

JAKIE ZJAWY NA NAS CZYHAJĄ?

Czyli co nowego w Javie

Marcin Baranowski
Snowflake

Programiści
to masochiści



Błędy kompilacji



Nieprzechodzące testy



Błędy czasu wykonania



Klient

Wow, a different
error message...
Finally some progress!

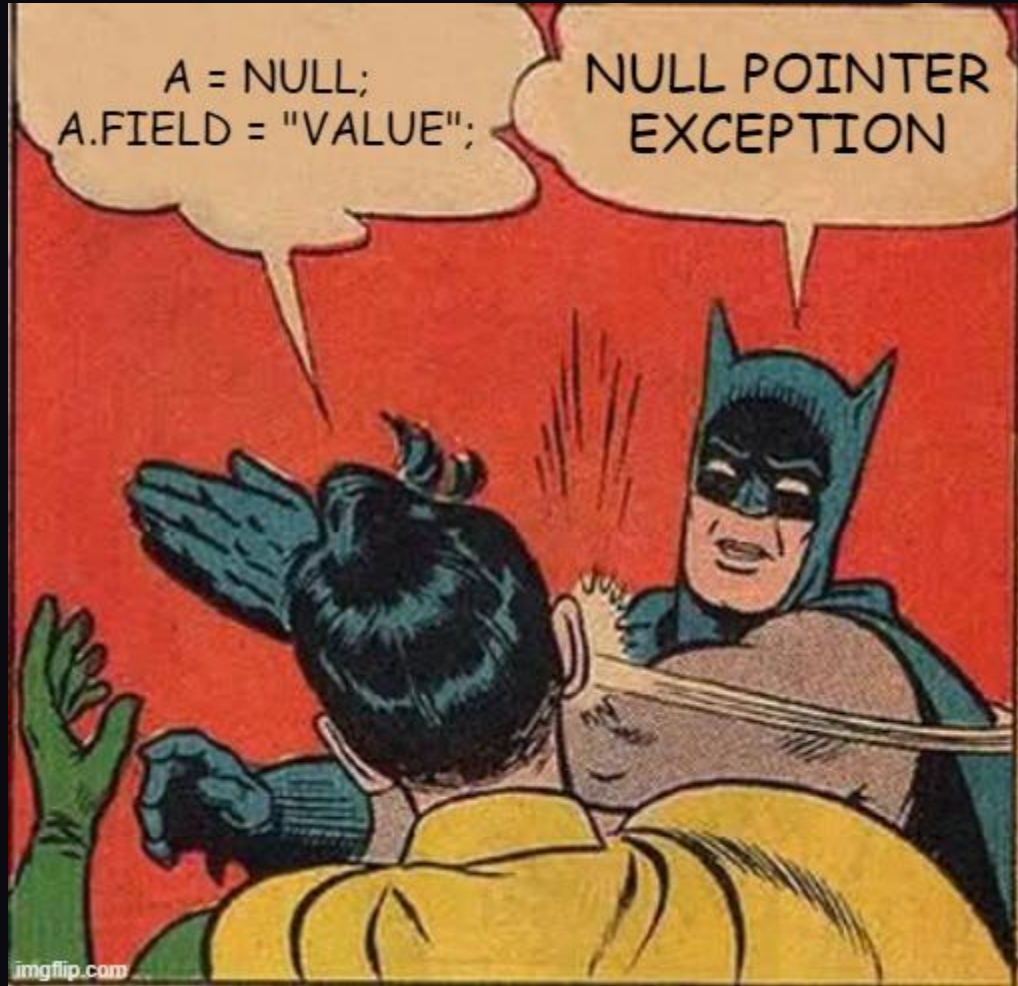


13,6 sekundy

Średnio co taki czas pojawia się nowe pytanie na stackoverflow

Feature'y

Więcej niż cztery





NullPointerException

Switch


```
public class Old {  
  
    static void ratePresentation(Grade grade) {  
        int score;  
        switch (grade) {  
            case BAD:  
            case WORST_EVER:  
                score = 1;  
                break;  
            case NEUTRAL:  
                score = 3;  
                break;  
            case GREAT:  
            case NICE: {  
                System.out.println("Impressive!");  
                score = 5;  
                break;  
            }  
            default:  
                throw new IllegalStateException("Unexpected value: " + grade);  
        }  
        System.out.println("Presentation score: " + score);  
    }  
  
    public static void main(String[] args) {  
        ratePresentation(Grade.GREAT);  
    }  
}
```

```
enum Grade {WORST_EVER, BAD, NEUTRAL, GREAT, NICE}
```

Switch expressions

Nowe podejście



```
public class New {  
  
    static void ratePresentation(Grade grade) {  
        int score = switch (grade) {  
            case BAD, WORST_EVER -> 1;  
            case NEUTRAL -> 3;  
            case GREAT, NICE -> {  
                System.out.println("Impressive!");  
                yield 5;  
            }  
        };  
        System.out.println("Presentation score: " + score);  
    }  
  
    public static void main(String[] args) {  
        ratePresentation(Grade.GREAT);  
    }  
}
```

```
enum Grade {WORST_EVER, BAD, NEUTRAL, GREAT, NICE}
```



```
public class Old {  
  
    static void ratePresentation(Grade grade) {  
        int score;  
        switch (grade) {  
            case BAD:  
            case WORST_EVER:  
                score = 1;  
                break;  
            case NEUTRAL:  
                score = 3;  
                break;  
            case GREAT:  
            case NICE: {  
                System.out.println("Impressive!");  
                score = 5;  
                break;  
            }  
            default:  
                throw new IllegalStateException("Unexpected value: " + grade);  
        }  
        System.out.println("Presentation score: " + score);  
    }  
  
    public static void main(String[] args) {  
        ratePresentation(Grade.GREAT);  
    }  
}
```

18
linii

```
public class New {  
  
    static void ratePresentation(Grade grade) {  
        int score = switch (grade) {  
            case BAD, WORST_EVER -> 1;  
            case NEUTRAL -> 3;  
            case GREAT, NICE -> {  
                System.out.println("Impressive!");  
                yield 5;  
            }  
        };  
        System.out.println("Presentation score: " + score);  
    }  
  
    public static void main(String[] args) {  
        ratePresentation(Grade.GREAT);  
    }  
}
```

8 linii

Usunięcie Nashorn Javascript



Co Adam Mickiewicz sądził o stringach?


```
String str =
    "Litwo! Ojczyzno moja! Ty jeste\u015B jak zdrowie,\n" +
    "Ile ci\u0119 trzeba ceni\u0107, ten tylko si\u0119 dowie,\n" +
    "Kto ci\u0119 straci\u0142. Dzi\u015B pi\u0119kno\u015B\u0107 tw\u0105 w ca\u0142ej ozdobie\n" +
    "Widz\u0119 i opisuj\u0119, bo t\u0119skni\u0119 po tobie\n" +
    "Panno \u015Bwi\u0119ta, co Jasnej bronisz Cz\u0119stochowy\n" +
    "I w Ostrej \u015Bwiecisz Bramie! Ty, co gr\u00F3d zamkowy\n" +
    "Nowogr\u00F3dzki ochraniasz z jego wiernym ludem!\n" +
    "Jak mnie dziecko do zdrowia powr\u00F3ci\u0142a\u015B cudem,\n" +
    "(Gdy od p\u0142acz\u0105cej matki pod Twój\u0105 opiek\u0119\n" +
    "Ofiarowany, martw\u0105 podnios\u0142em powiek\u0119\n" +
    "I zaraz mog\u0142em pieszo do Twych \u015Bwi\u0105ty\u0144 progu\n" +
    "I\u015B\u0107 za wr\u00F3cone \u017Cycie podzi\u0119kowa\u0107 Bogu),\n" +
    "Tak nas powr\u00F3ci\u015B cudem na Ojczyzn\u0142ono.\n" +
    "Tymczasem przeno\u015B moj\u0105 dusz\u0119 ut\u0119sknion\u0105\n" +
    "Do tych pag\u00F3rk\u00F3w le\u015Bnych, do tych \u0142\u0105k zielonych,\n" +
    "Szeroko nad b\u0142\u0119kitnym Niemnem rozci\u0105gnionych;\n" +
    "Do tych p\u00F3l malowanych zbo\u017Cem rozmaitem,\n" +
    "Wyz\u0142aczanych pszenic\u0105, posrebrzanych \u017Cytem;\n" +
    "Gdzie bursztynowy \u015Bwierzop, gryka jak \u015Bnieg bia\u0142a,\n" +
    "Gdzie panie\u0144skim rumie\u0144cem dzi\u0119cielina pa\u0142a,\n" +
    "A wszystko przepasane jakby wst\u0119g\u0105, miedz\u0105\n" +
    "Zielon\u0105, na niej z rzadka ciche grusze siedz\u0105.";
```

Textblocks

instanceof


```
/**
 * simulates blackbox - we don't know what exactly will come out
 *
 * @return either String or BigDecimal
 */
private static Object blackbox() {
```

```
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof String) {
        String result = (String) obj;
        System.out.println("input is String");
        System.out.println(result.toUpperCase());
    } else if (obj instanceof BigDecimal) {
        BigDecimal result = (BigDecimal) obj;
        if (result.equals(BigDecimal.ONE)) {
            System.out.println("input is BigDecimal");
            System.out.println(result.add(BigDecimal.ONE));
        }
    }

    System.out.println("Fin");
}
```

**instanceof
bez rzutowania?**

Pattern matching



```
public static void main(String[] args) {  
    Object obj = blackbox();  
  
    if (obj instanceof String result) {  
        System.out.println("input is String");  
        System.out.println(result.toUpperCase());  
    }  
  
    System.out.println("Fin");  
}
```

```
public static void main(String[] args) {  
    Object obj = blackbox();  
  
    if (obj instanceof String result) {  
        System.out.println("input is String");  
        System.out.println(result.toUpperCase());  
    }  
  
    System.out.println("Fin");  
}
```


Data class

Zbliżamy się do rekordu

Record



```
public record New(boolean isAwesome, String title) {  
  
    public static void main(String[] args) {  
        New instance = new New(true, "Awesome record");  
        System.out.println(instance.title());  
    }  
}
```

**Rekord w liczbie urodzonych
dzieci?**

Fiodorowa Wasiljewa

69

Jak to się ma do Javy?

Dziedziczenie

final

Klasy sealed

Antykoncepcja dla Javy

Nie chcemy tego w naszym kodzie ->

Nie chcemy tego w naszym kodzie ->

```
class Animal {  
}  
  
class Cat extends Animal {  
}  
  
class TrojanHorse extends Animal {  
}  
  
public class Old {  
    public static void main(String[] args) {  
        var cat = new Cat();  
        System.out.println(cat);  
  
        var trojanHorse = new TrojanHorse();  
        System.out.println(trojanHorse);  
    }  
}
```


sealed

non-sealed

permits

Jakiej muzyki słuchają klasy?

final class



sealed class



non-sealed class





class

```
class Rodzic { }
```

```
class NieplanownyPotomek extends Rodzic { }
```

Finalizers

Deprecated



```
record Point(int x, int y) {}
```

```
record Rectangle(Point topLeft, Point bottomRight) {}
```



```
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof Rectangle rect) {
        var topLeft = rect.topLeft();
        var bottomRight = rect.bottomRight();
        var left = topLeft.x();
        var right = bottomRight.x();
        var width = right - left;
        var top = topLeft.y();
        var bottom = bottomRight.y();
        var height = bottom - top;
        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
    }

    System.out.println("Fin");
}
```

10 linii

```
record Point(int x, int y) {}
```

```
record Rectangle(Point topLeft, Point bottomRight) {}
```

Co powiecie na 4 linie?

Record patterns



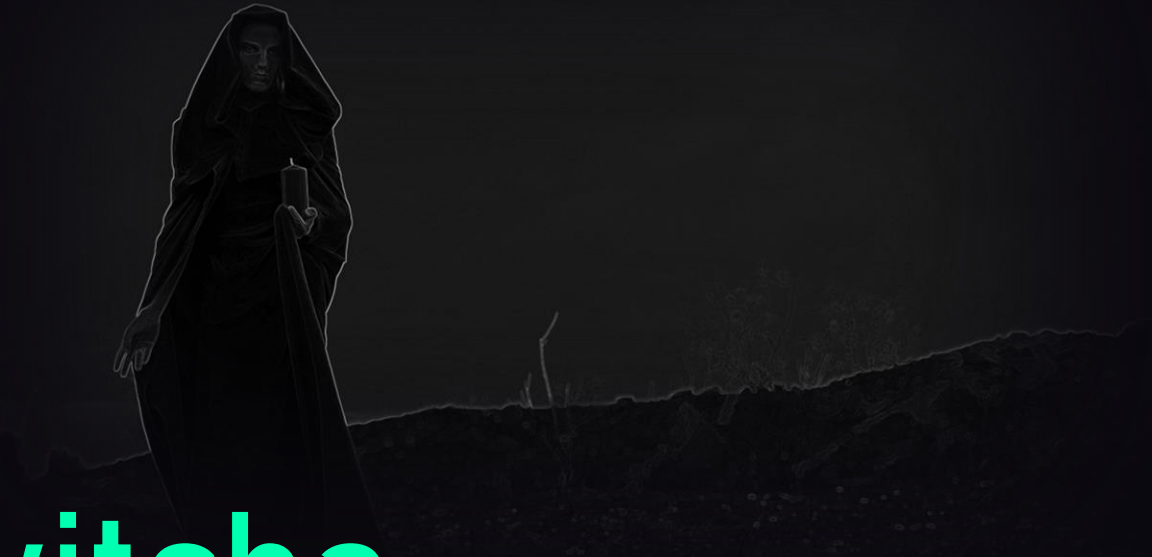
```
record Point(int x, int y) {}
```

```
public static void main(String[] args) {  
    Object obj = blackbox();
```

```
record Rectangle(Point topLeft, Point bottomRight) {}
```

```
    if (obj instanceof Rectangle(Point(int leftX, int topY), Point(int rightX, int bottomY))) {  
        var width = rightX - leftX;  
        var height = bottomY - topY;  
        System.out.println("Width: " + width);  
        System.out.println("Height: " + height);  
    }
```

```
    System.out.println("Fin");  
}
```



Ulepszenia switcha



Ulepszenia switcha

null

```
public class New {  
  
    static void ratePresentation(Grade grade) {  
        int score = switch (grade) {  
            case null, BAD, WORST_EVER -> 1;  
            case NEUTRAL -> 3;  
            case GREAT, NICE -> {  
                System.out.println("Wow!");  
                yield 5;  
            }  
        };  
        System.out.println("Presentation score: " + score);  
    }  
  
    public static void main(String[] args) {  
        ratePresentation(null);  
    }  
}
```



Ulepszenia switcha

pattern matching


```
public static void main(String[] args) {  
    Object obj = blackbox();  
  
    switch(obj) {  
        case BigDecimal bigDecimal ->  
            System.out.println("Big decimal: " + bigDecimal.add(BigDecimal.ONE));  
        case String str ->  
            System.out.println("String: " + str.toUpperCase());  
        case null, default ->  
            System.out.println("World is unexpected!");  
    }  
}
```



Ulepszenia switcha

when

```
static void testTriangle(Shape s) {  
    switch (s) {  
        case null:  
            break;  
        case Triangle t:  
            if (t.calculateArea() > 100) {  
                System.out.println("Large triangle");  
                break;  
            }  
            // no break  
        default:  
            System.out.println("A shape, possibly a small triangle");  
    }  
}
```

```
static void testTriangle(Shape s) {  
    switch (s) {  
        case null -> {}  
        case Triangle t when t.calculateArea() > 100 ->  
            System.out.println("Large triangle");  
        default ->  
            System.out.println("A shape, possibly a small triangle");  
    }  
}
```



Ulepszenia switcha

kolejność case'ów

```
static void error(Object o) {  
    switch (o) {  
        case CharSequence cs ->  
            System.out.println("A sequence of length " + cs.length());  
        case String s ->                                     // won't compile in JDK 17+  
            System.out.println("A string: " + s);  
        default -> {}  
    }  
}
```

```
static void error(Object o) {  
    switch (o) {  
        case String s ->  
            System.out.println("A string: " + s);  
        case CharSequence cs ->  
            System.out.println("A sequence of length " + cs.length());  
        default -> {}  
    }  
}
```



Ulepszenia switcha

Sealed classes


```
static void checkAnimal(Animal animal) {  
    switch (animal) {  
        case Cat ignored -> System.out.println("just cat");  
        case Dog ignored -> System.out.println("just dog");  
        case Fish ignored -> System.out.println("just fish");  
    }  
}
```

```
sealed interface Animal permits Cat, Dog, Fish { }
```

```
final class Cat implements Animal { }
```

```
final class Dog implements Animal { }
```

```
final class Fish implements Animal { }
```

Default jest niepotrzebny ze względu na zamkniętą hierarchię (**sealed interface**)

Tradycyjny wątek

wątków w Javie

=

wątków systemowych

<=

MAX # wątków systemowych

Virtual threads



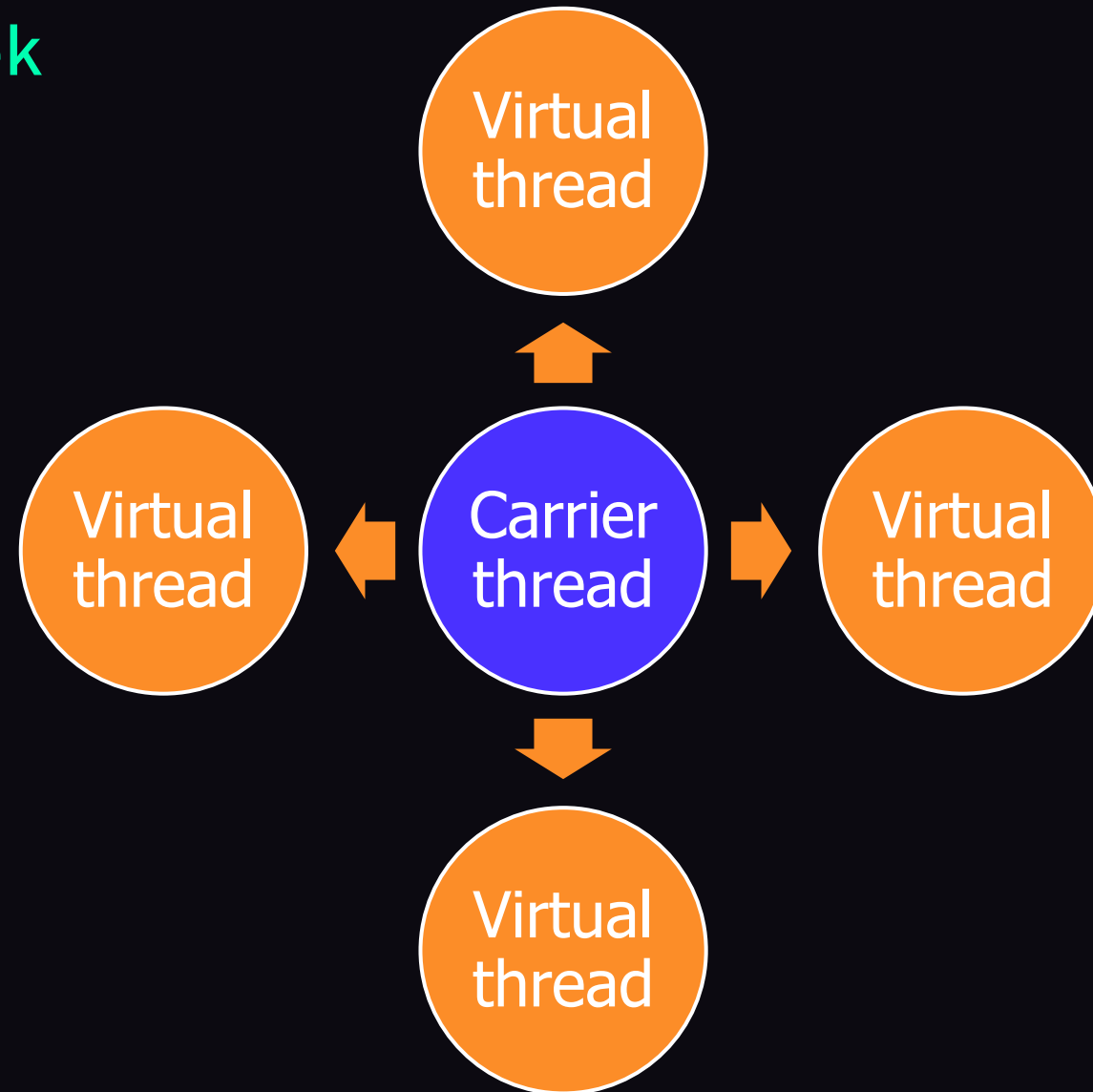
Wirtualny wątek

wątków w Javie



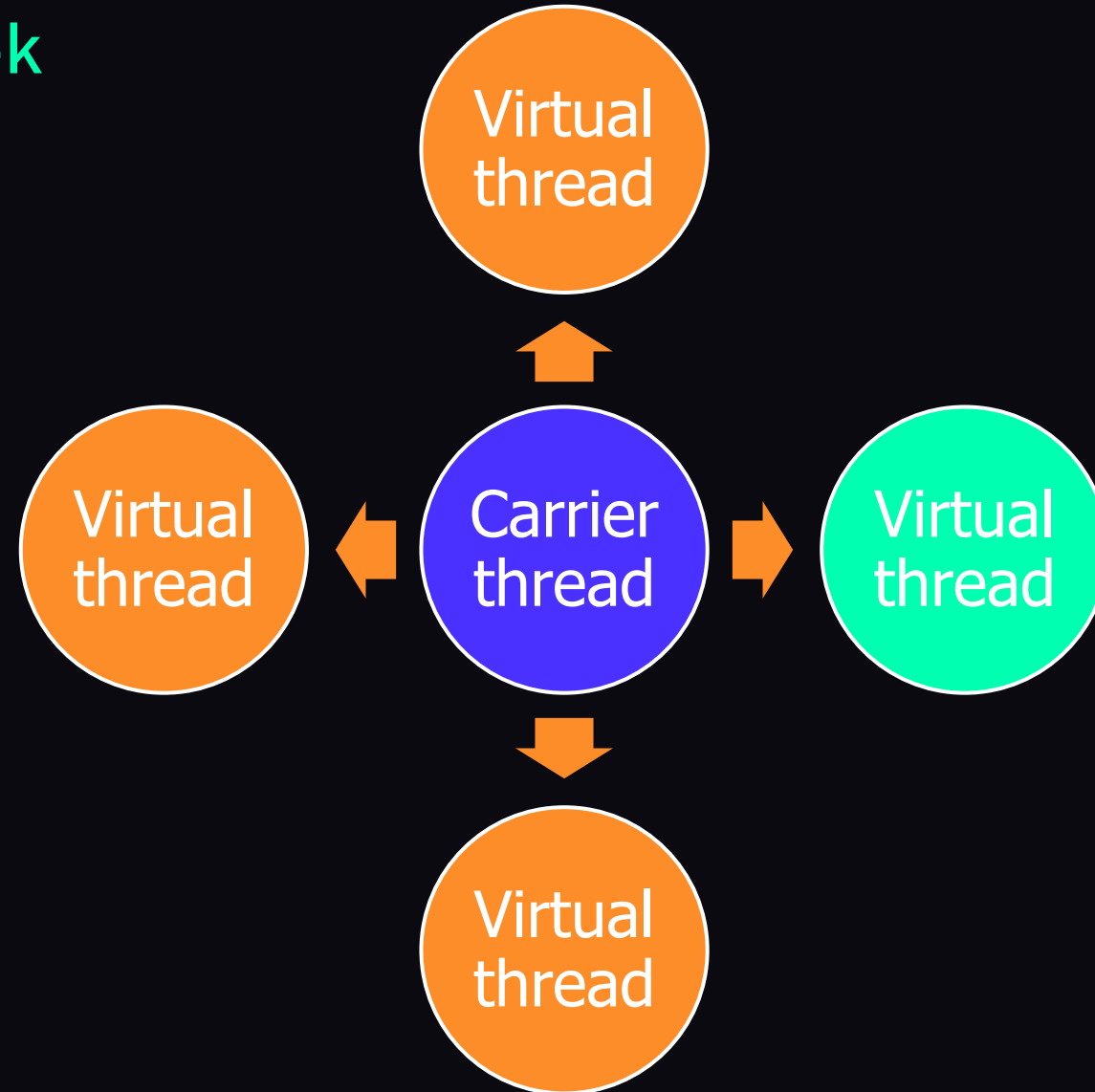
MAX # wątków systemowych

Wirtualny wątek



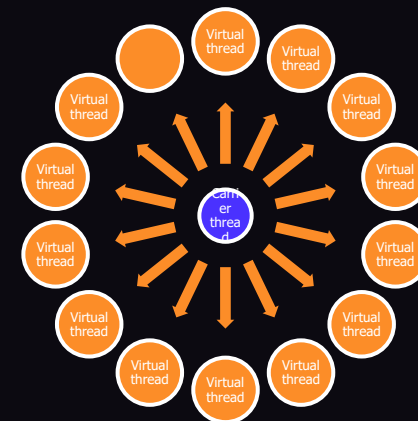
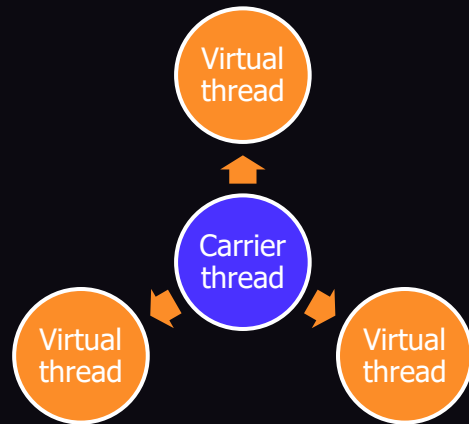
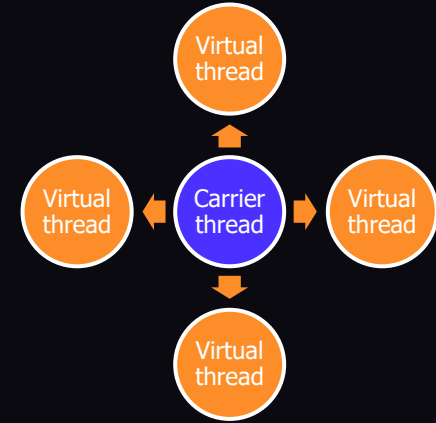
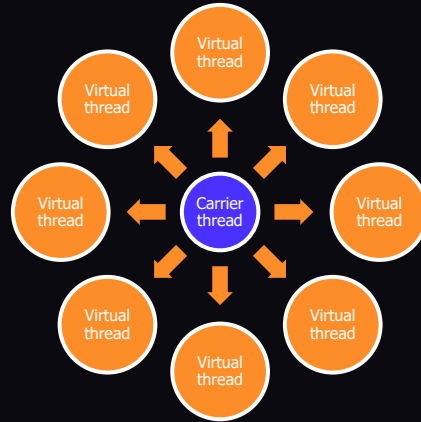
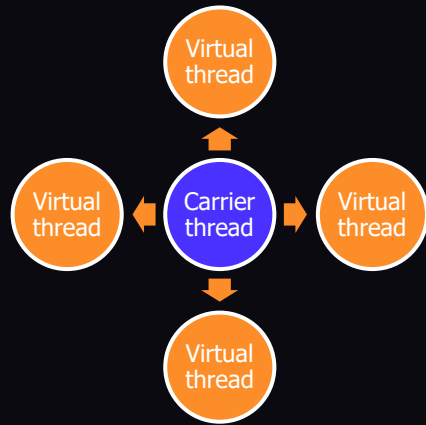
Carrier thread to "klasyczny" wątek, który zarządza wirtualnymi wątkami

Wirtualny wątek



W danej chwili Carrier thread może obsługiwać tylko jeden wątek wirtualny

Wirtualny wątek



W ramach jednej aplikacji może istnieć wiele wątków-zarządców z wieloma wirtualnymi wątkami

Wirtualny wątek

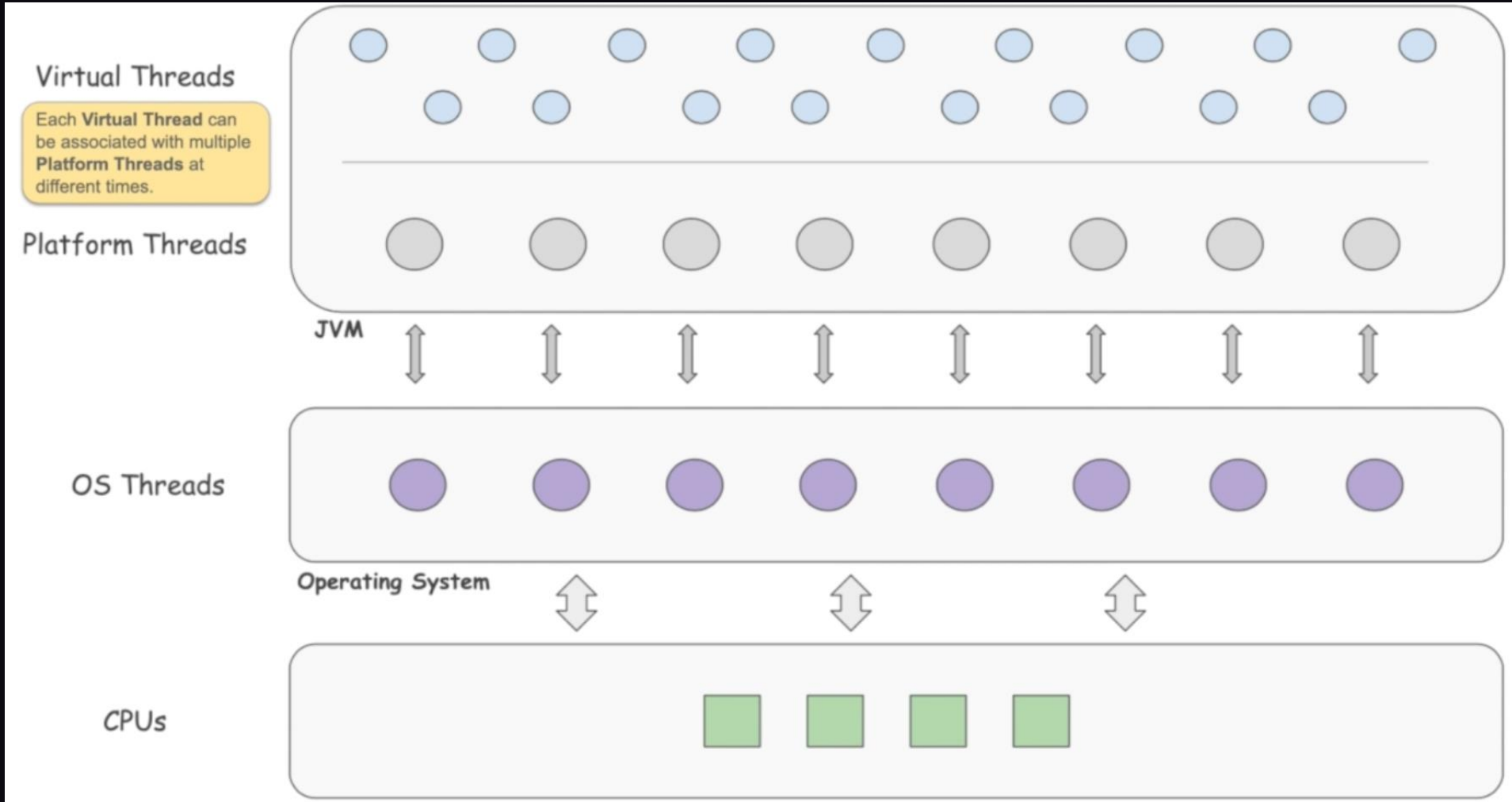
Carrier thread przelacza się na inny wątek wirtualny, gdy jego aktualny wątek jest zablokowany

Wirtualny wątek

Jakie operacje blokują wątek?

- Komunikacja z bazą danych
- Połączenie sieciowe
- Wszelkie inne operacje we-wy
- Sleep

Wirtualny wątek



Klasyczny (platformowy) wątek

```
public static void main(String[] args) {  
    IntStream.range(0, THREADS_NO)  
        .forEach(i -> new Thread(new Task()).start());  
}
```

Wirtualny wątek

```
public static void main(String[] args) {  
    IntStream.range(0, THREADS_NO)  
        .forEach(i -> Thread.ofVirtual().start(new Task()));  
}
```

Wirtualny wątek

```
public static void main(String[] args) {  
    IntStream.range(0, THREADS_NO)  
        .forEach(i -> Thread.ofVirtual().start(new Task()));  
}
```

Wirtualny wątek



Nie należy używać z pulą wątków



Można zapchać „pulę” używając **synchronized**



Brak priorytetów



Nowinka (nadal preview)

To wszystkie
zjawy na dziś

Zmniejszanie długu technologicznego
jest jak ostrzenie noża:

Oстрыm nożem szybciej się kroi
i
Ciężiej się skaleczyć



vianto.pl

Tak samo z kodem:

Zadbany kod łatwiej się czyta
i
Trudniej popełnić w nim błąd

Pracuj nad długiem
technologicznym

**Niech biznes wie,
że to ważne**

Testuj zmiany



**“What does this
do?”**



“Shit”



“shit shit shit”



**“If I leave, no
one will notice”**

mbaranowski.pl