike extends Bike {  
}
```

```
final class Bicycle extends Bike {  
}
```

**non-sealed**

open inheritance

**sealed**








restrict inheritance

**final**

block inheritance





PROCEDURE						
						
Drain the washer fluid tank	Take 15mm Wrench	Unmount the washer fluid tank	Remove the washer fluid tank	Inspect the washer fluid tank	Replace the washer fluid tank	Refill the system
PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER	PART NUMBER
0000001245	0000001255	0000001273	0000001273	0000001273	0000001274	0000001274



# Java 17 Exercises



**TRANSITION**  
TECHNOLOGIES

# Thanks for attention!



Michał Krysiuk



krysiukm@tt.com.pl



mivchal333