JAKIE ZJAWY NA NAS CZYHAJĄ?

Czyli co nowego w Javie

Marcin Baranowski
Snowflake

Programiści to masochiści

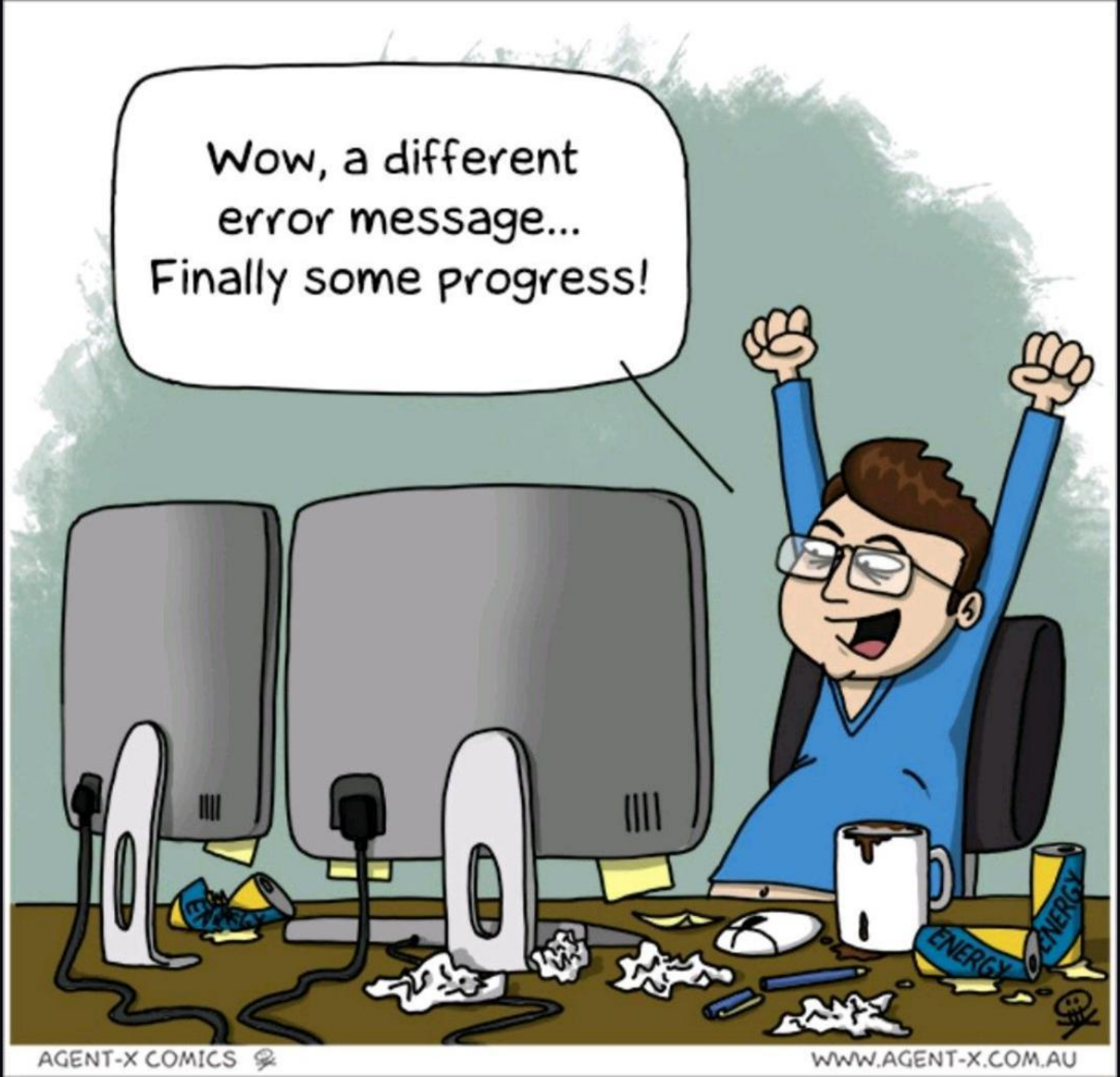
Błędy kompilacji

Nieprzechodzące testy

Błędy czasu wykonania

Klient

Icons by https://pl.freepik.com/rawpixel-com


Wow, a different error message... Finally some progress!
ENERGY
ENERGY
ENERGY
AGENT-X COMICS
WWW.AGENT-X.COM.AU

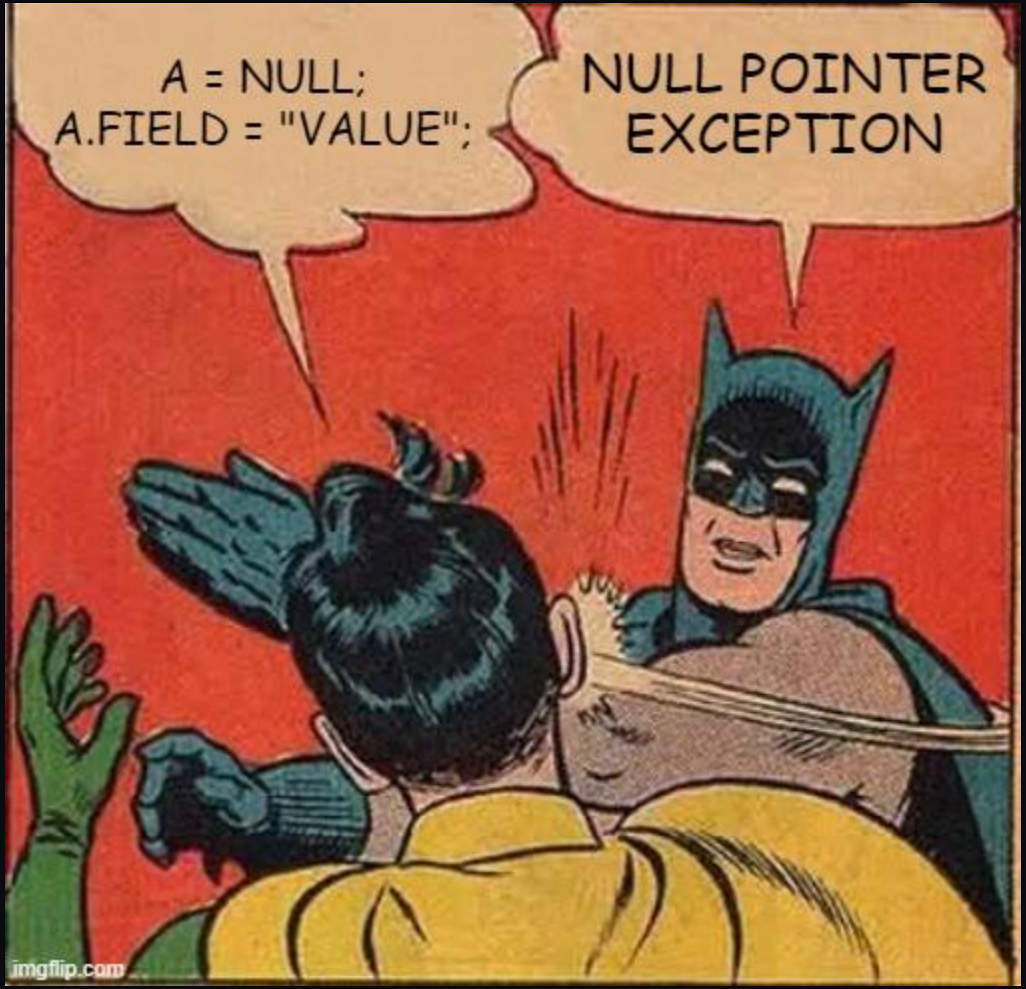
13,6 sekundy

Feature'y

Więcej niż cztery

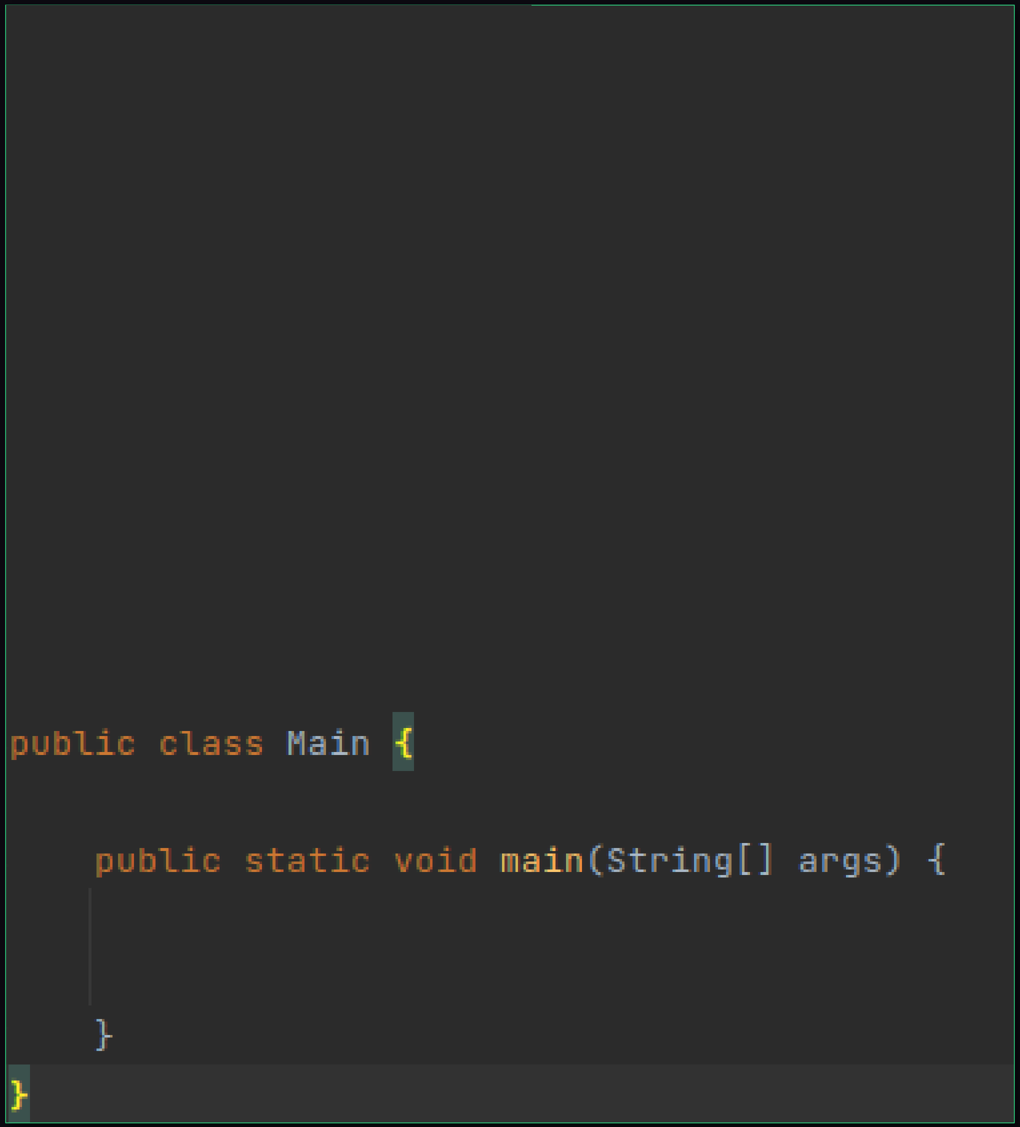
Image: freepik.com

A = NULL;
A.FIELD = "VALUE";
imgflip.com

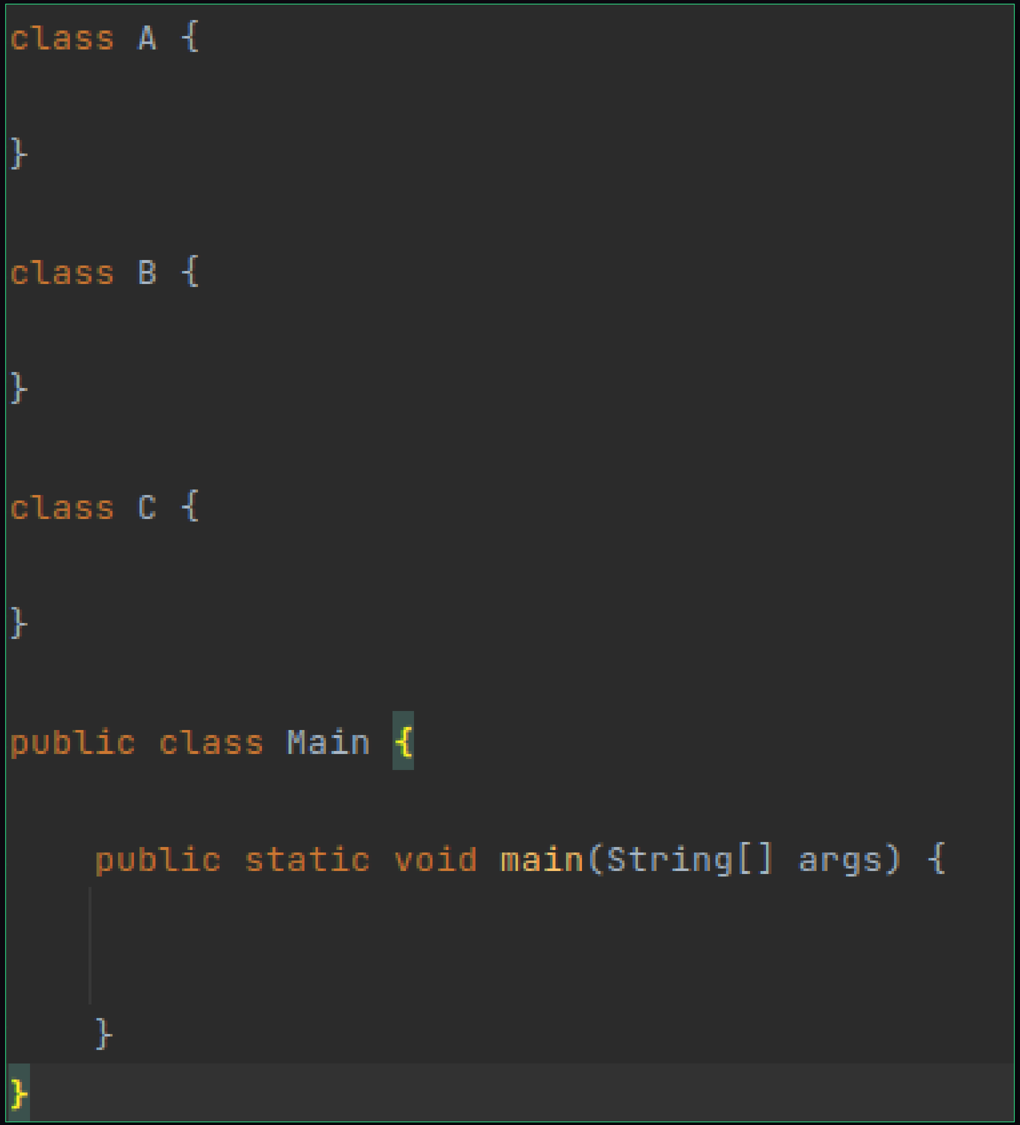

A = NULL;
A.FIELD = "VALUE";
NULL POINTER EXCEPTION
imgflip.com

NullPointerException

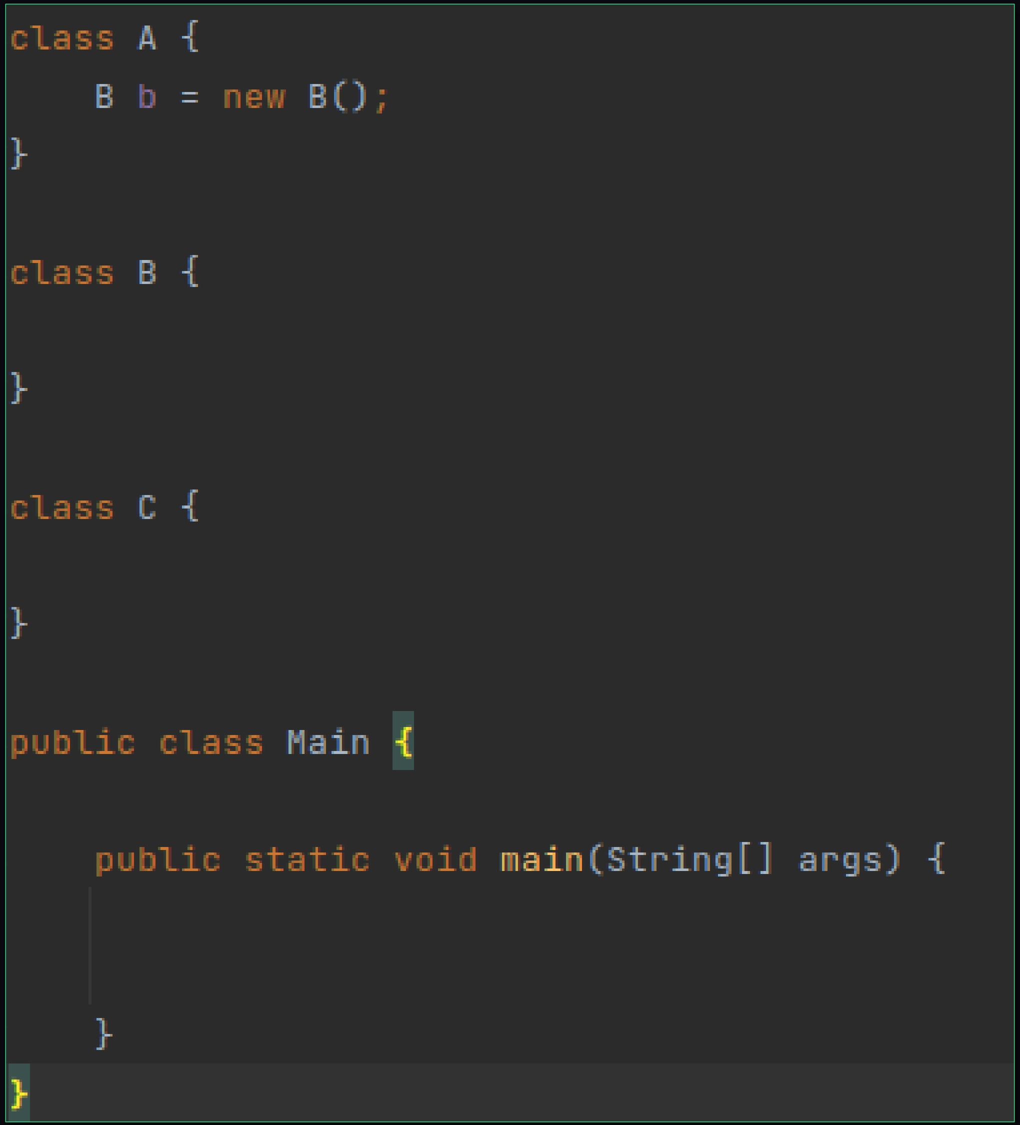
Image: freepik.com



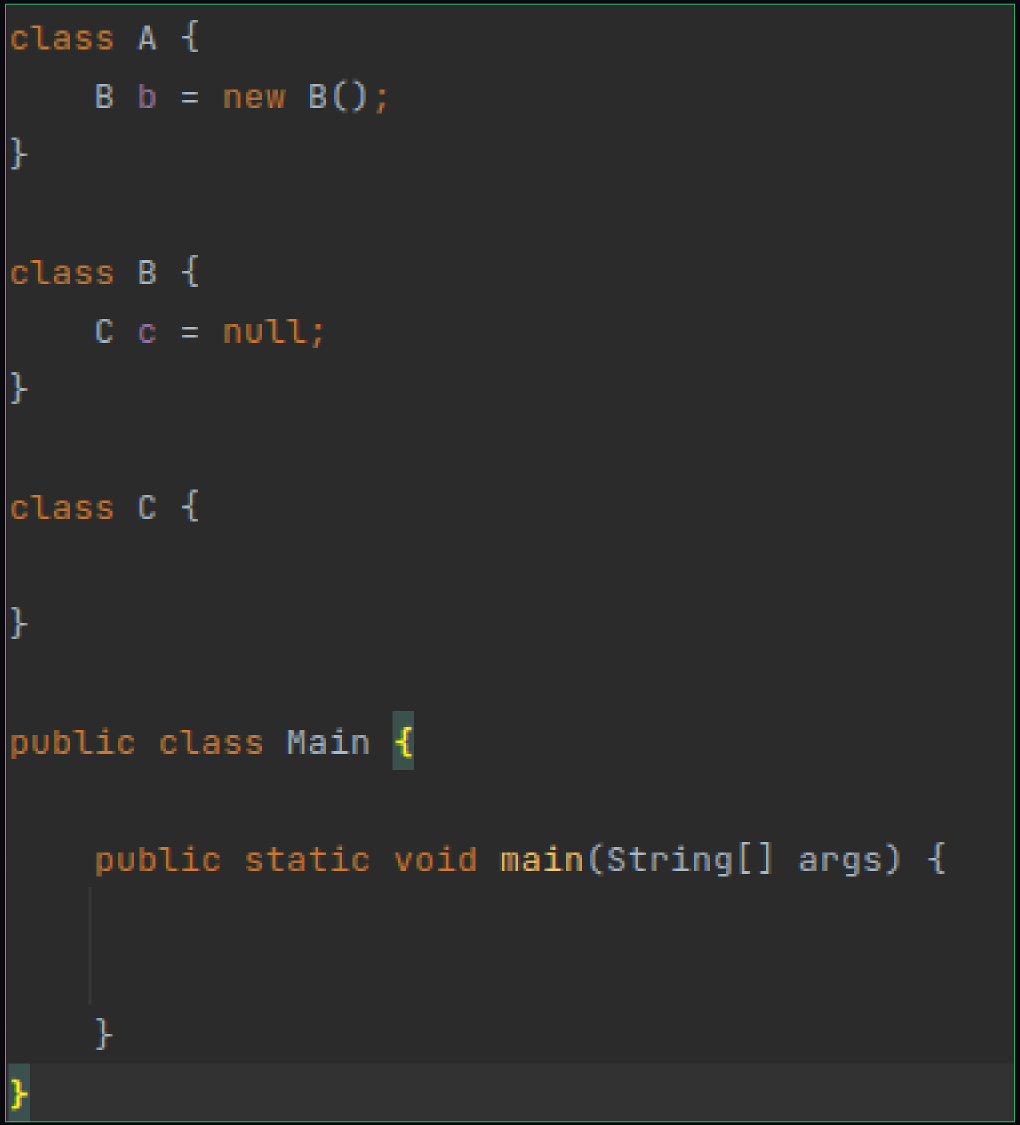
```
public class Main {

    public static void main(String[] args) {


    }
}
```



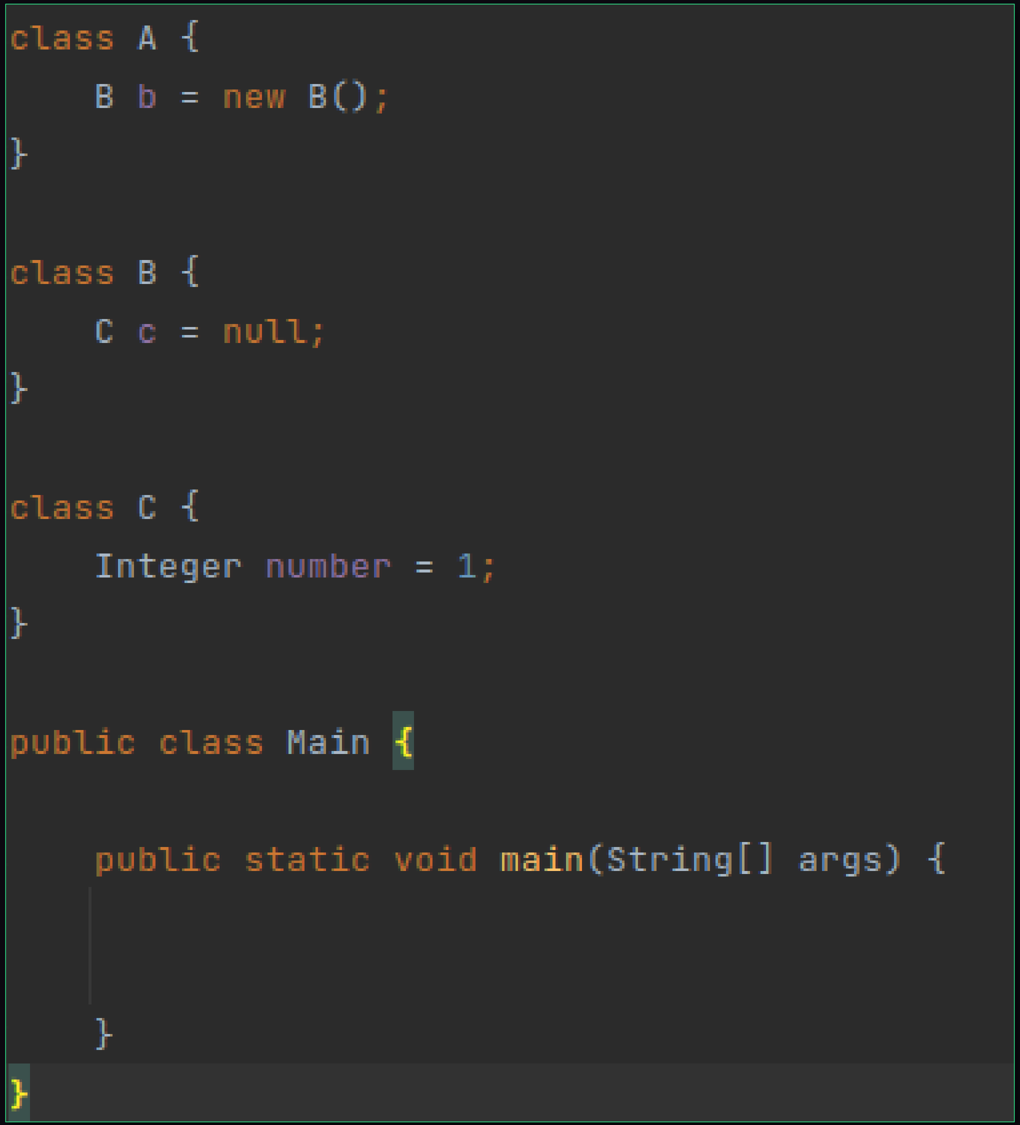
```java
class A {

}

class B {

}

class C {

}

public class Main {

    public static void main(String[] args) {

    }
}
```

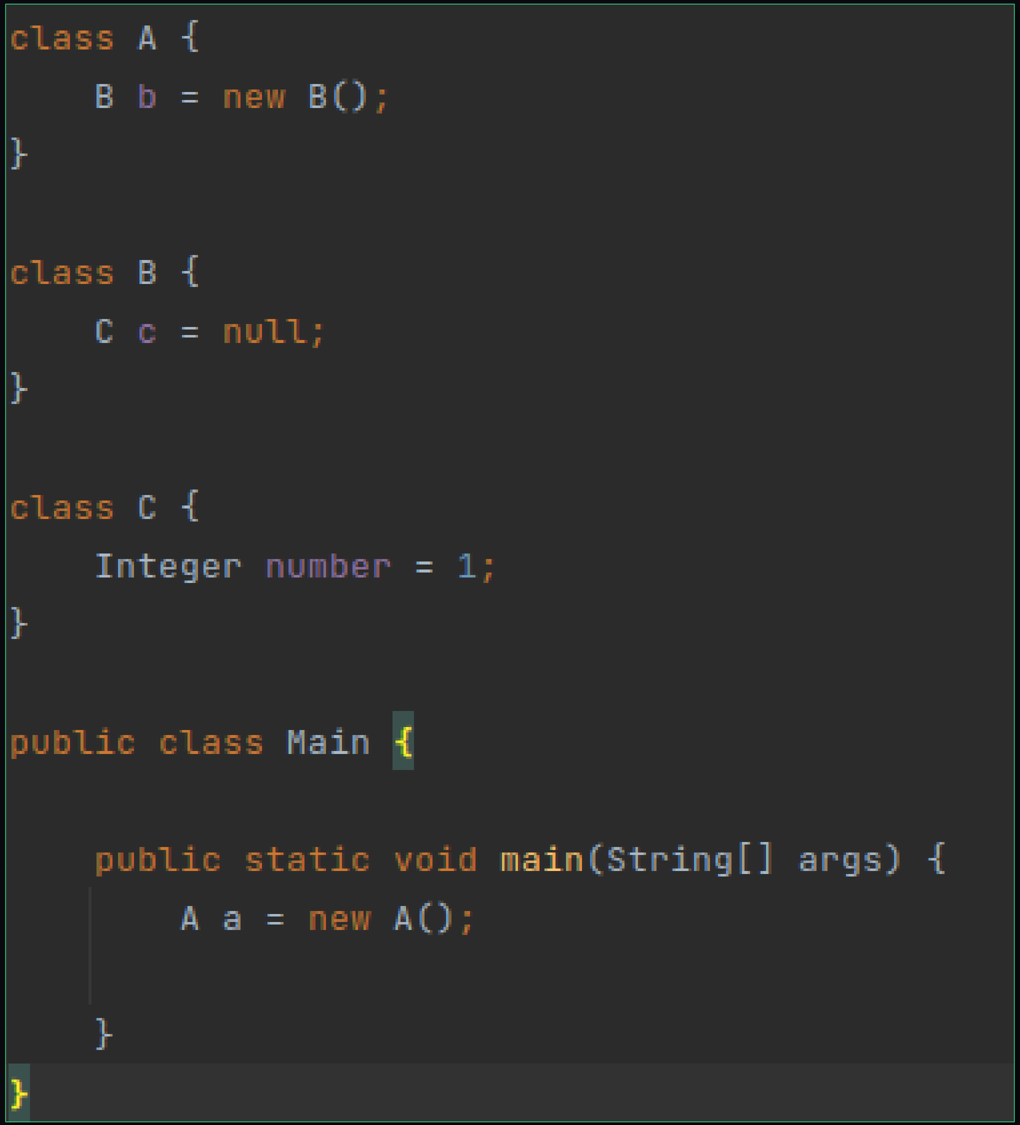


```java
class A {
    B b = new B();
}

class B {

}

class C {

}

public class Main {

    public static void main(String[] args) {


    }
}
```

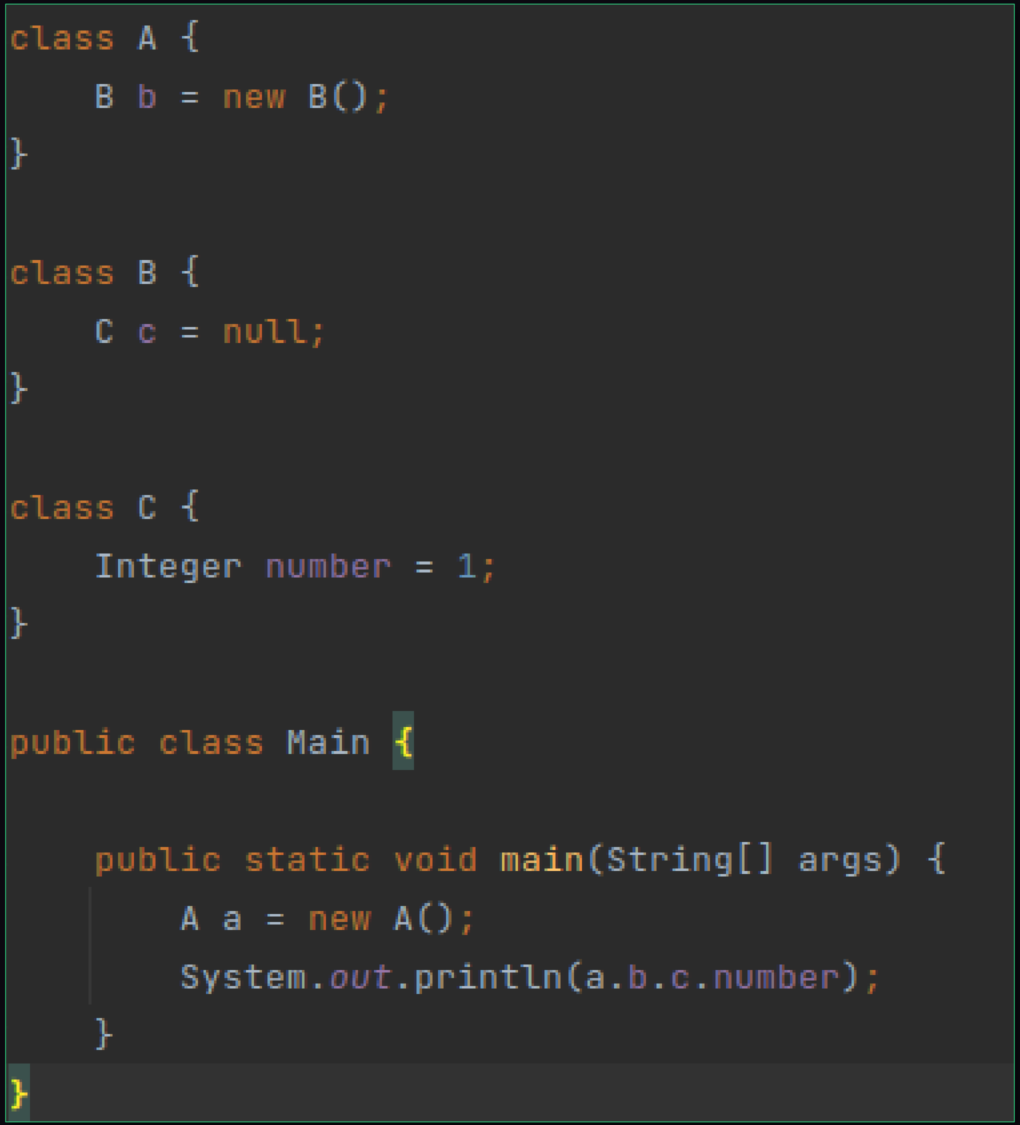


```java
class A {
    B b = new B();
}

class B {
    C c = null;
}

class C {

}

public class Main {

    public static void main(String[] args) {


    }
}
```

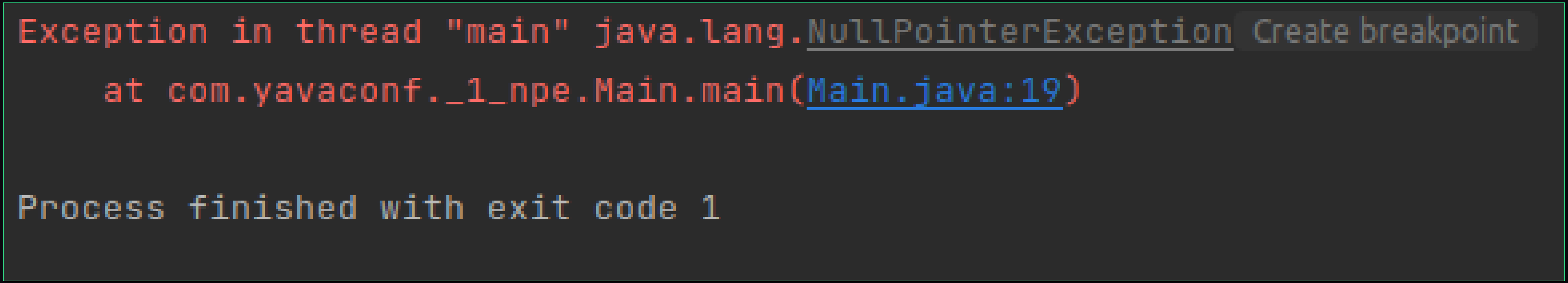

```
class A {
    B b = new B();
}

class B {
    C c = null;
}

class C {
    Integer number = 1;
}

public class Main {

    public static void main(String[] args) {


    }
}
```



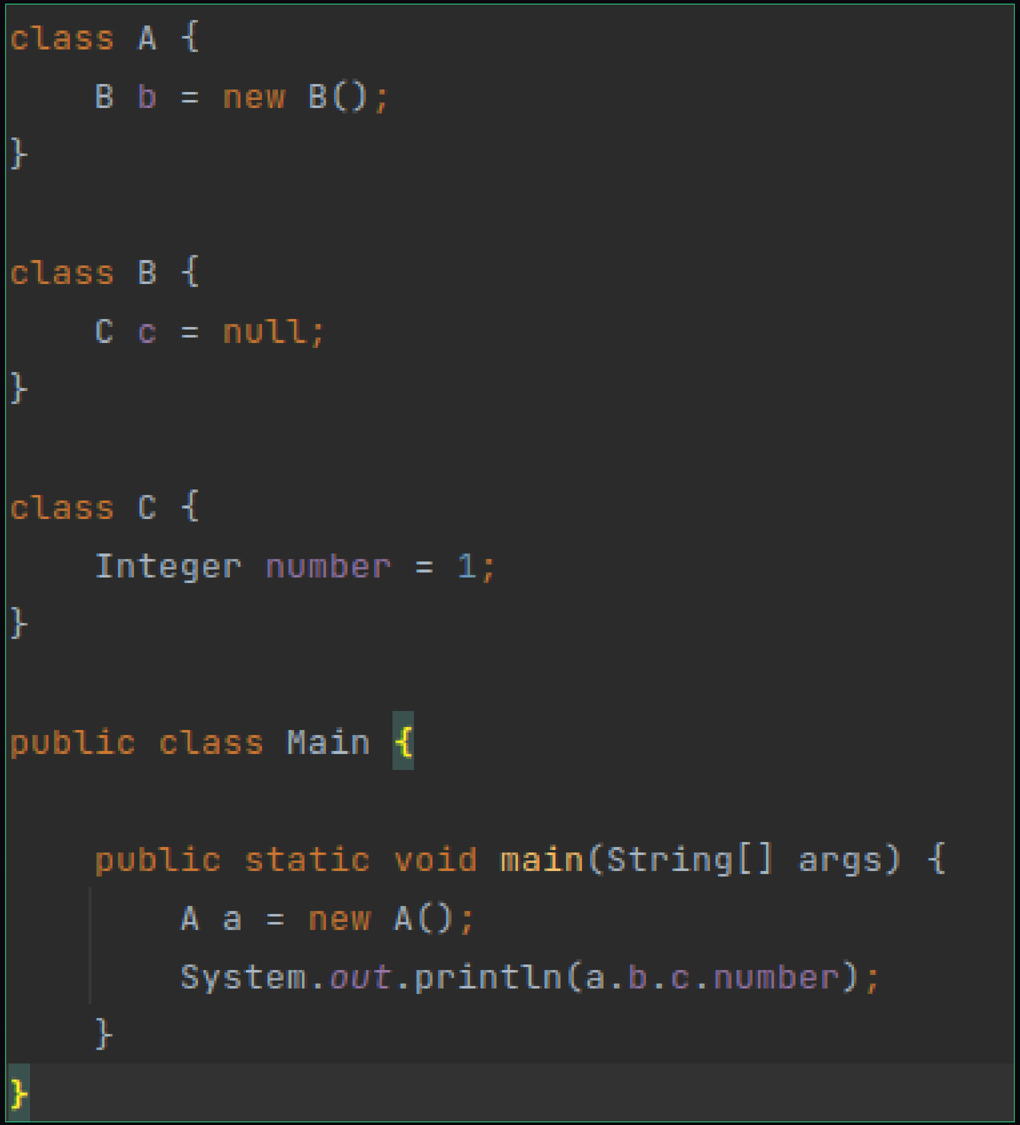
```java
class A {
    B b = new B();
}

class B {
    C c = null;
}

class C {
    Integer number = 1;
}

public class Main {

    public static void main(String[] args) {
        A a = new A();

    }
}
```



```java
class A {
    B b = new B();
}

class B {
    C c = null;
}

class C {
    Integer number = 1;
}

public class Main {

    public static void main(String[] args) {
        A a = new A();
        System.out.println(a.b.c.number);
    }
}
```

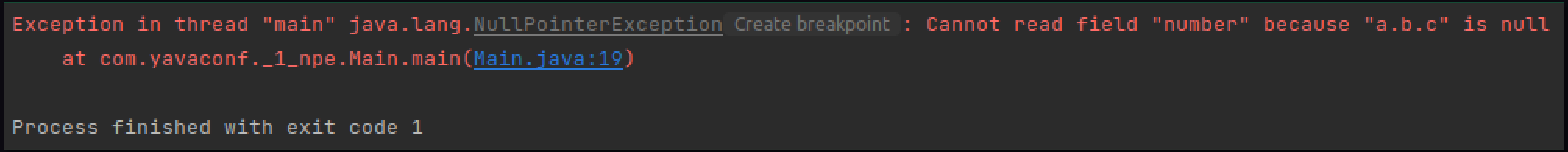


```
Exception in thread "main" java.lang.NullPointerException
    at com.yavaconf._1_npe.Main.main(Main.java:19)

Process finished with exit code 1
```

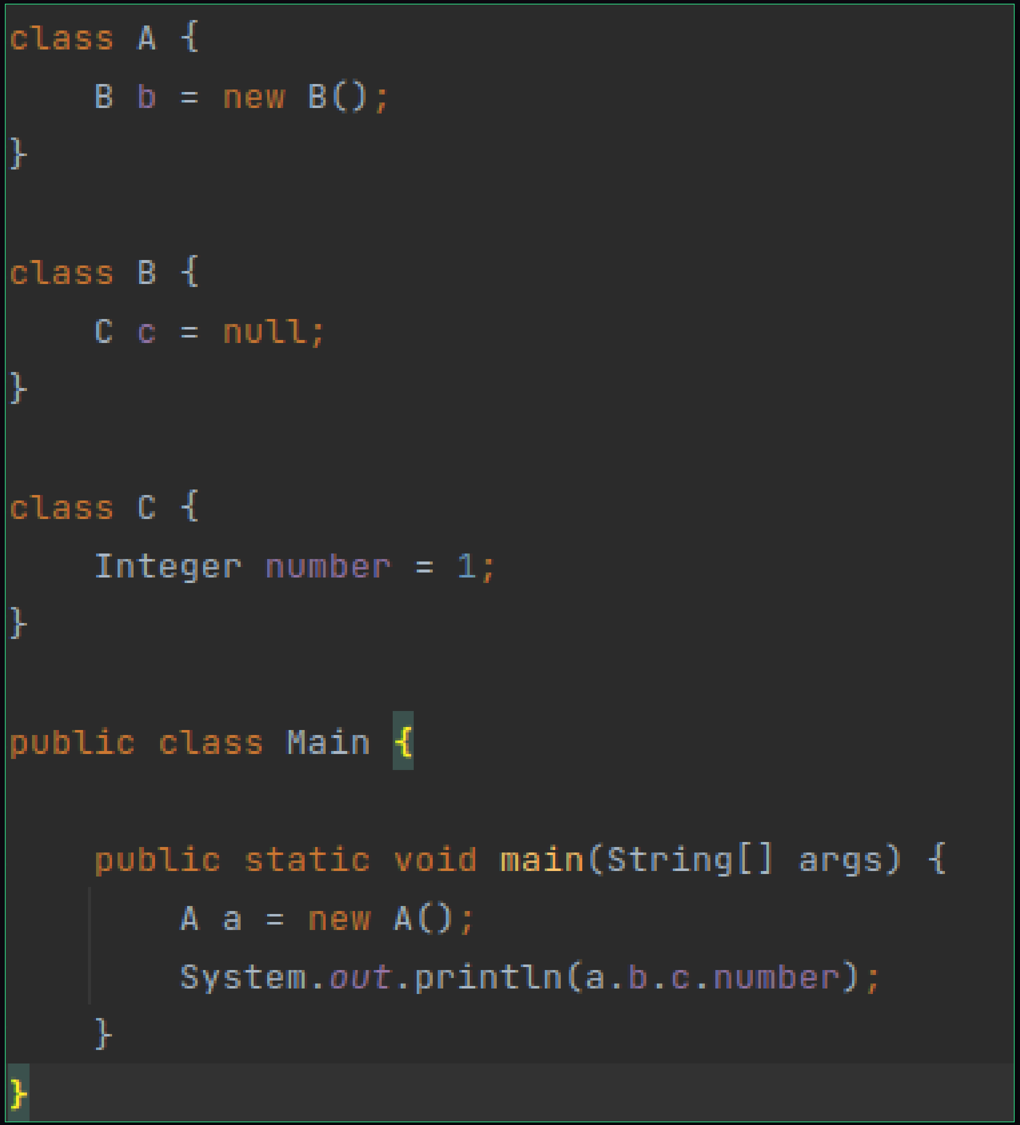


```java
class A {
    B b = new B();
}

class B {
    C c = null;
}

class C {
    Integer number = 1;
}

public class Main {

    public static void main(String[] args) {
        A a = new A();
        System.out.println(a.b.c.number);
    }
}
```

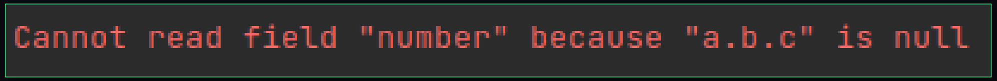
JDK 14



```
Exception in thread "main" java.lang.NullPointerException Create breakpoint : Cannot read field "number" because "a.b.c" is null
	at com.yavaconf._1_npe.Main.main(Main.java:19)

Process finished with exit code 1
```



```
class A {
    B b = new B();
}

class B {
    C c = null;
}

class C {
    Integer number = 1;
}

public class Main {

    public static void main(String[] args) {
        A a = new A();
        System.out.println(a.b.c.number);
    }
}
```



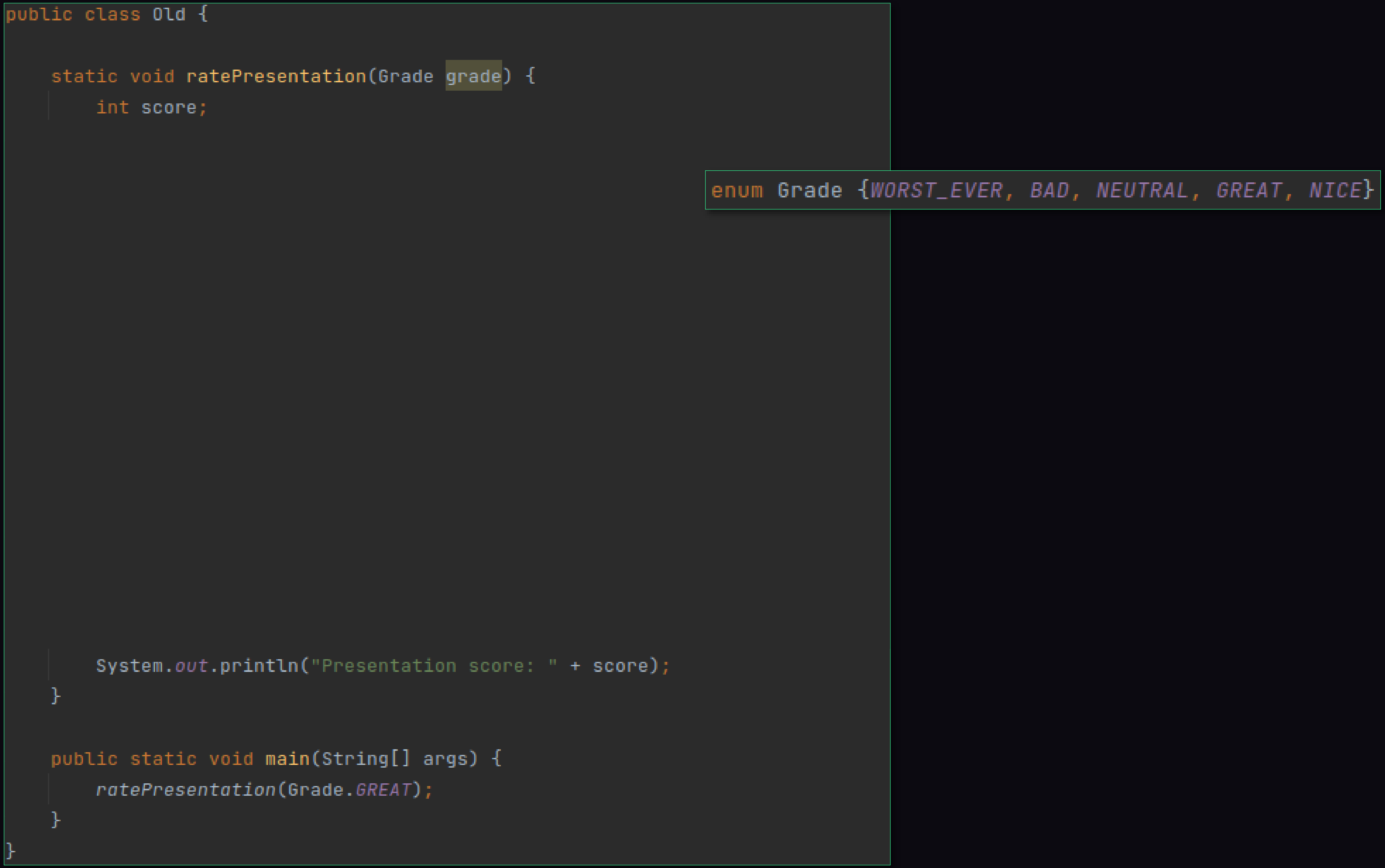
```
Cannot read field "number" because "a.b.c" is null
```

Gdyby kod mógł mówić?

Co by o Was powiedział?

Switch

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

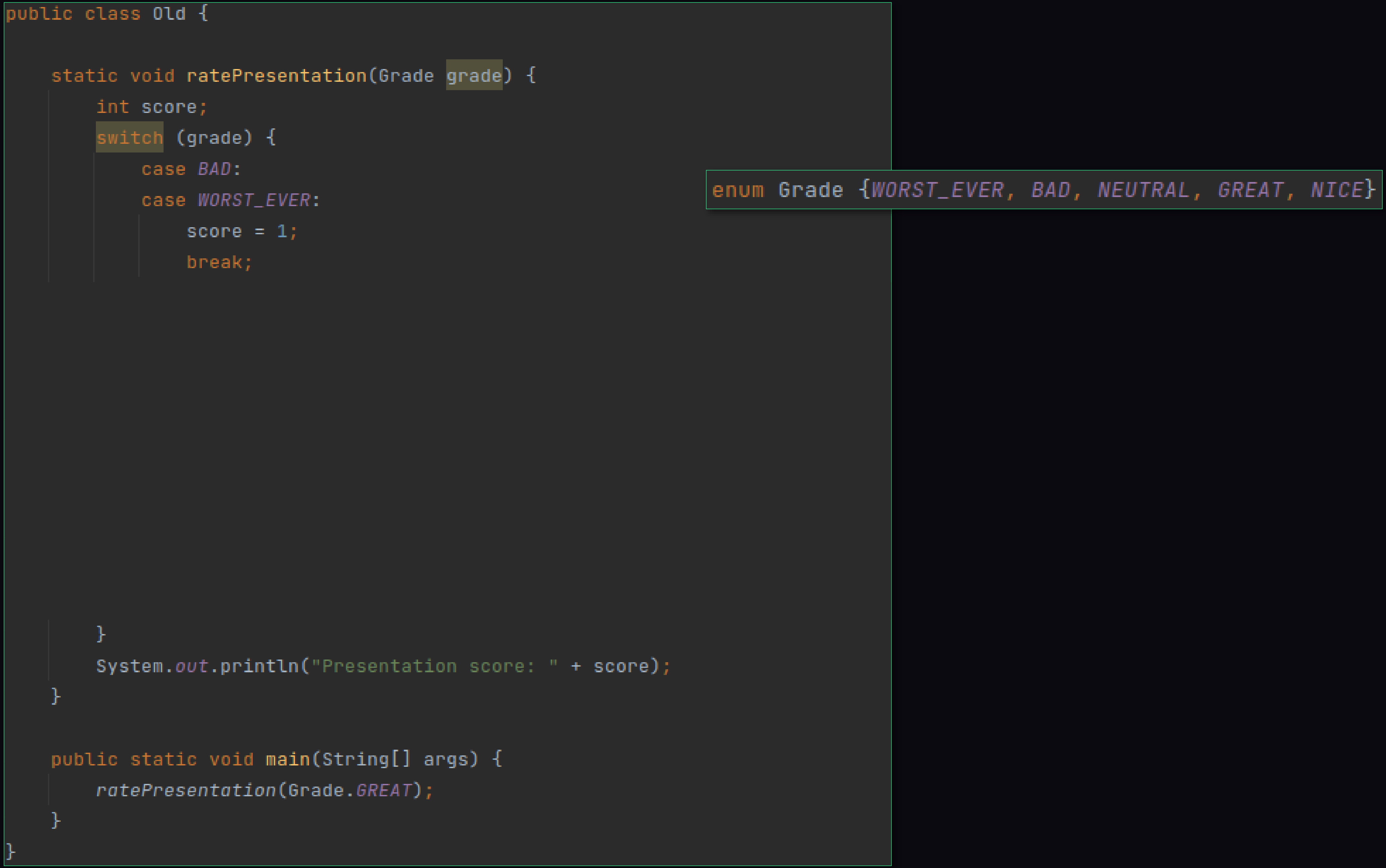


```java
public class Old {

    static void ratePresentation(Grade grade) {
        int score;

















        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

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

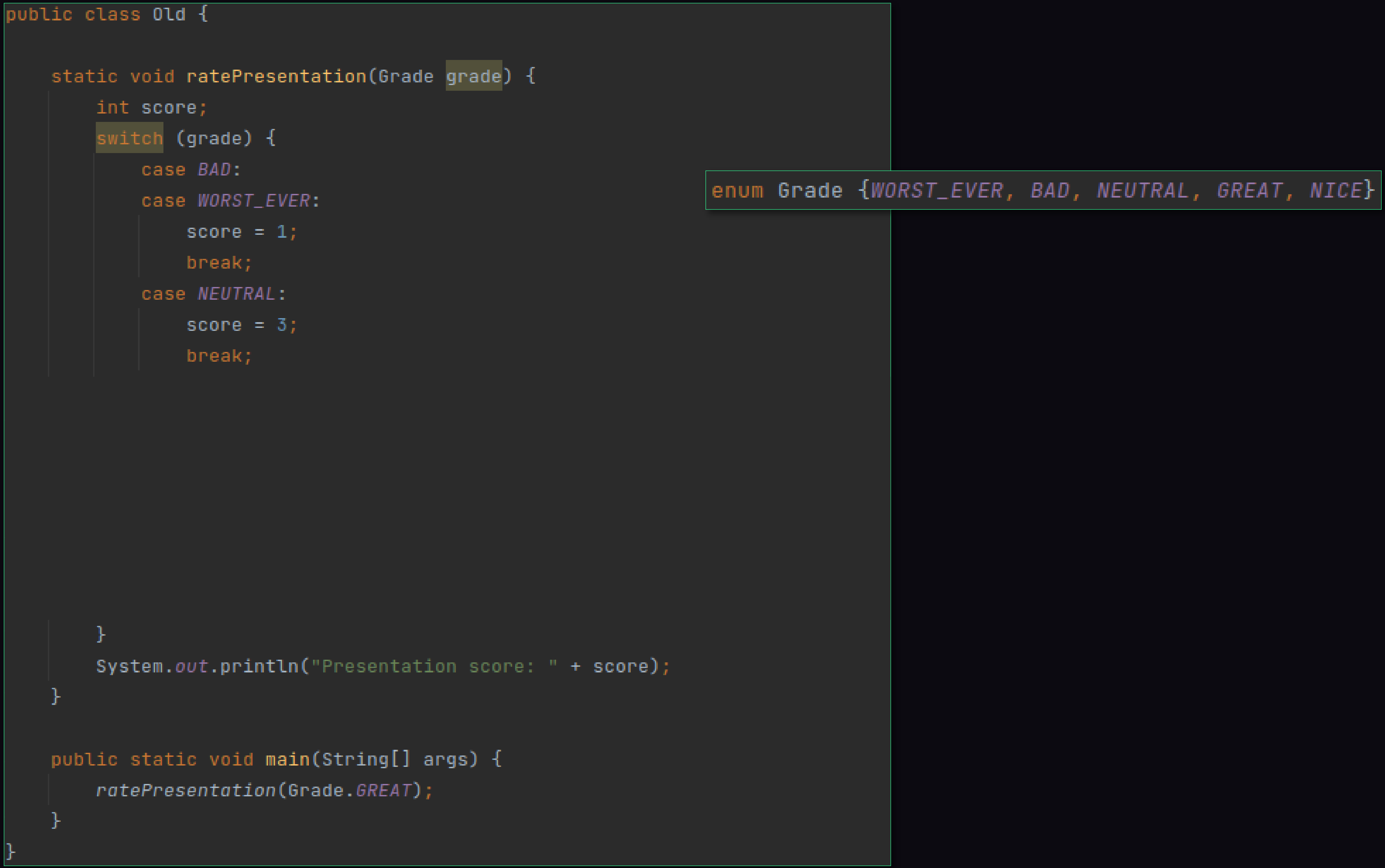
```java
public class Old {

    static void ratePresentation(Grade grade) {
        int score;
        switch (grade) {

        }
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

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

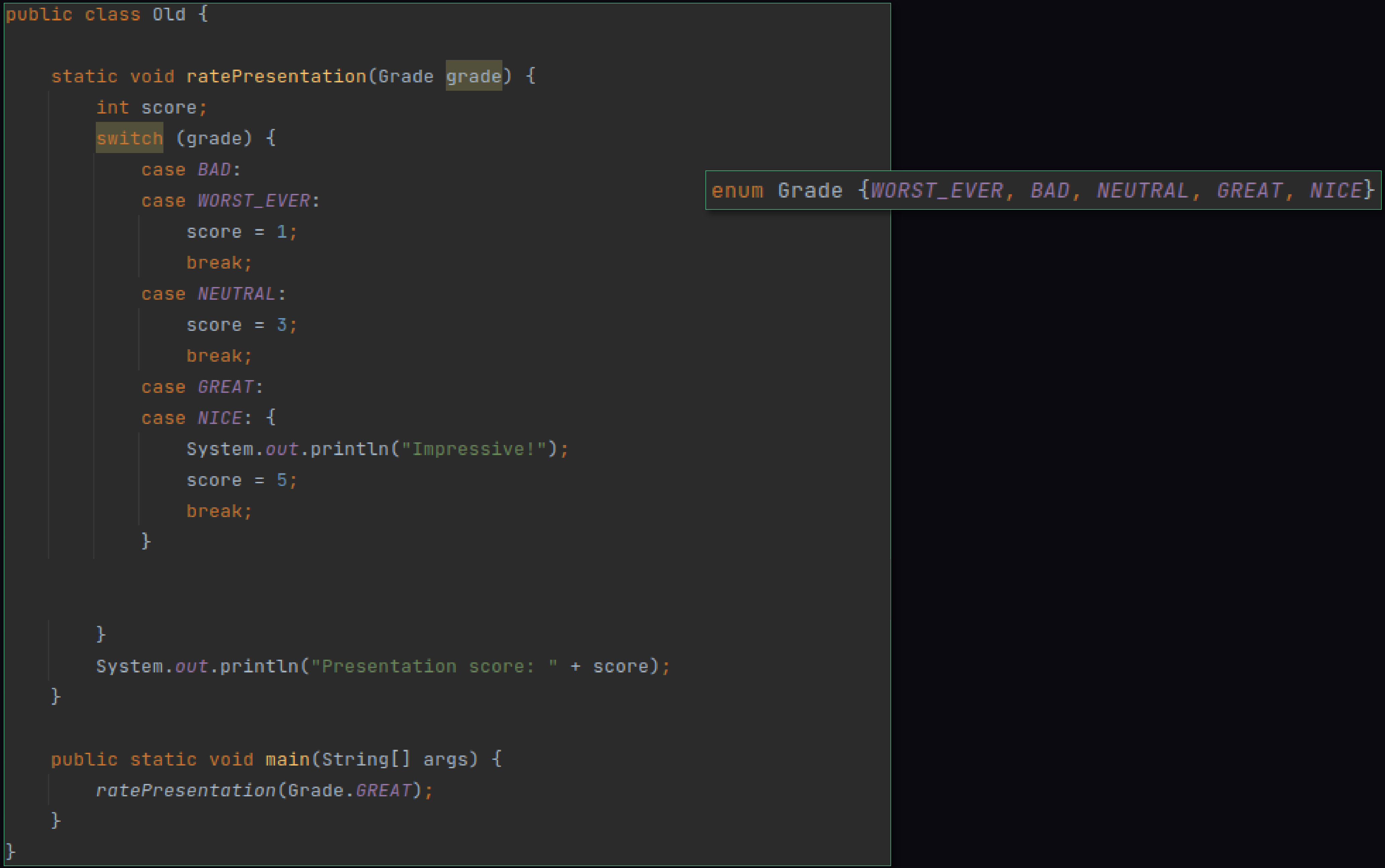


```
public class Old {

    static void ratePresentation(Grade grade) {
        int score;
        switch (grade) {
            case BAD:
            case WORST_EVER:
                score = 1;
                break;









        }
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

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

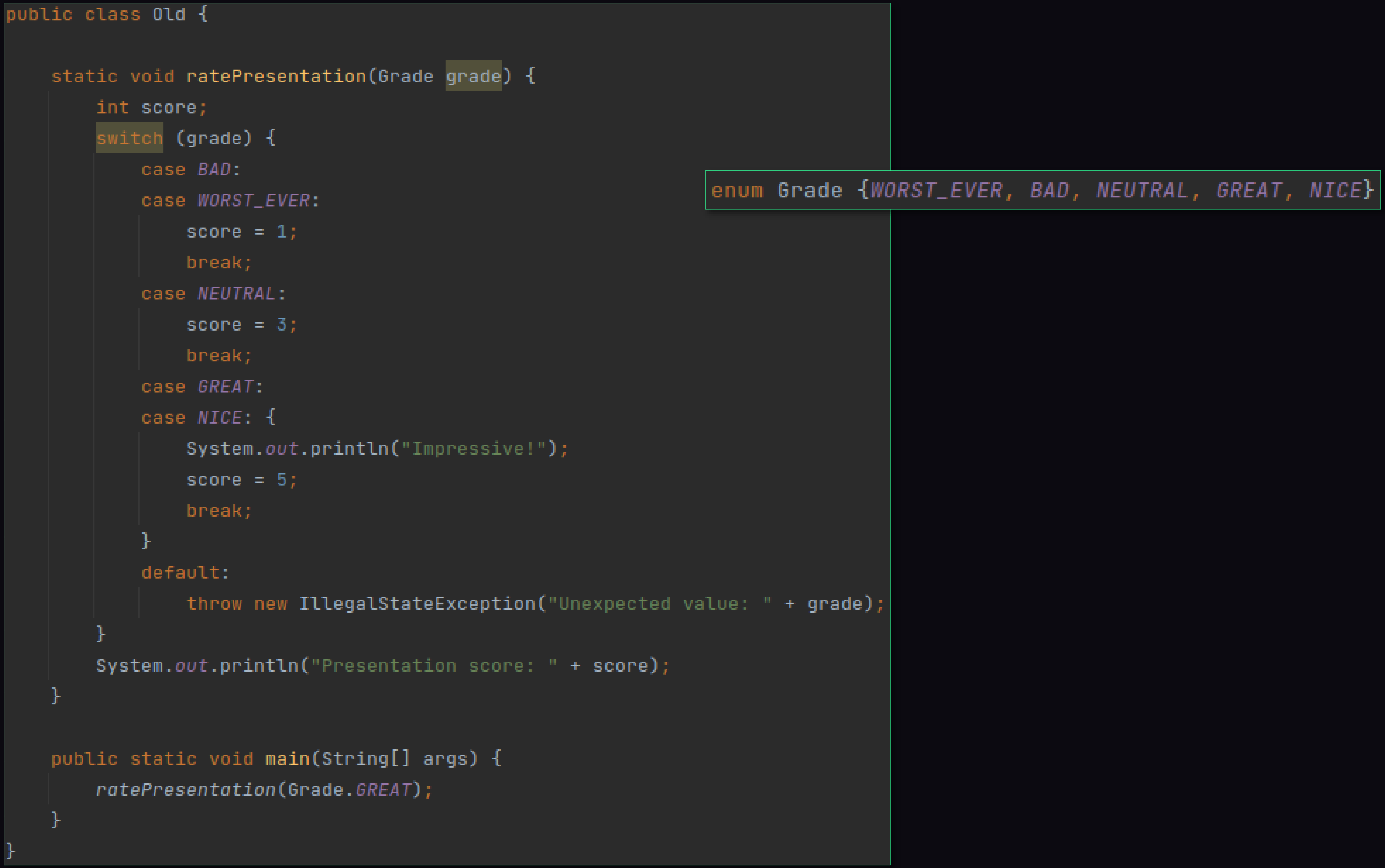

```java
public class Old {

    static void ratePresentation(Grade grade) {
        int score;
        switch (grade) {
            case BAD:
            case WORST_EVER:
                score = 1;
                break;
            case NEUTRAL:
                score = 3;
                break;







        }
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

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



```java
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;
            }


        }
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

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



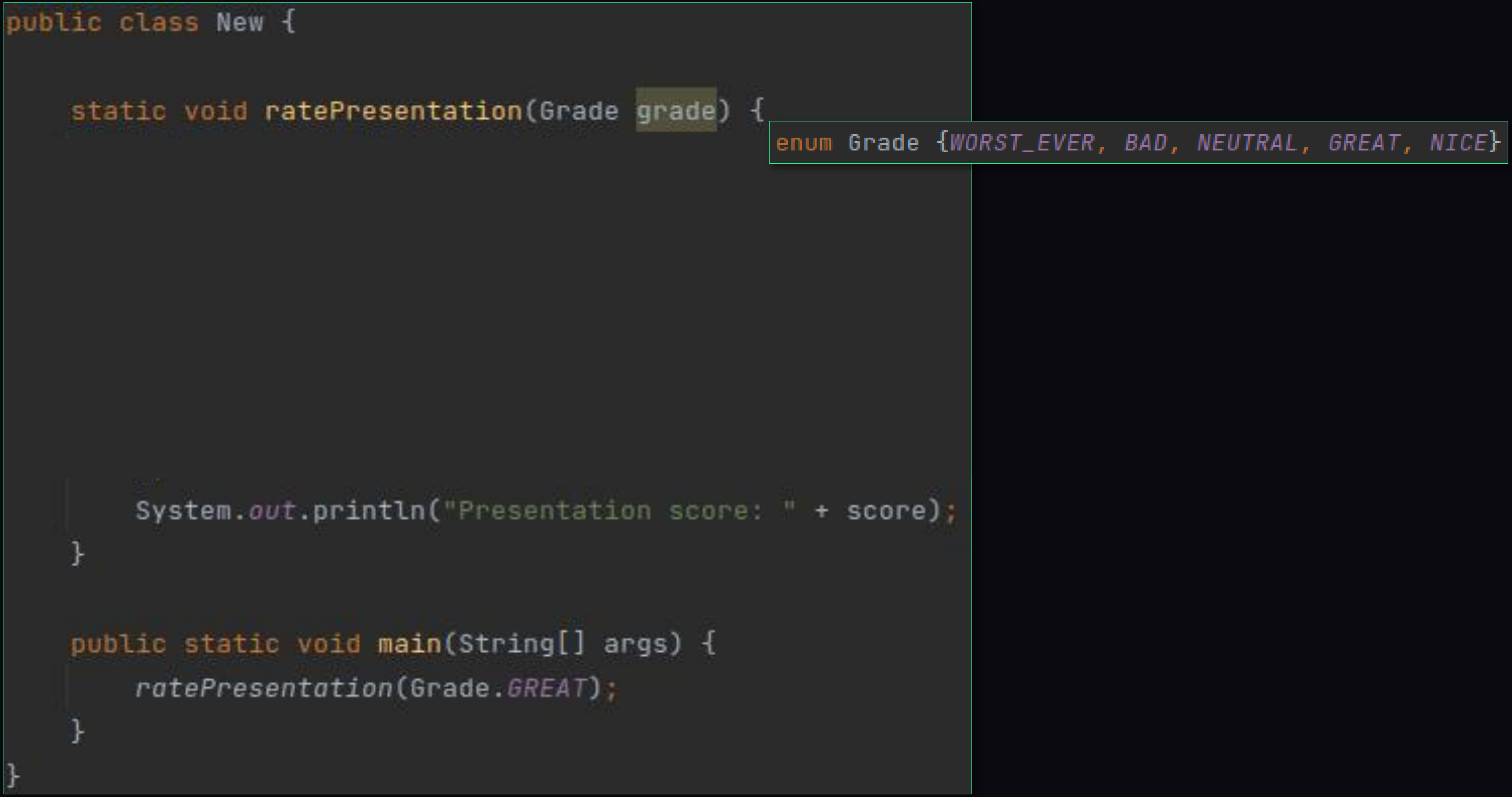
```java
public class Old {

    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);
    }
}
```

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

Switch expressions

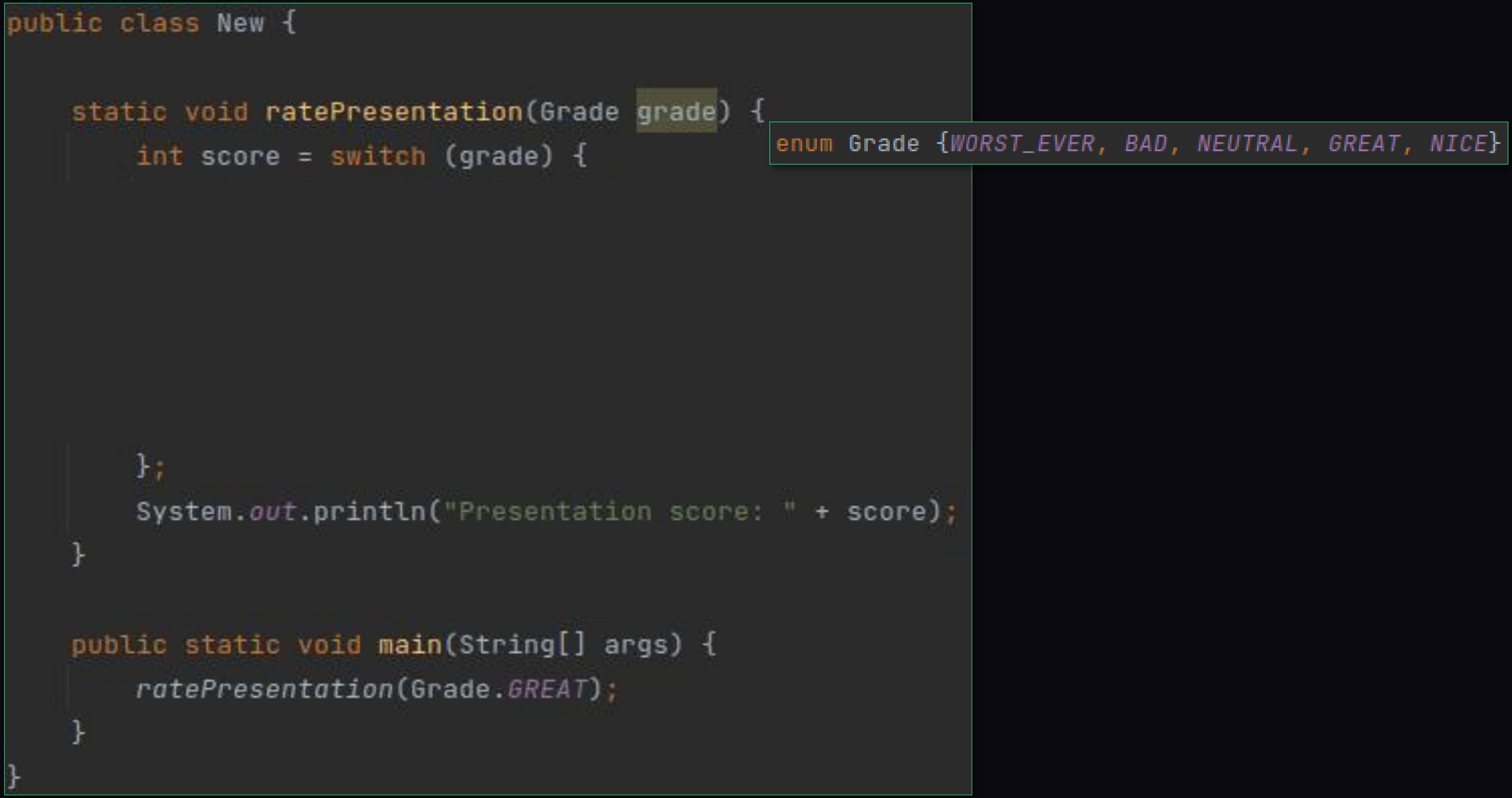
Nowe podejście

JDK 14



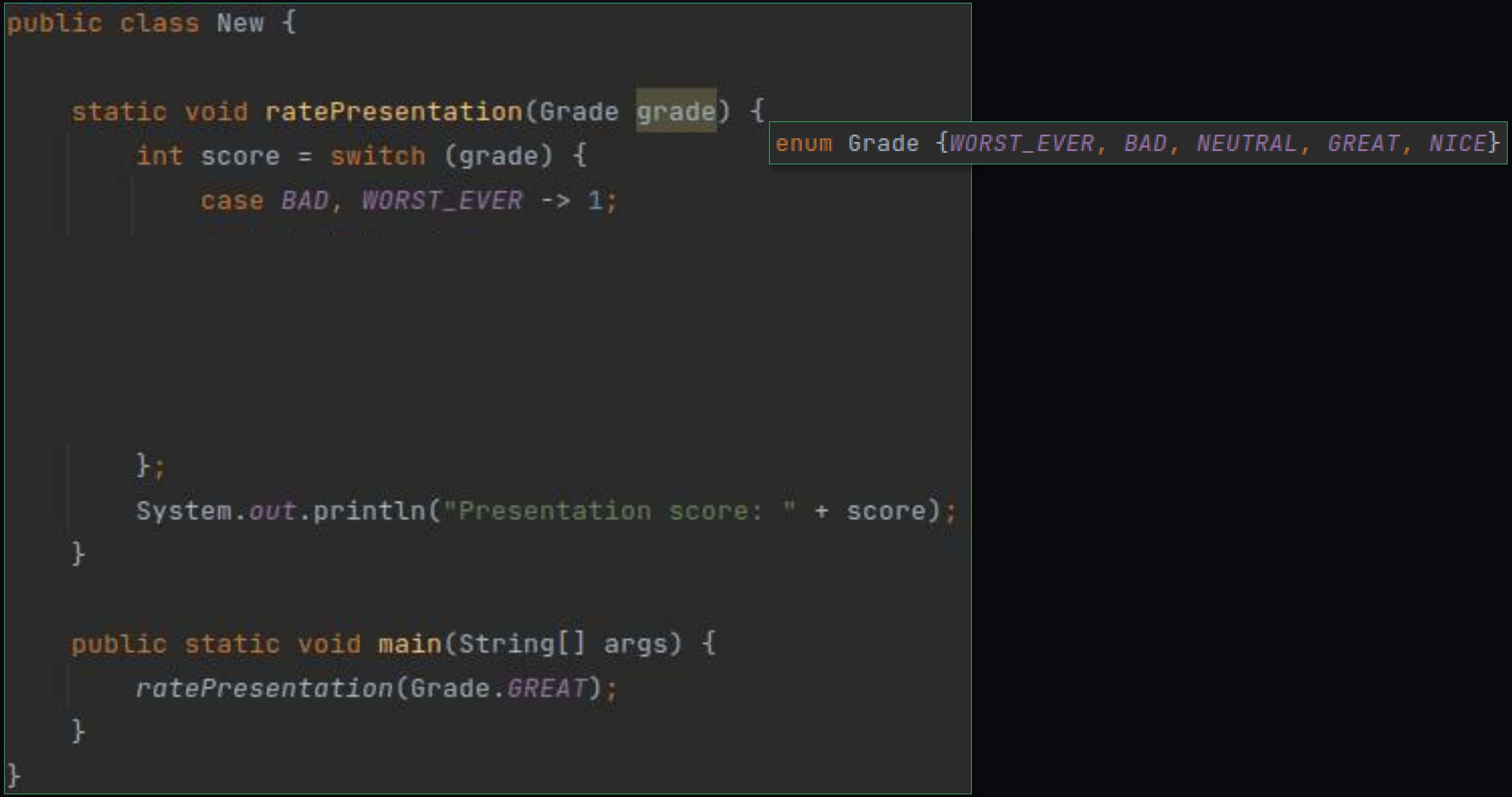
```java
public class New {

    static void ratePresentation(Grade grade) {









        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

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



```
public class New {

    static void ratePresentation(Grade grade) {
        int score = switch (grade) {







        };
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

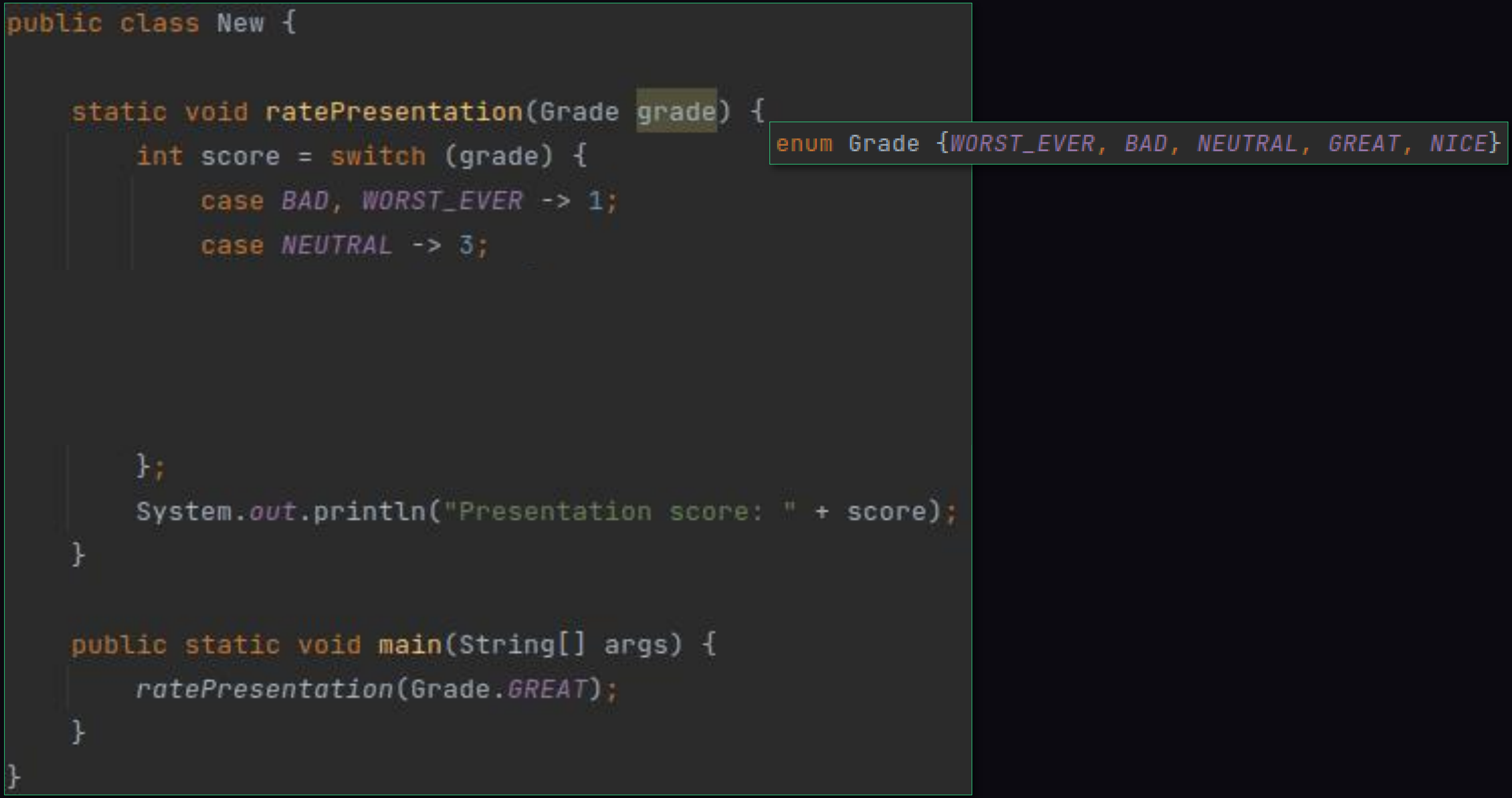
```
enum Grade {WORST_EVER, BAD, NEUTRAL, GREAT, NICE}
```



```
public class New {

    static void ratePresentation(Grade grade) {
        int score = switch (grade) {
            case BAD, WORST_EVER -> 1;





        };
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

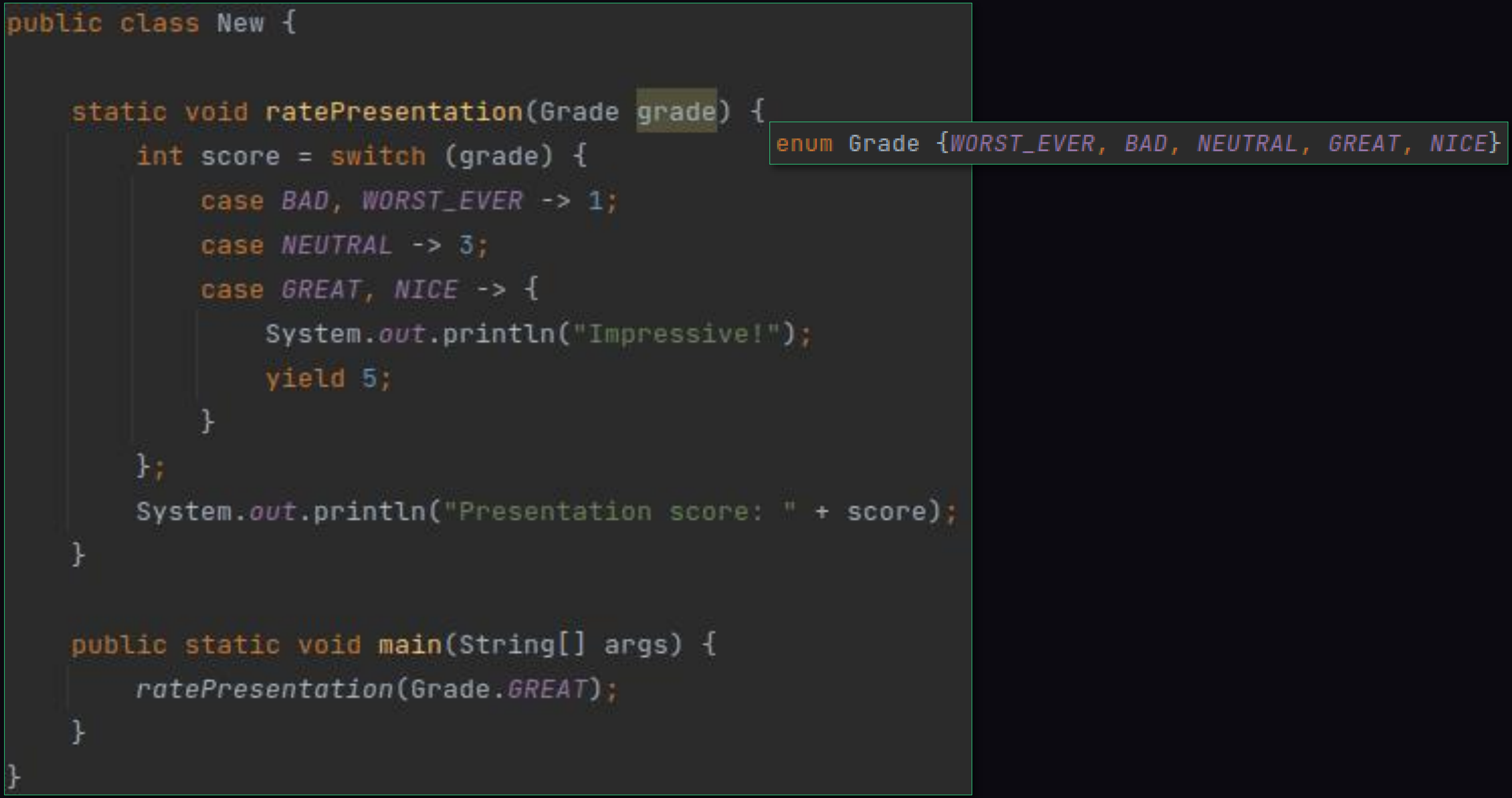
```
enum Grade {WORST_EVER, BAD, NEUTRAL, GREAT, NICE}
```

JDK 14



```
public class New {

    static void ratePresentation(Grade grade) {
        int score = switch (grade) {
            case BAD, WORST_EVER -> 1;
            case NEUTRAL -> 3;




        };
        System.out.println("Presentation score: " + score);
    }

    public static void main(String[] args) {
        ratePresentation(Grade.GREAT);
    }
}
```

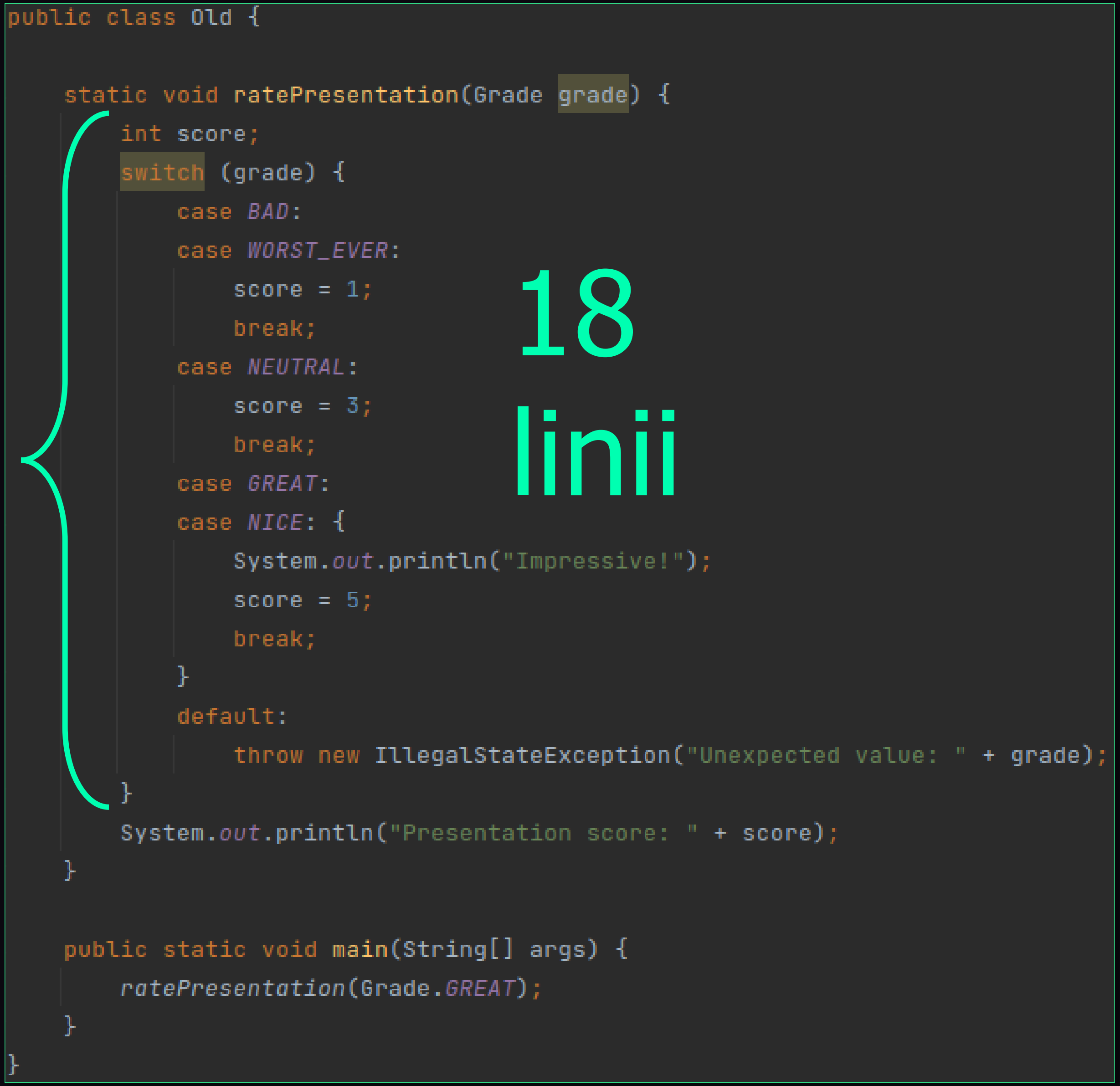
```
enum Grade {WORST_EVER, BAD, NEUTRAL, GREAT, NICE}
```



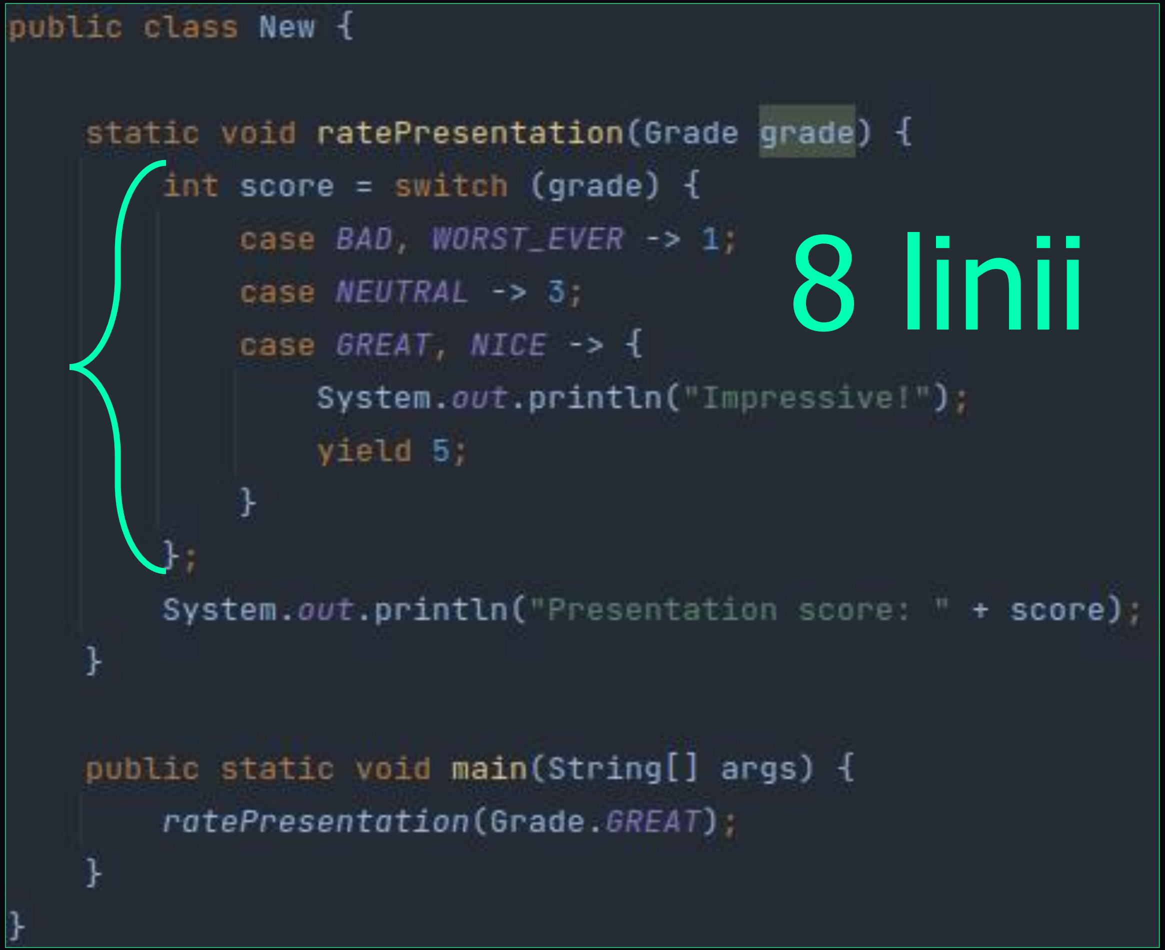
```
public class New {

    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}
```

JDK 14



```
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);
    }
}
```



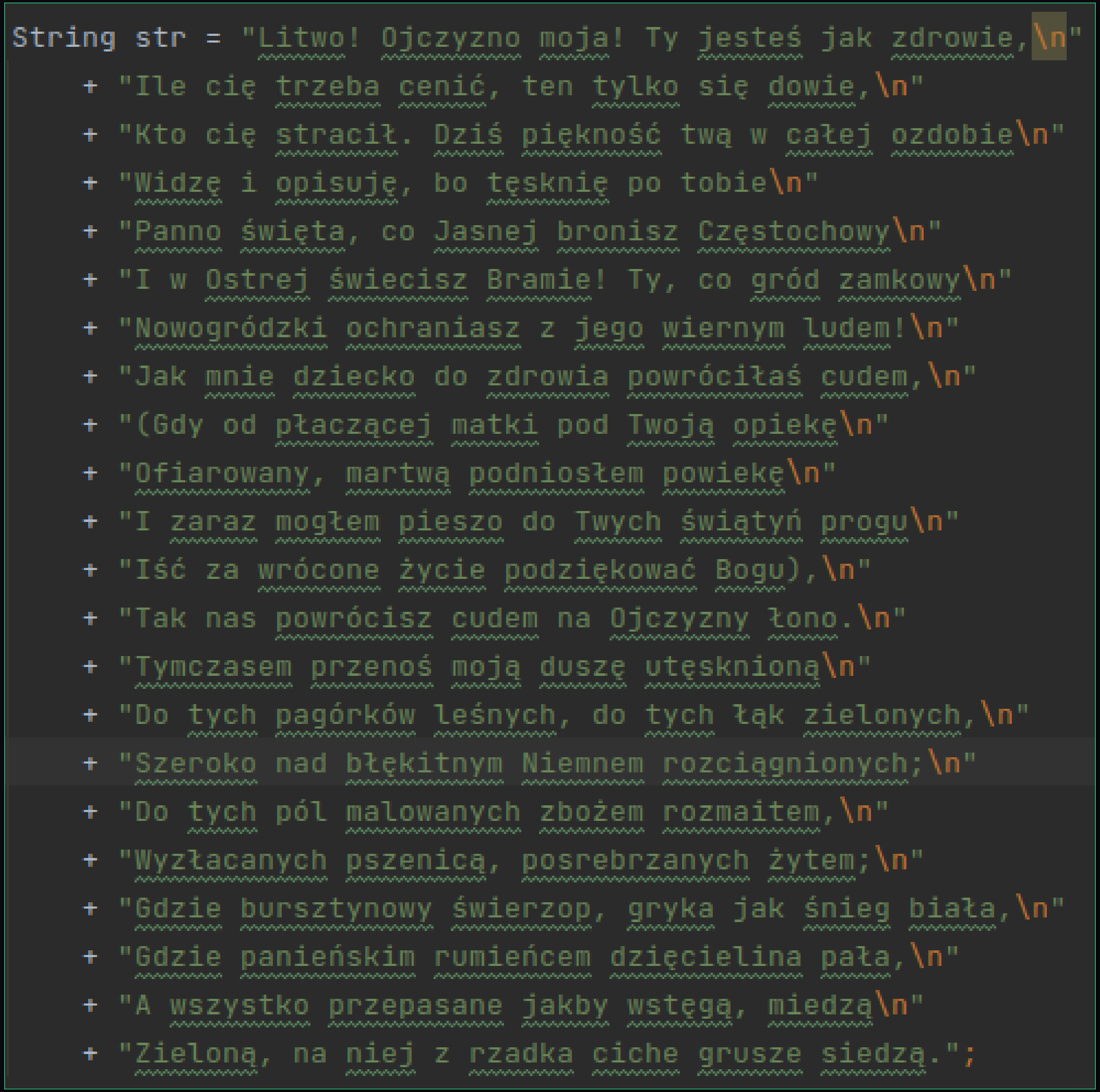
```
public class New {

    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);
    }
}
```

Ile jest planet w układzie słonecznym?

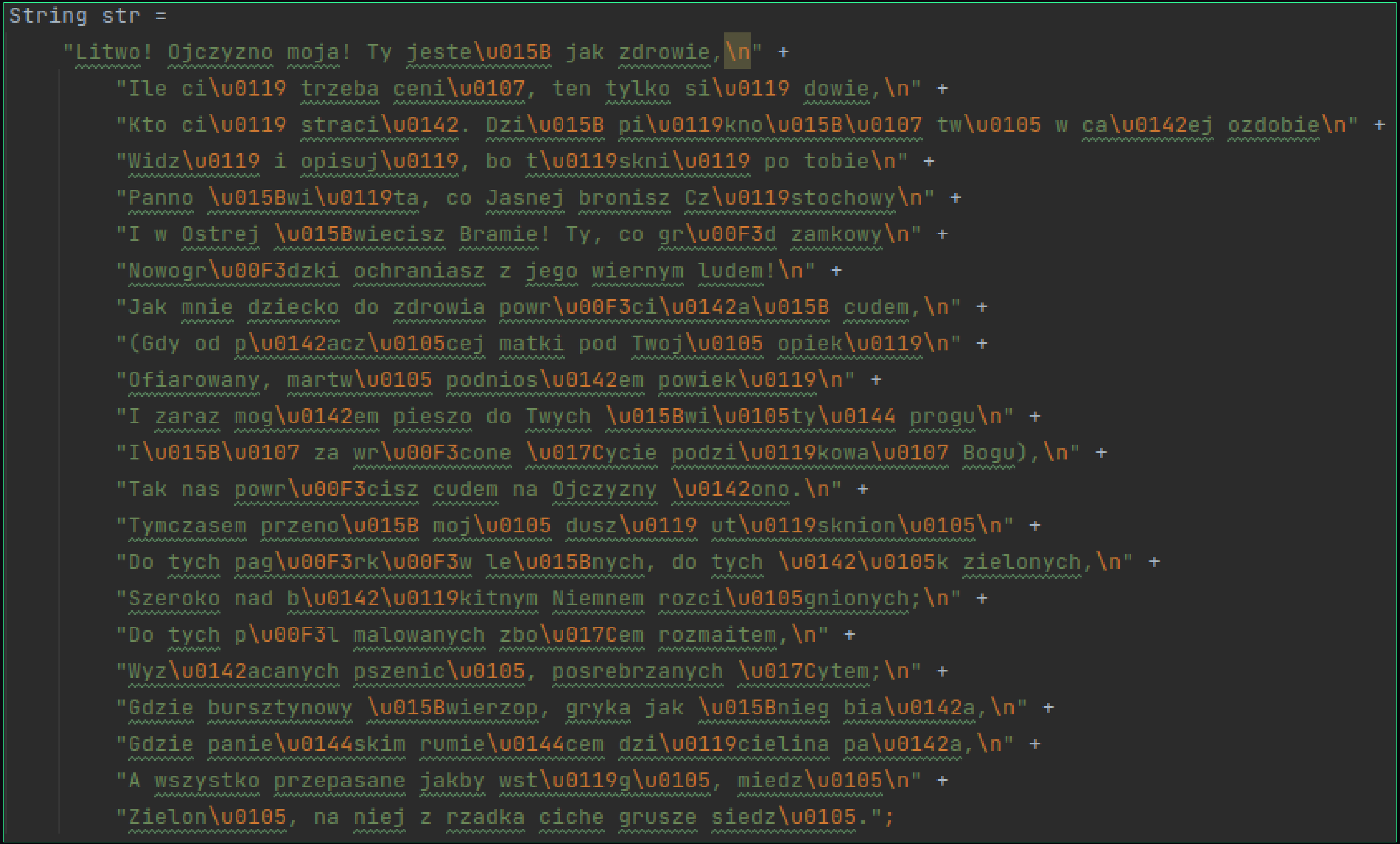
Usunięcie
Nashorn Javascript

Image: freepik.com

Co Adam Mickiewicz sądził o stringach?



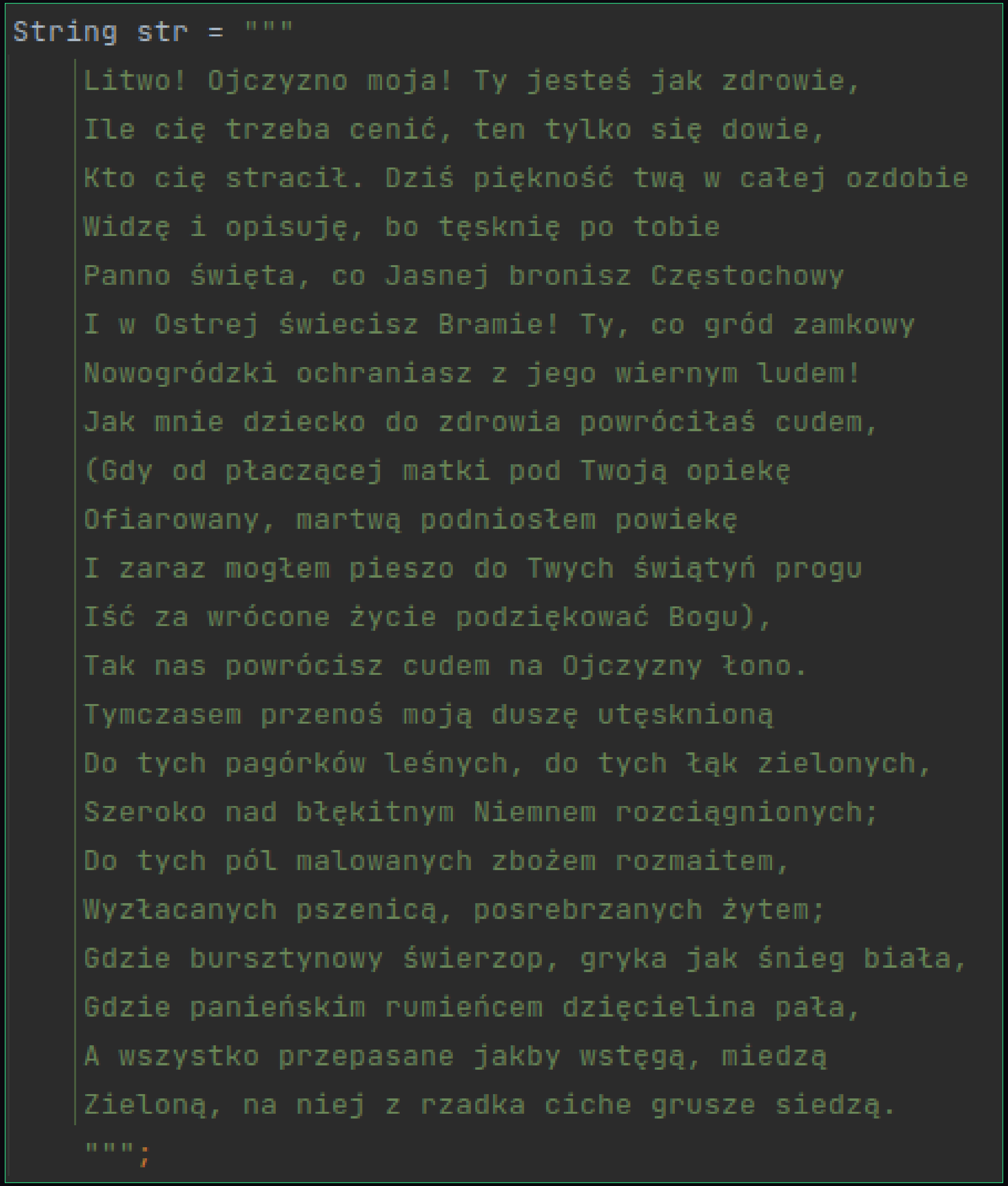
```java
String str = "Litwo! Ojczyzno moja! Ty jesteś jak zdrowie,\n"
        + "Ile cię trzeba cenić, ten tylko się dowie,\n"
        + "Kto cię stracił. Dziś piękność twą w całej ozdobie\n"
        + "Widzę i opisuję, bo tęsknię po tobie\n"
        + "Panno święta, co Jasnej bronisz Częstochowy\n"
        + "I w Ostrej świecisz Bramie! Ty, co gród zamkowy\n"
        + "Nowogródzki ochraniasz z jego wiernym ludem!\n"
        + "Jak mnie dziecko do zdrowia powróciłaś cudem,\n"
        + "(Gdy od płaczącej matki pod Twoją opiekę\n"
        + "Ofiarowany, martwą podniosłem powiekę\n"
        + "I zaraz mogłem pieszo do Twych świątyń progu\n"
        + "Iść za wrócone życie podziękować Bogu),\n"
        + "Tak nas powrócisz cudem na Ojczyzny łono.\n"
        + "Tymczasem przenoś moją duszę utęsknioną\n"
        + "Do tych pagórków leśnych, do tych łąk zielonych,\n"
        + "Szeroko nad błękitnym Niemnem rozciągnionych;\n"
        + "Do tych pól malowanych zbożem rozmaitem,\n"
        + "Wyzłacanych pszenicą, posrebrzanych żytem;\n"
        + "Gdzie bursztynowy świerzop, gryka jak śnieg biała,\n"
        + "Gdzie panieńskim rumieńcem dzięcielina pała,\n"
        + "A wszystko przepasane jakby wstęgą, miedzą\n"
        + "Zieloną, na niej z rzadka ciche grusze siedzą.";
```



```java
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 Twoj\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\u00F3cisz cudem na Ojczyzny \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\u0142acanych 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.";
```

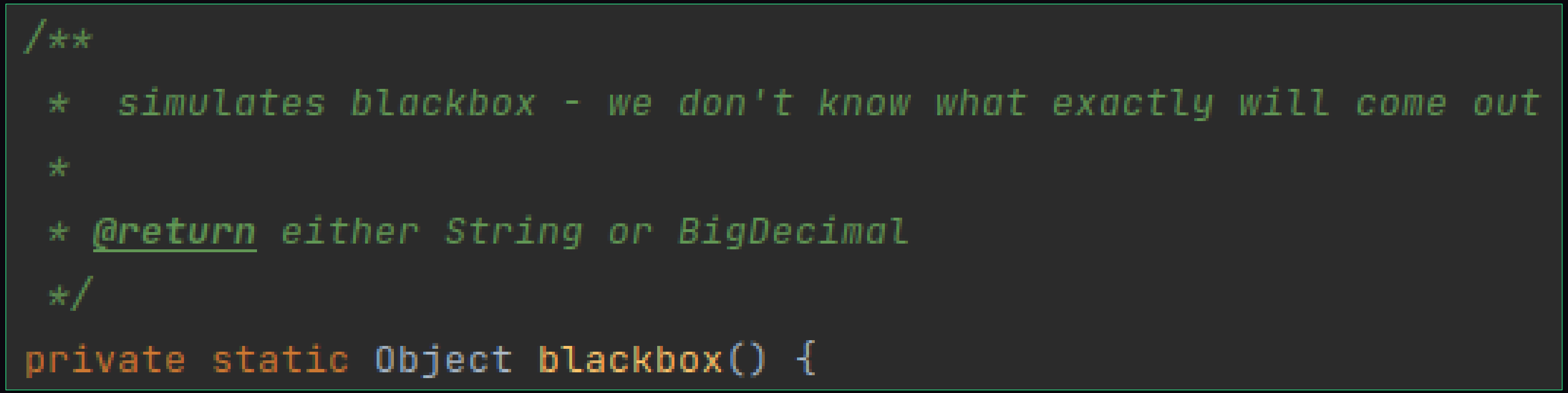
Textblocks

JDK 15

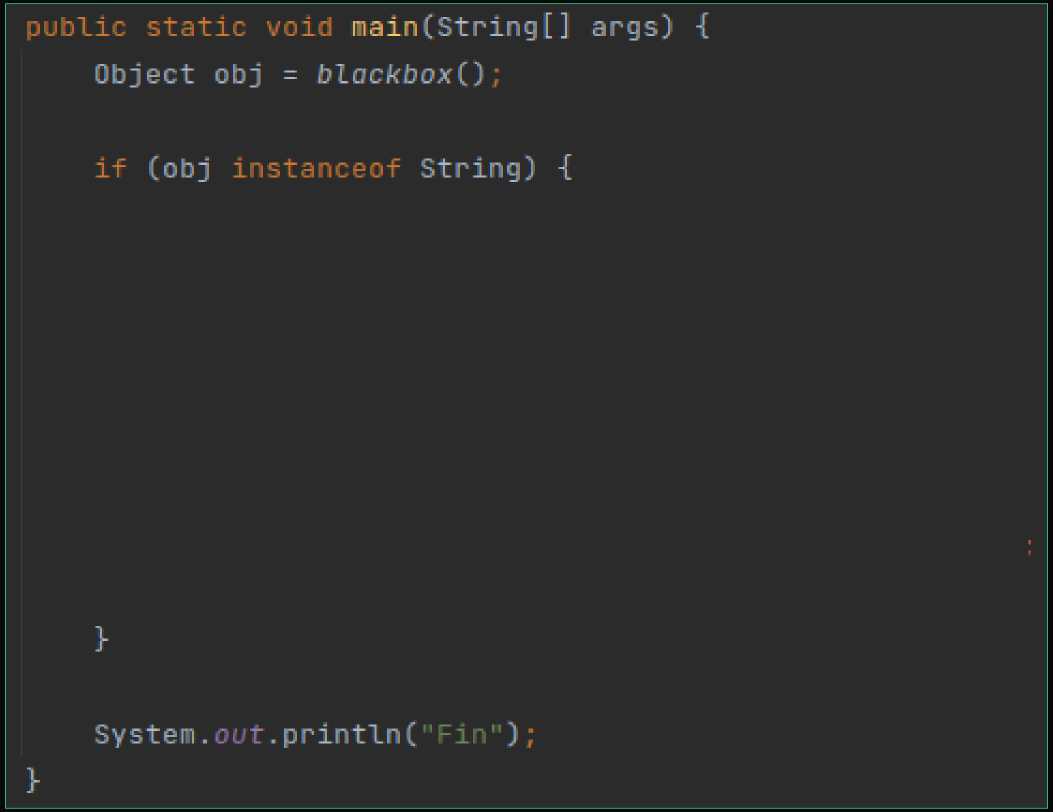


```java
String str = """
    Litwo! Ojczyzno moja! Ty jesteś jak zdrowie,
    Ile cię trzeba cenić, ten tylko się dowie,
    Kto cię stracił. Dziś piękność twą w całej ozdobie
    Widzę i opisuję, bo tęsknię po tobie
    Panno święta, co Jasnej bronisz Częstochowy
    I w Ostrej świecisz Bramie! Ty, co gród zamkowy
    Nowogródzki ochraniasz z jego wiernym ludem!
    Jak mnie dziecko do zdrowia powróciłaś cudem,
    (Gdy od płaczącej matki pod Twoją opiekę
    Ofiarowany, martwą podniosłem powiekę
    I zaraz mogłem pieszo do Twych świątyń progu
    Iść za wrócone życie podziękować Bogu),
    Tak nas powrócisz cudem na Ojczyzny łono.
    Tymczasem przenoś moją duszę utęsknioną
    Do tych pagórków leśnych, do tych łąk zielonych,
    Szeroko nad błękitnym Niemnem rozciągnionych;
    Do tych pól malowanych zbożem rozmaitem,
    Wyzłacanych pszenicą, posrebrzanych żytem;
    Gdzie bursztynowy świerzop, gryka jak śnieg biała,
    Gdzie panieńskim rumieńcem dzięcielina pała,
    A wszystko przepasane jakby wstęgą, miedzą
    Zieloną, na niej z rzadka ciche grusze siedzą.
    """;
```

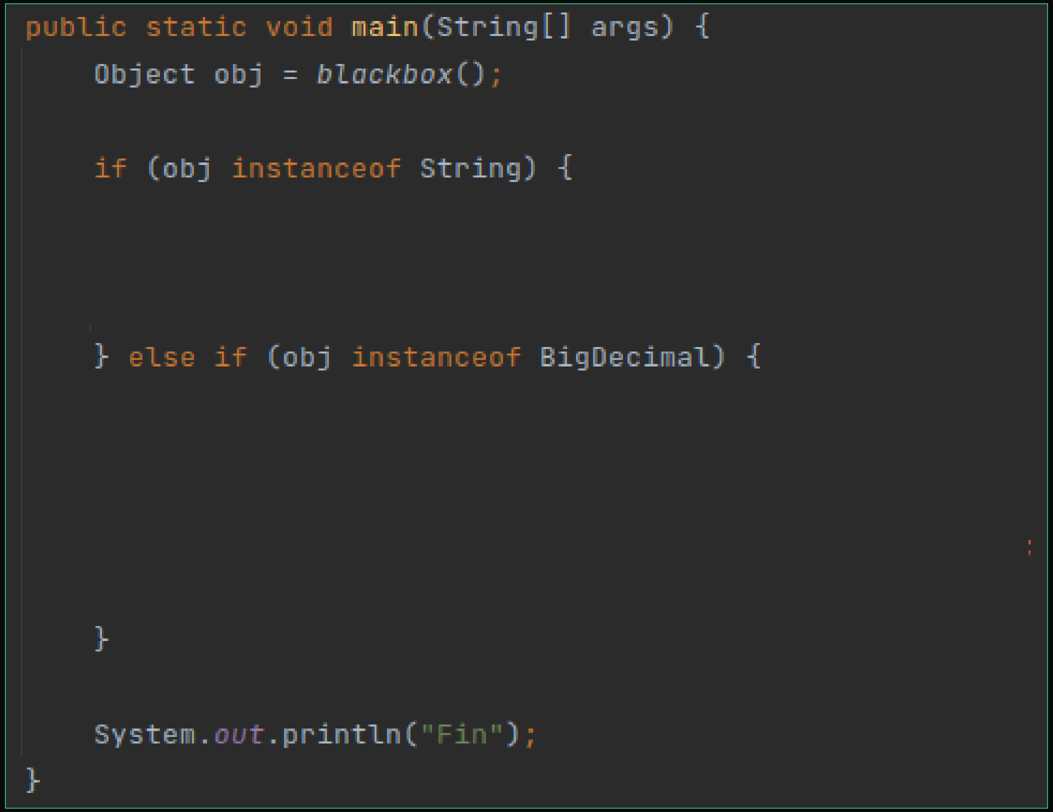
instanceof



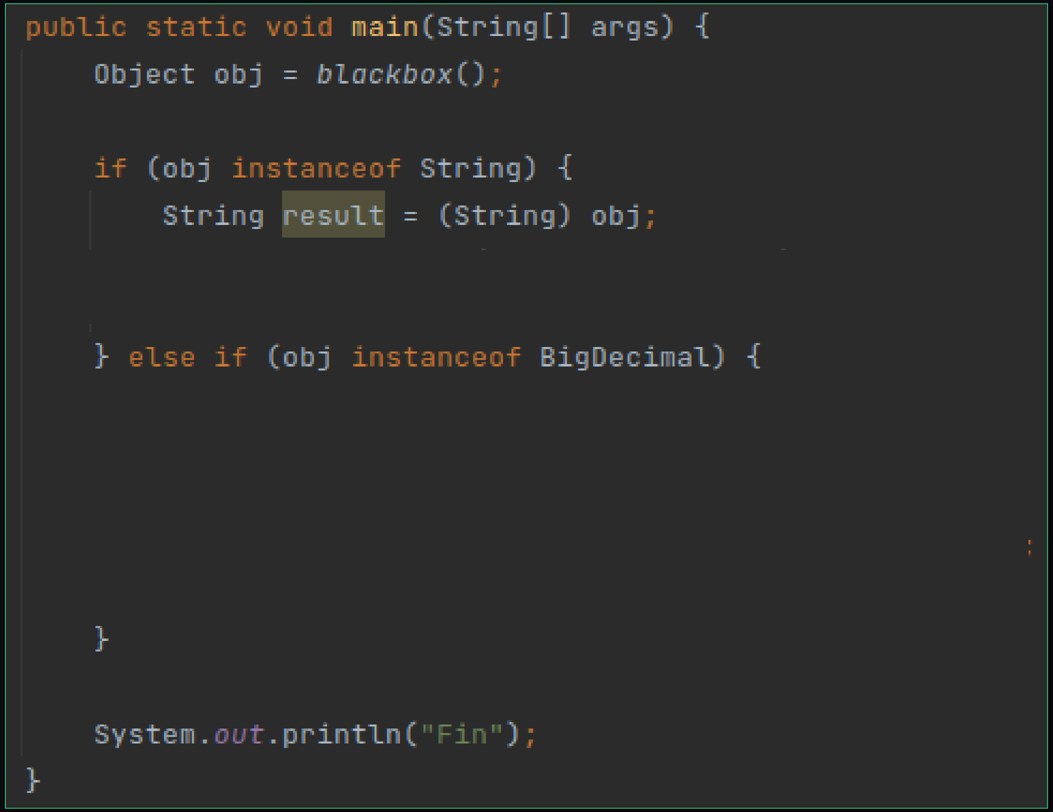
```
/**
 *  simulates blackbox - we don't know what exactly will come out
 *
 * @return either String or BigDecimal
 */
private static Object blackbox() {
```



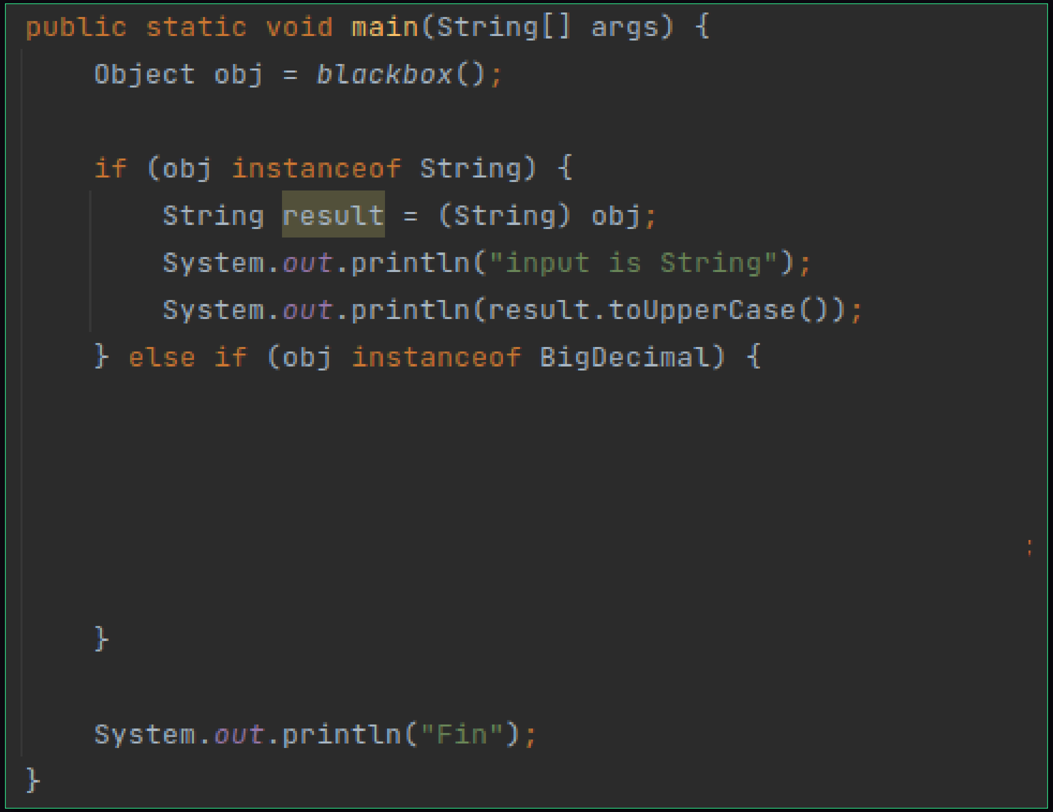
```java
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof String) {









    }

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



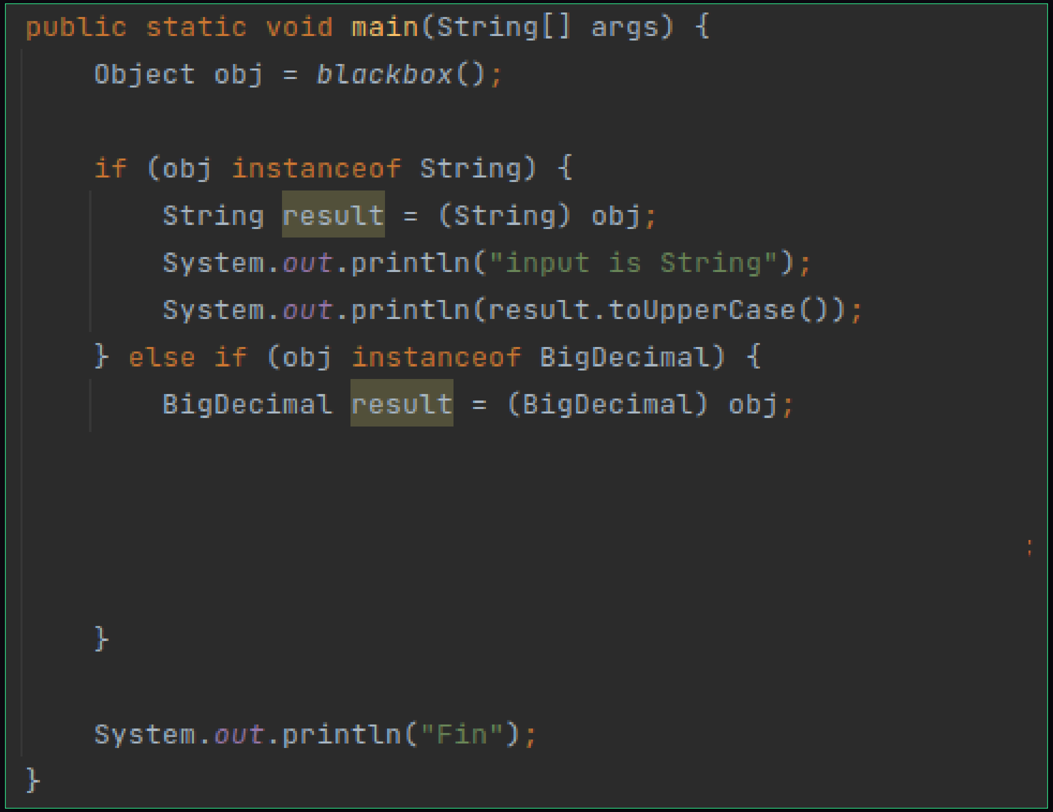
```
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof String) {



    } else if (obj instanceof BigDecimal) {






    }

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



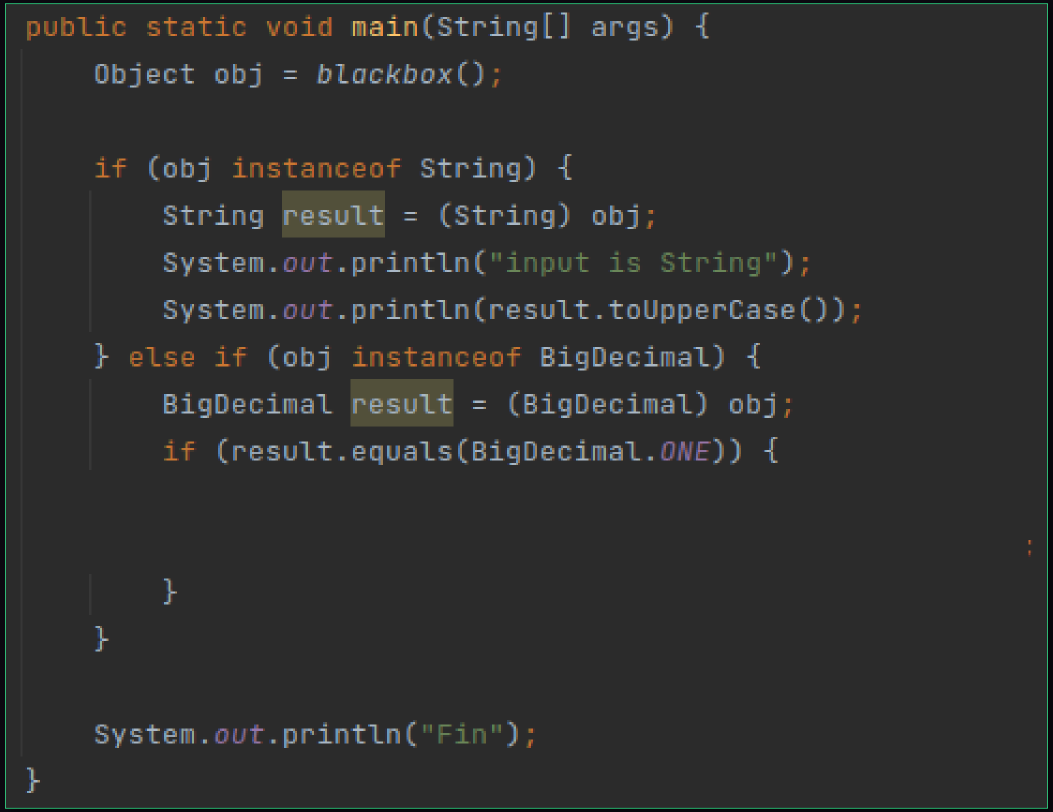
```java
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof String) {
        String result = (String) obj;


    } else if (obj instanceof BigDecimal) {





    }

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



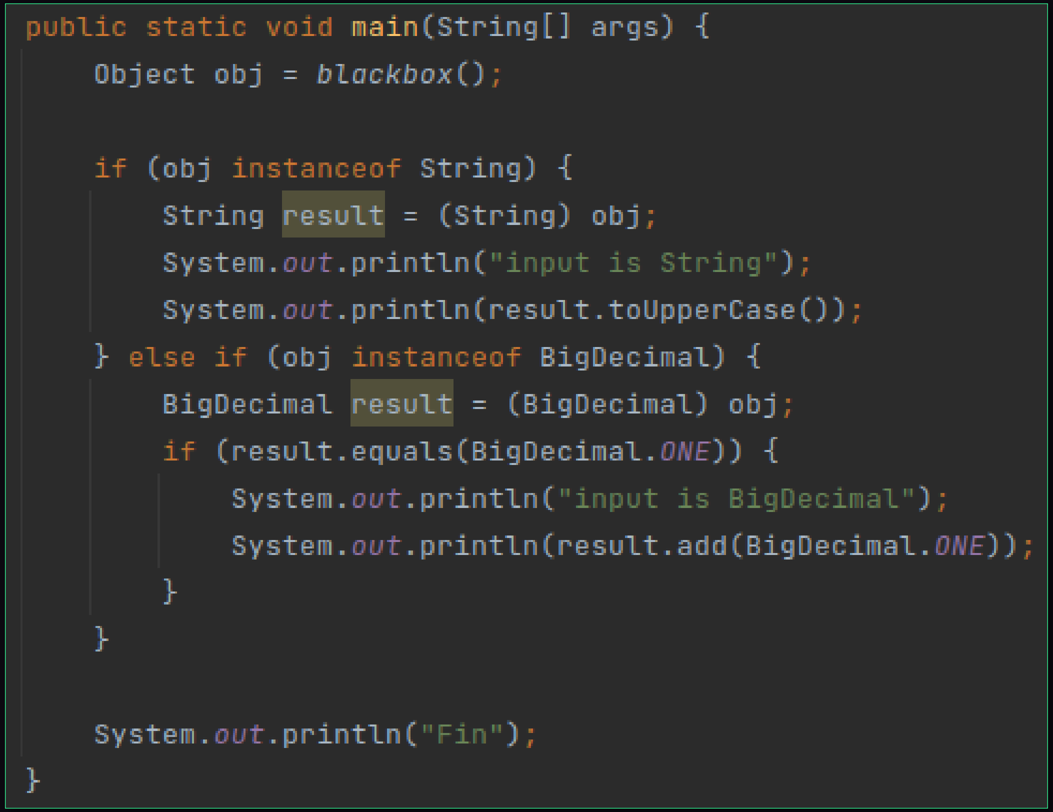
```java
public static void main(String[] args) {
    Object obj = blackbox();

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





    }

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



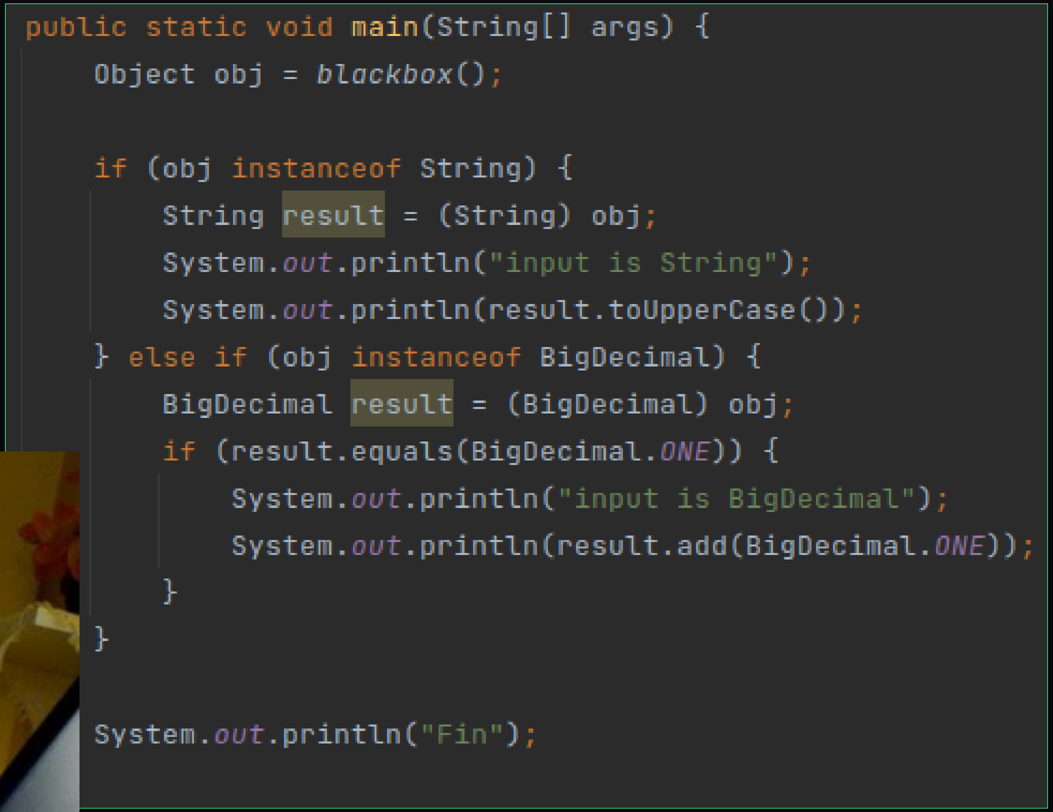
```
public static void main(String[] args) {
    Object obj = blackbox();

    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;




    }

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



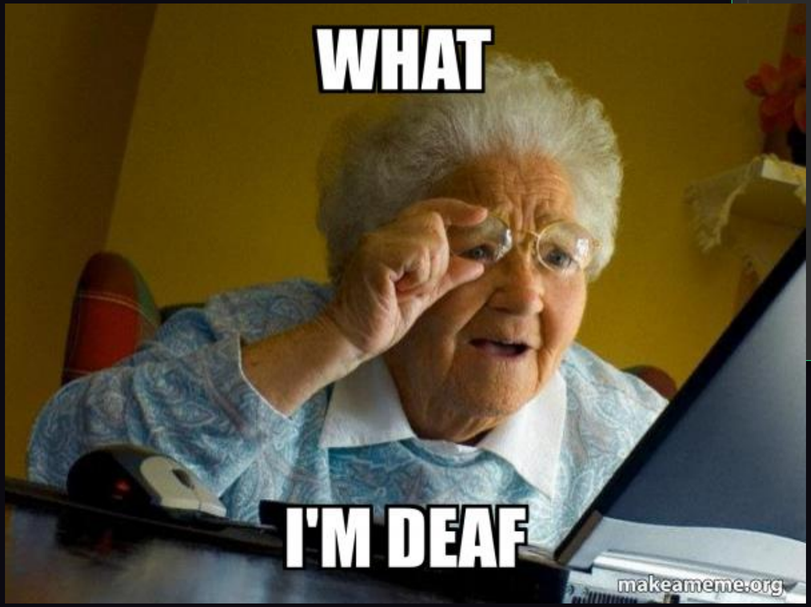
```java
public static void main(String[] args) {
    Object obj = blackbox();

    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("Fin");
}
```



```java
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");
}
```



```java
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");
```



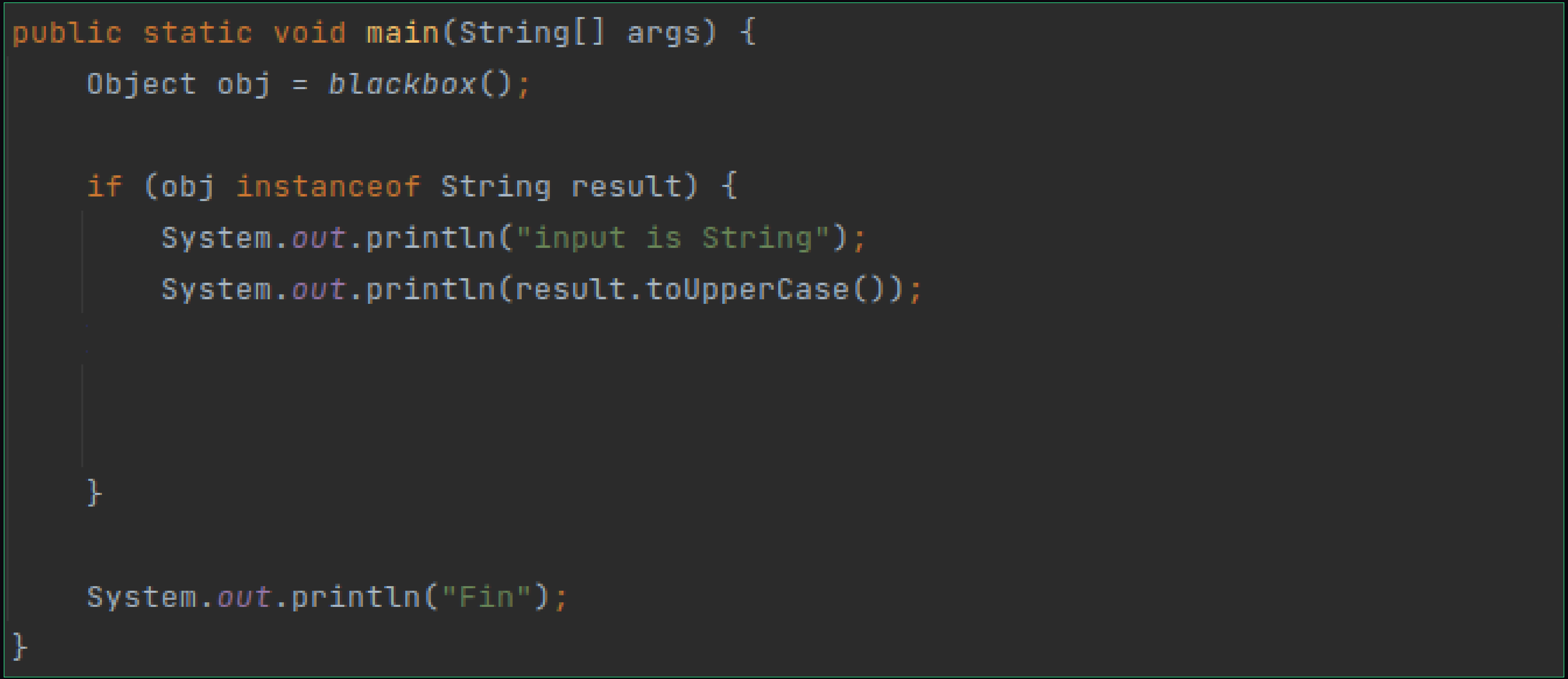
instanceof
bez rzutowania?

JDK 16

Pattern matching

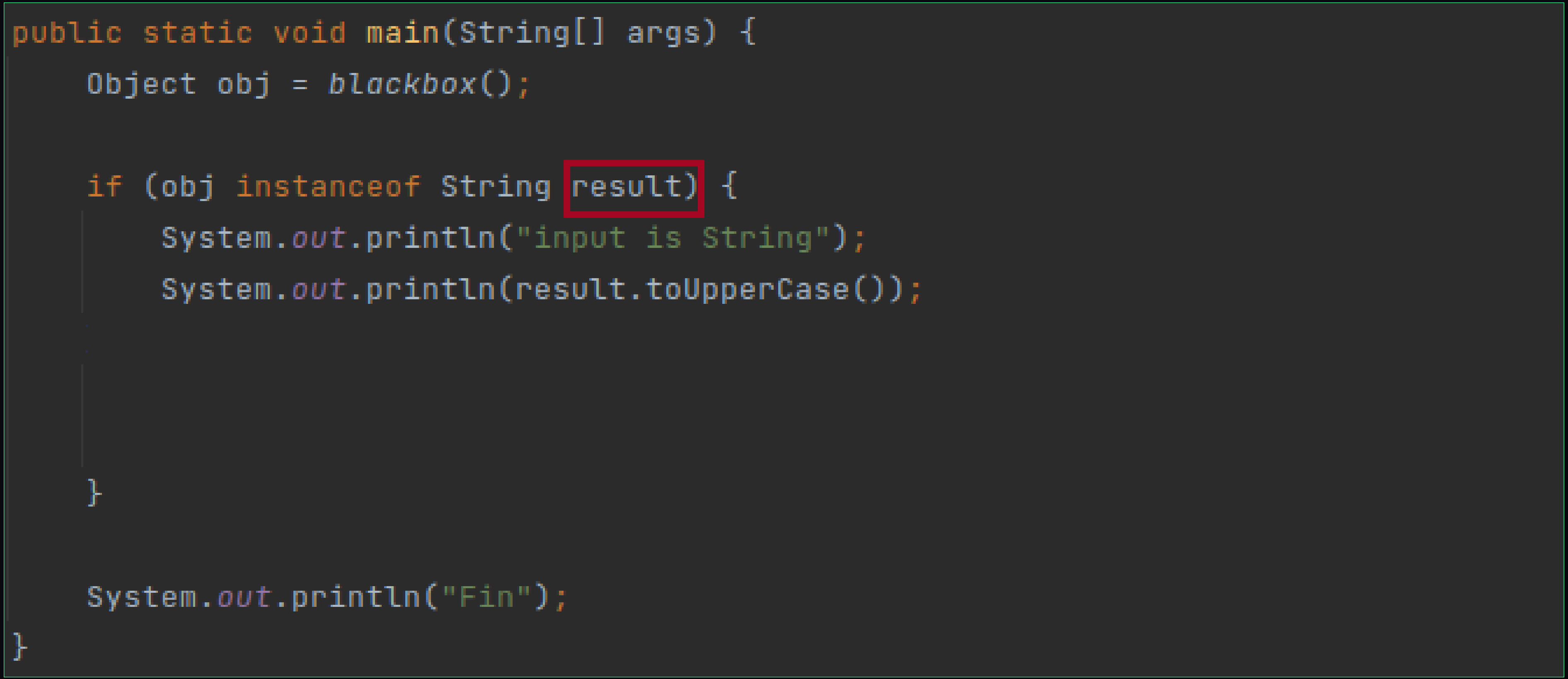
Image: freepik.com

JDK 16



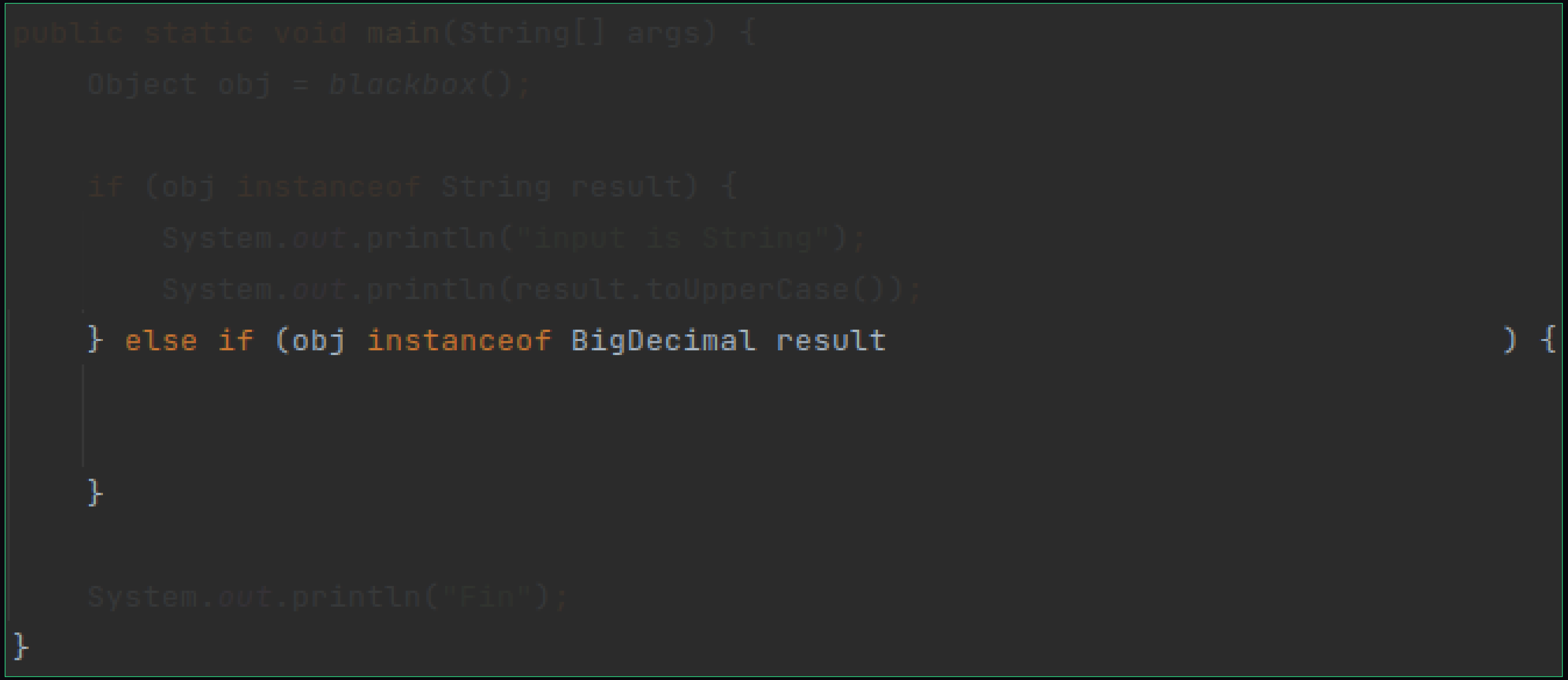
```
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof String result) {
        System.out.println("input is String");
        System.out.println(result.toUpperCase());



    }

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

JDK 16



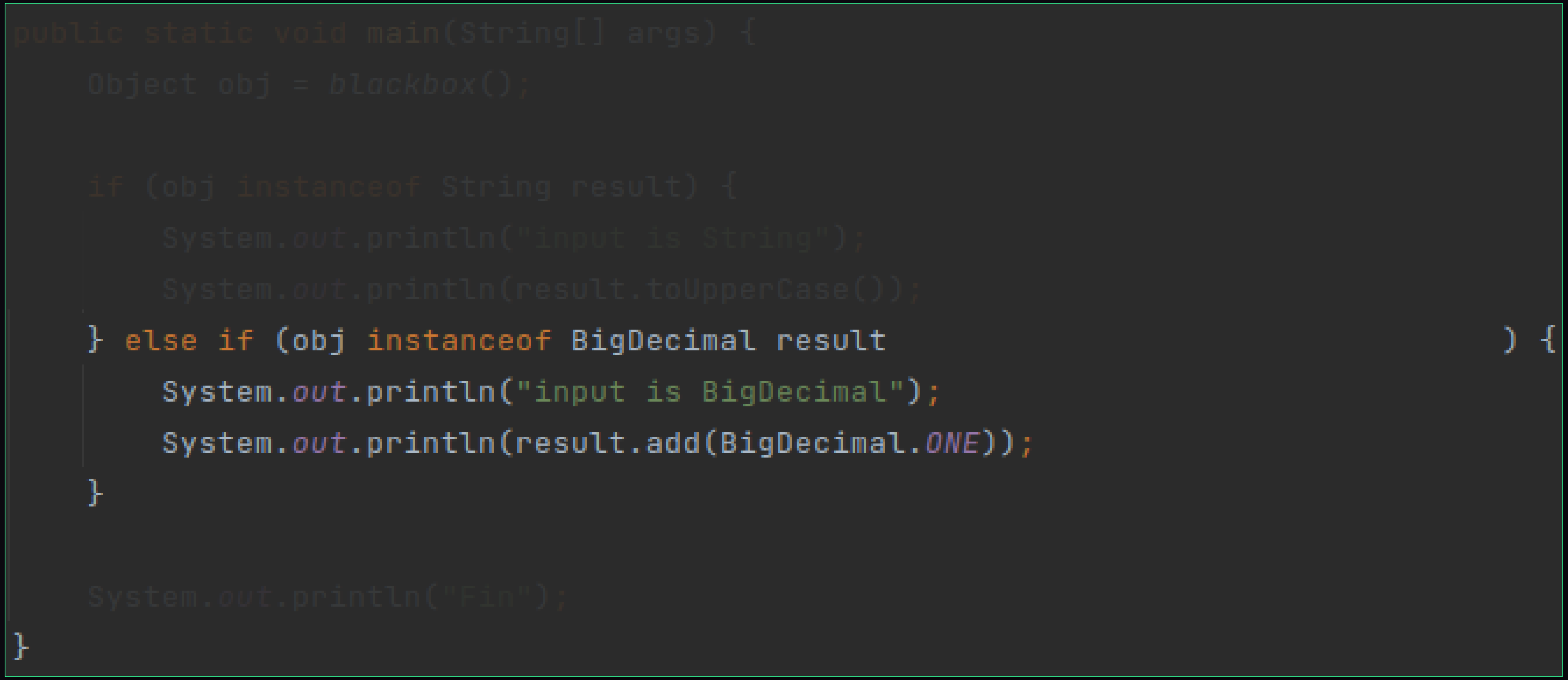
```java
public static void main(String[] args) {
    Object obj = blackbox();

    if (obj instanceof String result) {
        System.out.println("input is String");
        System.out.println(result.toUpperCase());



    }

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

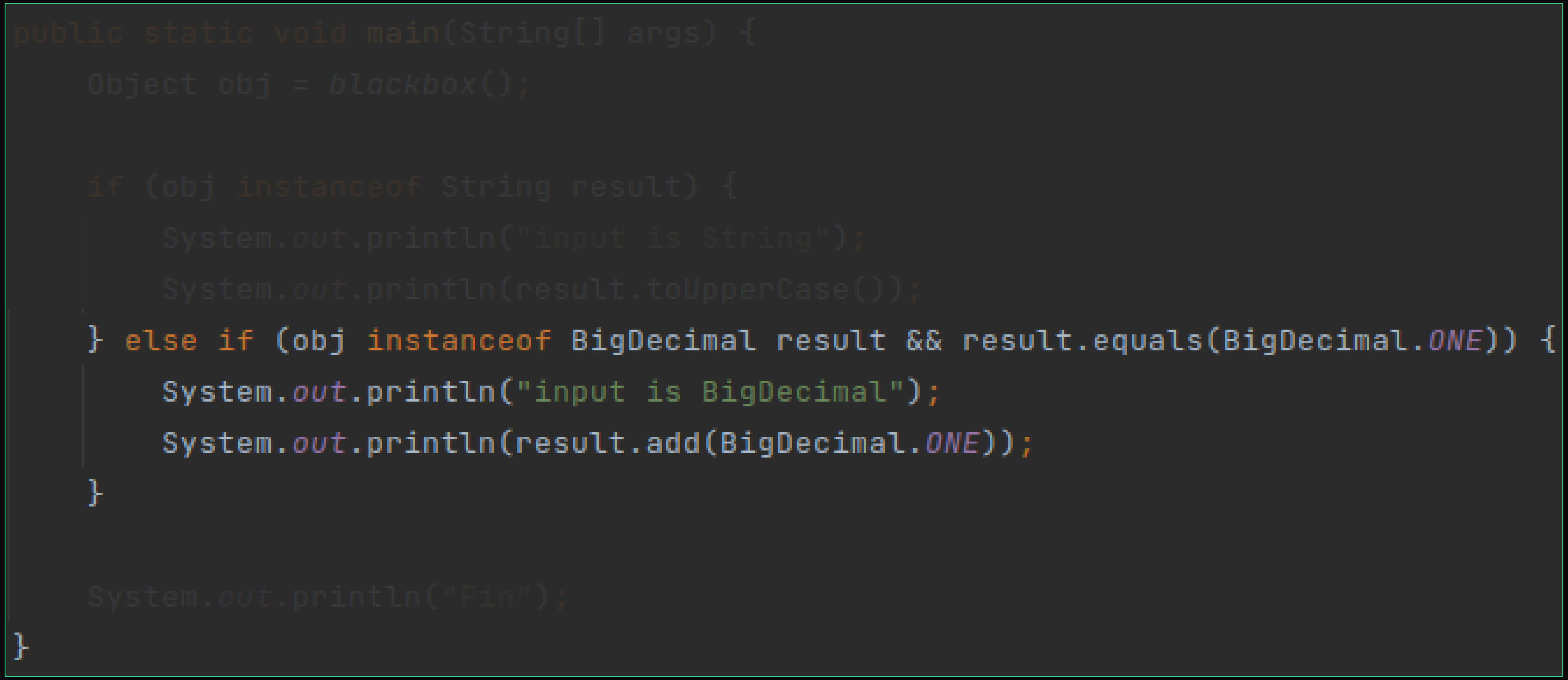
JDK 16



```
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());
    } else if (obj instanceof BigDecimal result                              ) {

    }

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

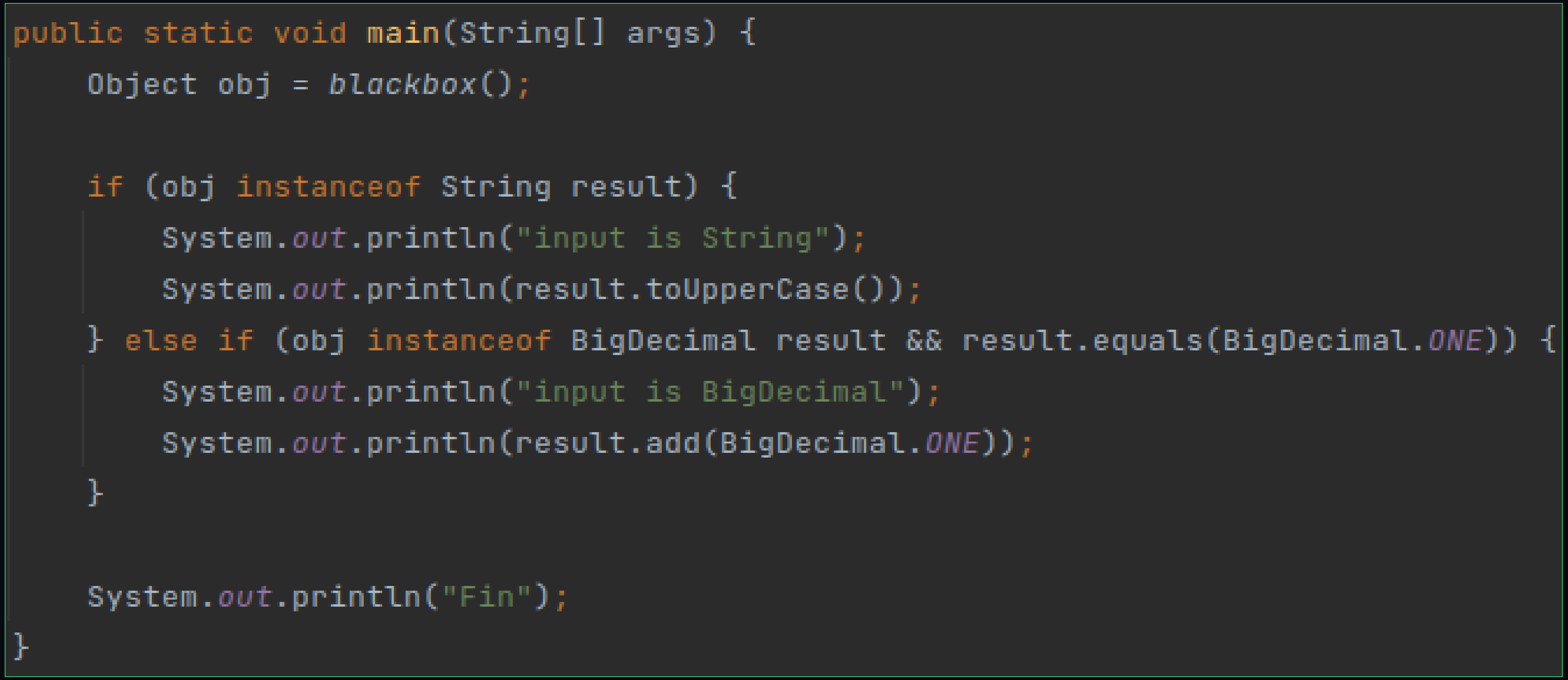
JDK 16


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

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



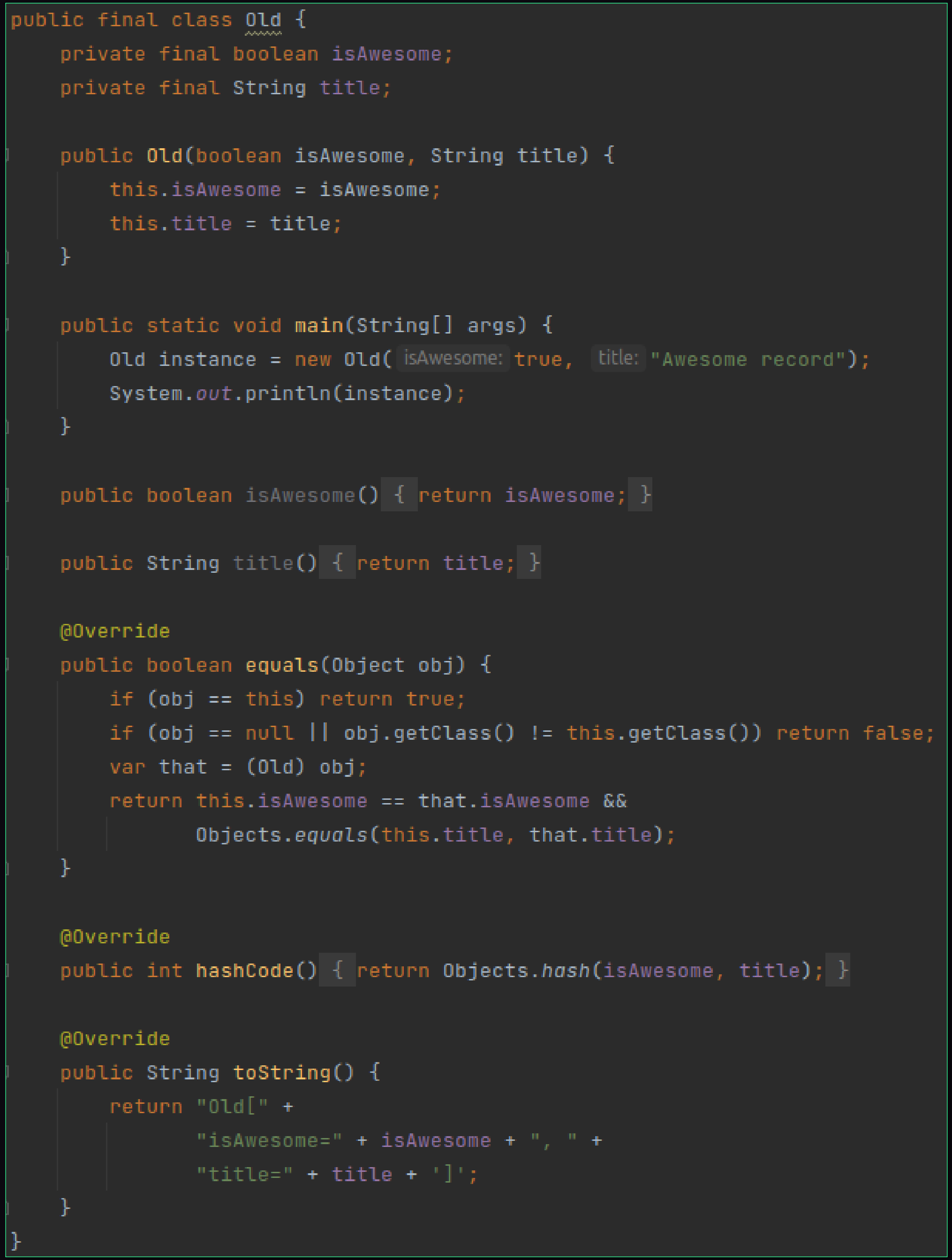
```java
public static void main(String[] args) {
    Object obj = blackbox();

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

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

JDK 16

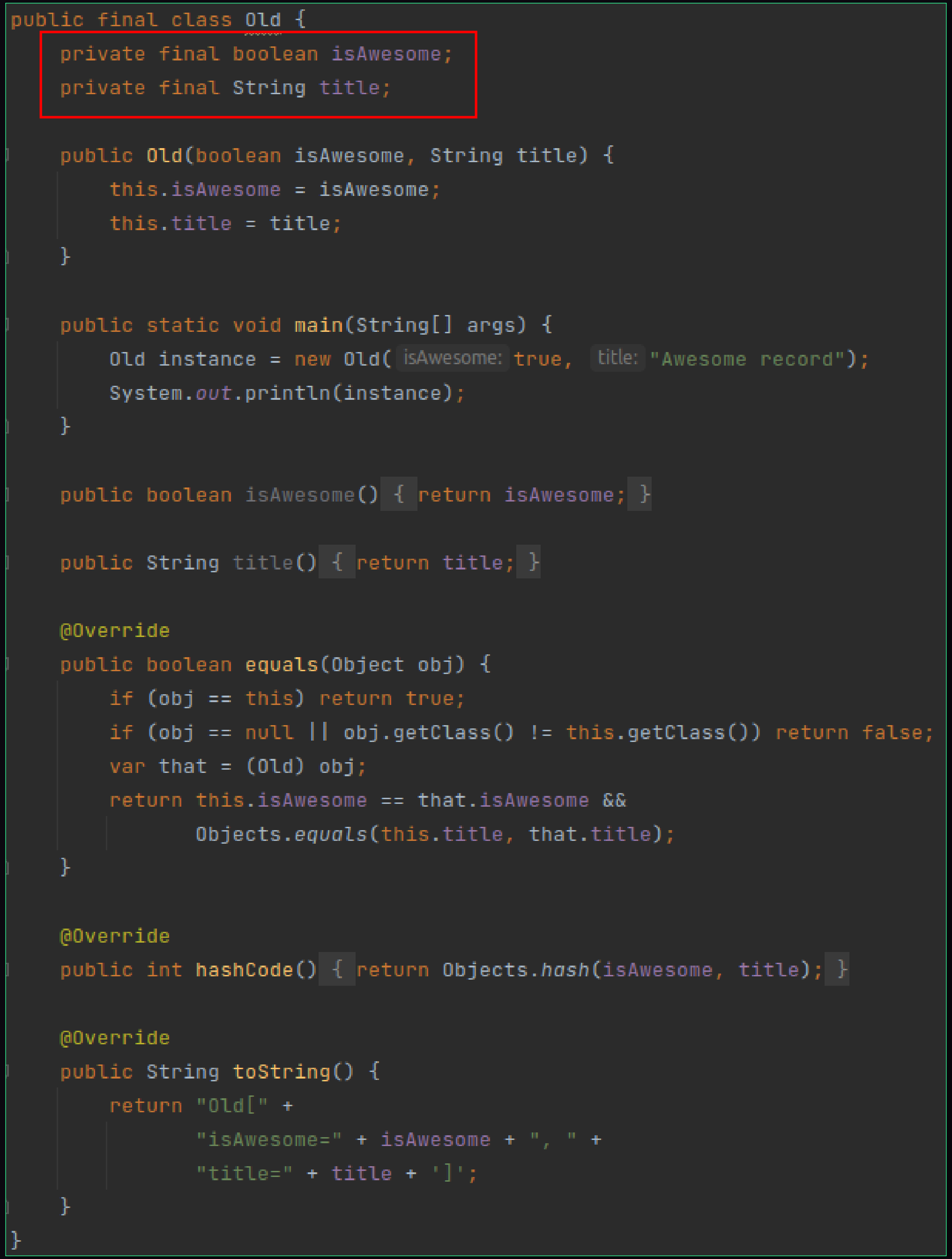


```java
public static void main(String[] args) {
    Object obj = blackbox();

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

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

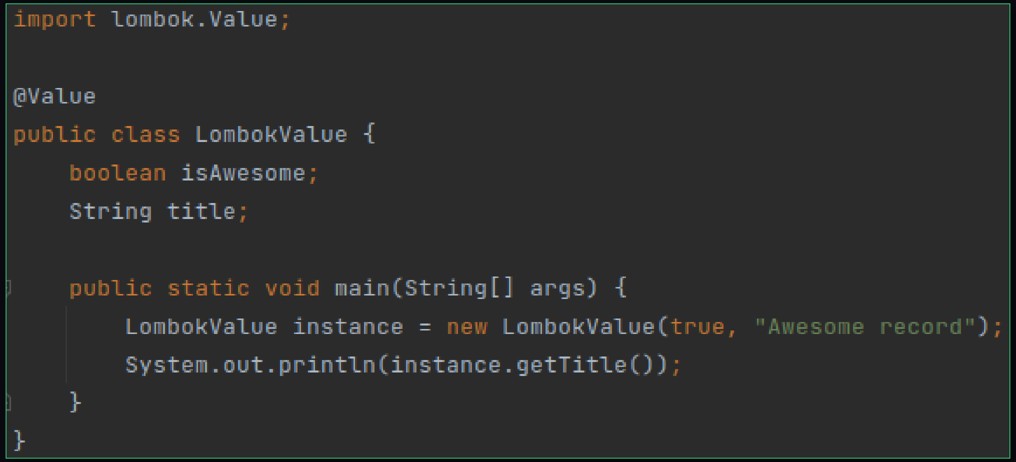
Data class



```java
public final class Old {
    private final boolean isAwesome;
    private final String title;

    public Old(boolean isAwesome, String title) {
        this.isAwesome = isAwesome;
        this.title = title;
    }

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

    public boolean isAwesome() { return isAwesome; }

    public String title() { return title; }

    @Override
    public boolean equals(Object obj) {
        if (obj == this) return true;
        if (obj == null || obj.getClass() != this.getClass()) return false;
        var that = (Old) obj;
        return this.isAwesome == that.isAwesome &&
                Objects.equals(this.title, that.title);
    }

    @Override
    public int hashCode() { return Objects.hash(isAwesome, title); }

    @Override
    public String toString() {
        return "Old[" +
                "isAwesome=" + isAwesome + ", " +
                "title=" + title + ']';
    }
}
```



```java
public final class Old {
    private final boolean isAwesome;
    private final String title;

    public Old(boolean isAwesome, String title) {
        this.isAwesome = isAwesome;
        this.title = title;
    }

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

    public boolean isAwesome() { return isAwesome; }

    public String title() { return title; }

    @Override
    public boolean equals(Object obj) {
        if (obj == this) return true;
        if (obj == null || obj.getClass() != this.getClass()) return false;
        var that = (Old) obj;
        return this.isAwesome == that.isAwesome &&
                Objects.equals(this.title, that.title);
    }

    @Override
    public int hashCode() { return Objects.hash(isAwesome, title); }

    @Override
    public String toString() {
        return "Old[" +
                "isAwesome=" + isAwesome + ", " +
                "title=" + title + ']';
    }
}
```



```java
import lombok.Value;

@Value
public class LombokValue {
    boolean isAwesome;
    String title;

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

Zbliżamy się do rekordu

Record

Image: freepik.com

```java
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?

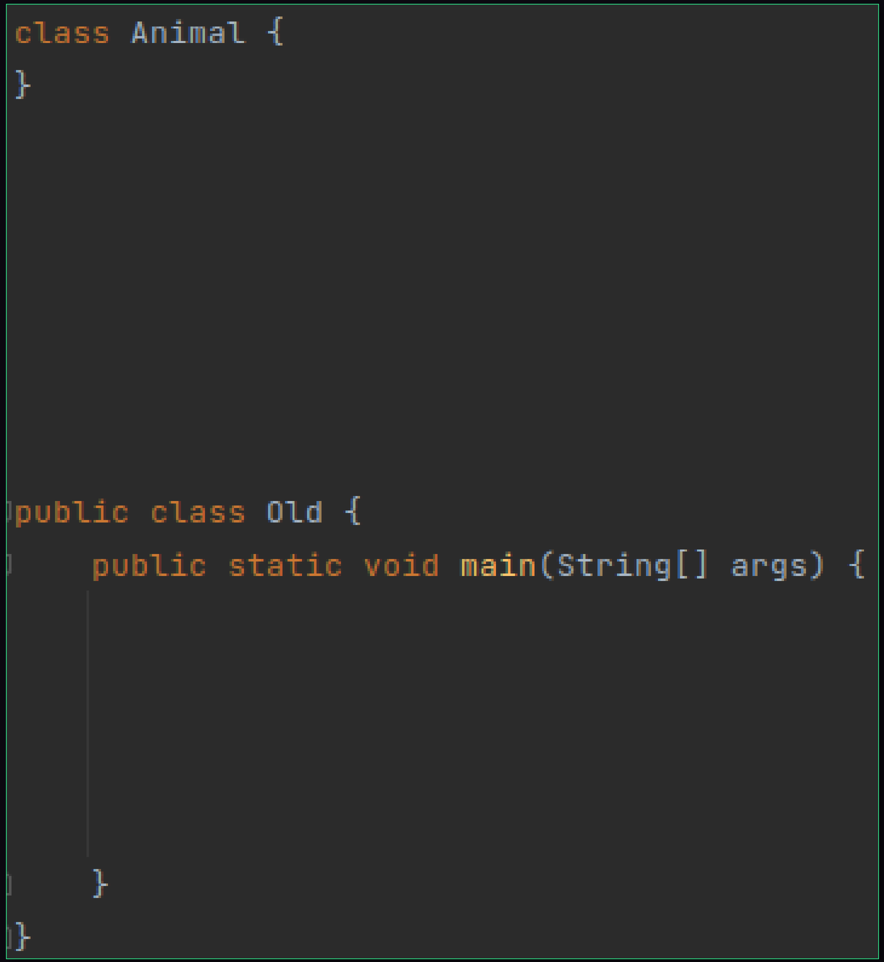
Dziedziczenie

final

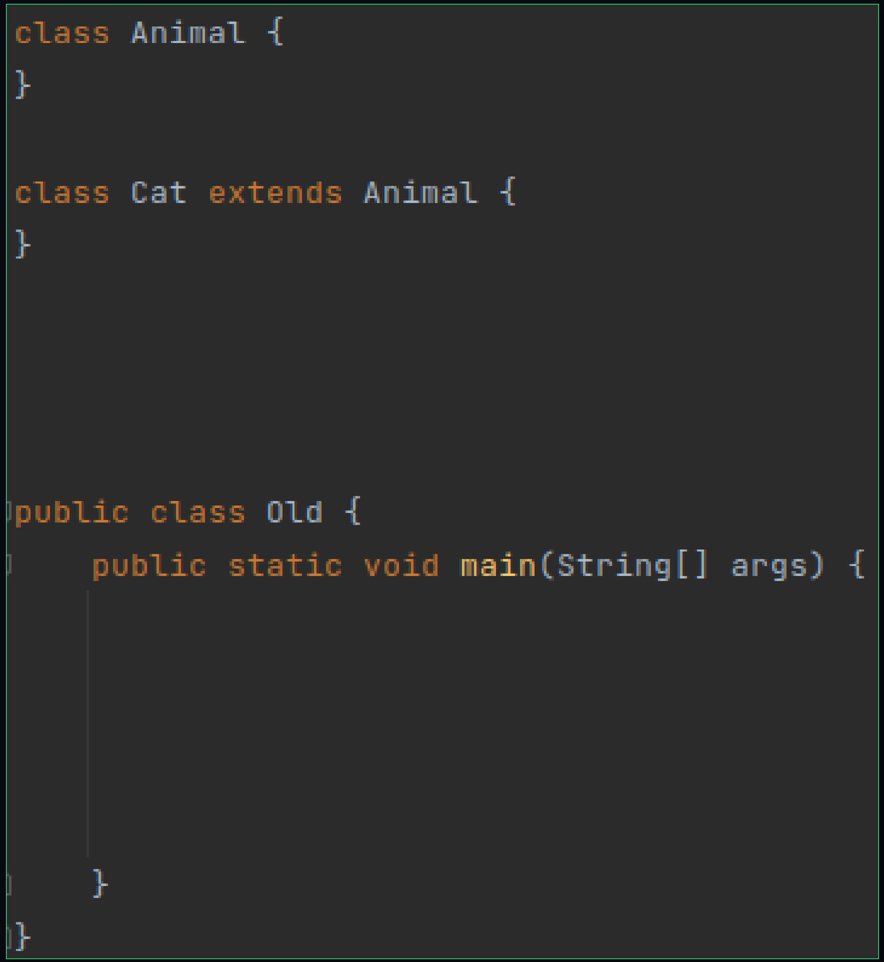
Klasy sealed

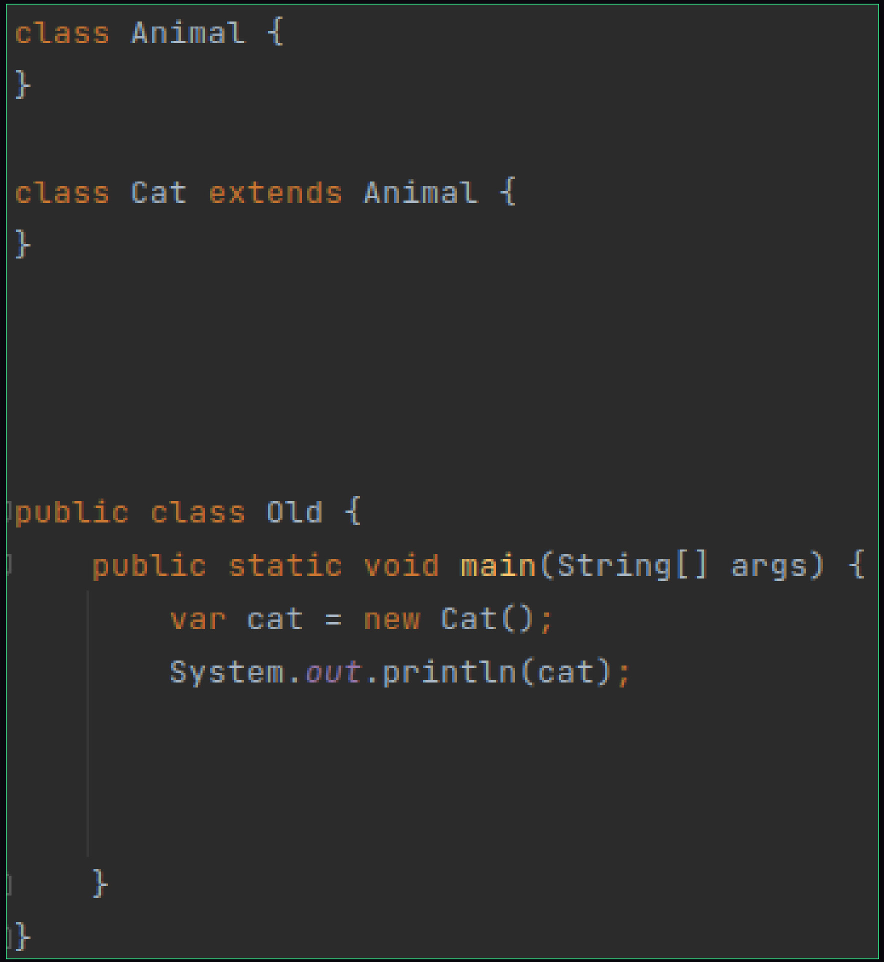
Image: freepik.com

Antykoncepcja dla Javy

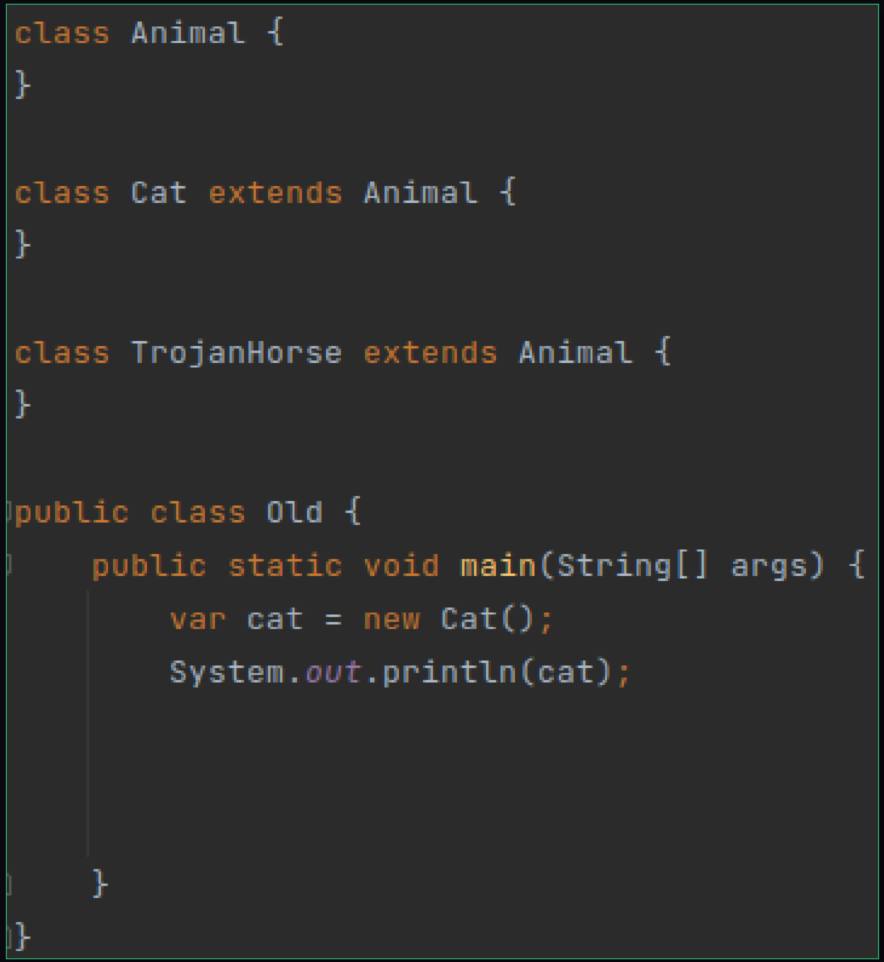


```java
class Animal {
}






public class Old {
    public static void main(String[] args) {




    }
}
```

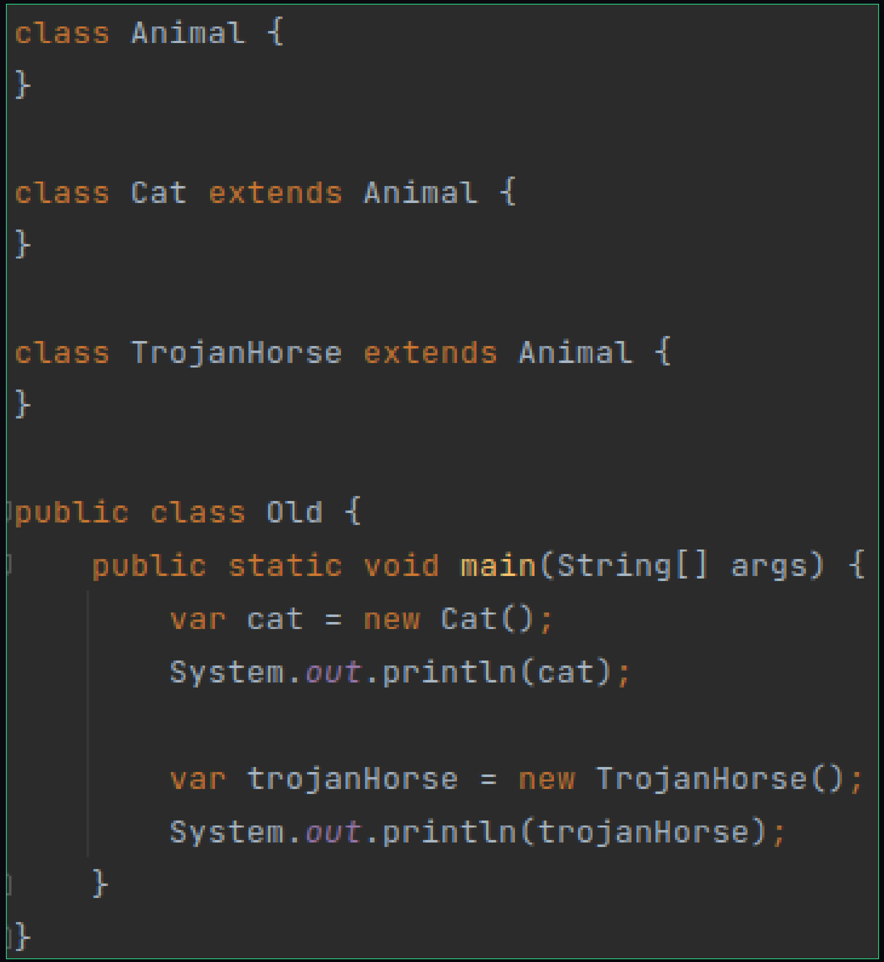

```java
class Animal {
}

class Cat extends Animal {
}




public class Old {
    public static void main(String[] args) {





    }
}
```



```java
class Animal {
}

class Cat extends Animal {
}



public class Old {
    public static void main(String[] args) {
        var cat = new Cat();
        System.out.println(cat);


    }
}
```



```java
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);



    }
}
```



```java
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);
    }
}
```

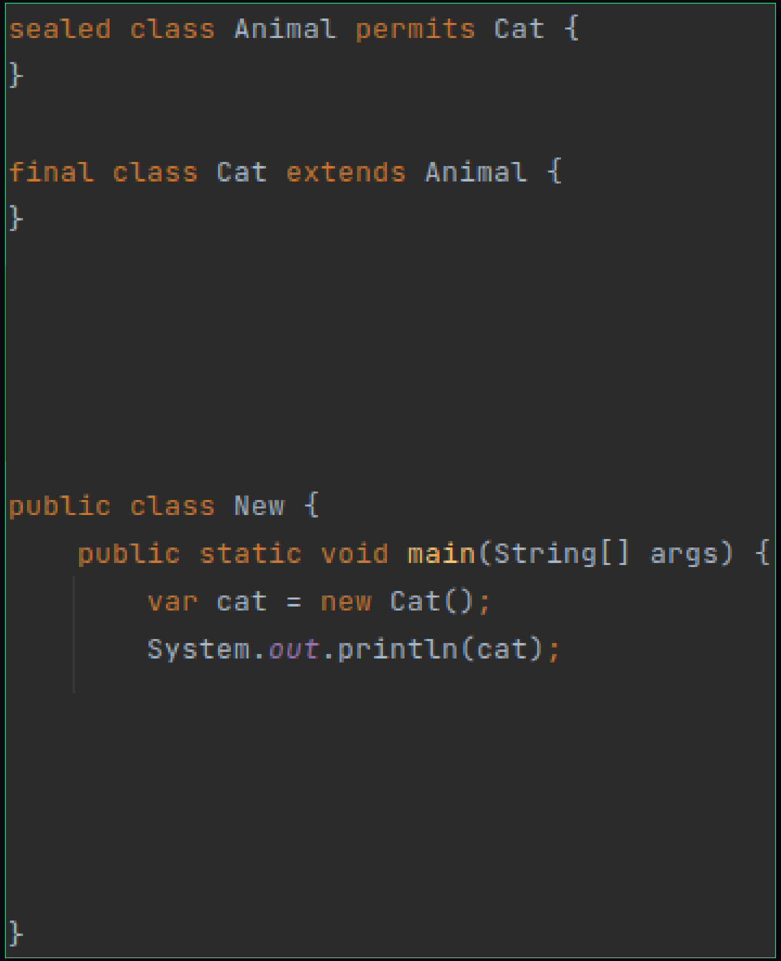
JDK 17

```java
sealed class Animal permits Cat {
}
```

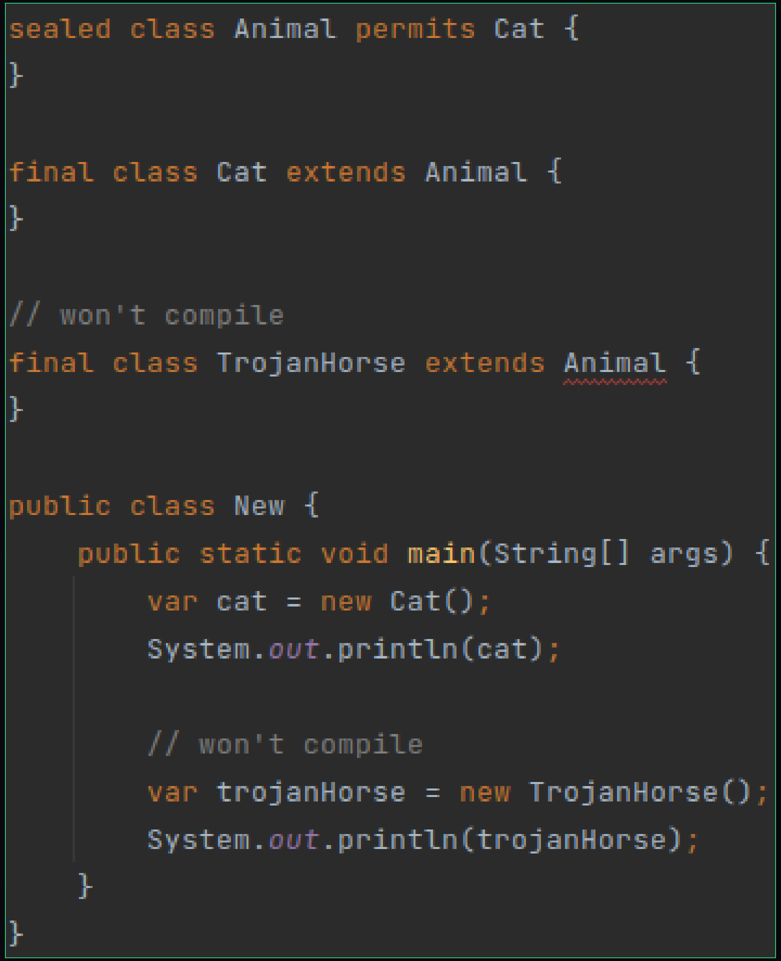
JDK 17

```
sealed class Animal permits Cat {
}

final class Cat extends Animal {
}
```

JDK 17



```java
sealed class Animal permits Cat {
}

final class Cat extends Animal {
}





public class New {
    public static void main(String[] args) {
        var cat = new Cat();
        System.out.println(cat);




}
```


```java
sealed class Animal permits Cat {
}

final class Cat extends Animal {
}

// won't compile
final class TrojanHorse extends Animal {
}

public class New {
    public static void main(String[] args) {
        var cat = new Cat();
        System.out.println(cat);

        // won't compile
        var trojanHorse = new TrojanHorse();
        System.out.println(trojanHorse);
    }
}
```

sealed non-sealed permits

JDK 17

```java
sealed class Animal permits Cat {}

non-sealed class Cat extends Animal {}

class OutdoorCat extends Cat {}
```

```
sealed class Animal permits Cat {}

non-sealed class Cat extends Animal {}

class OutdoorCat extends Cat {}

final class DomesticCat extends Cat {}
```

Jakiej muzyki słuchają klasy?

final class

sealed class

non-sealed class

class

```
class Rodzic { }
```

class

```
class Rodzic { }

class NieplanownyPotomek extends Rodzic {}
```

Finalizers

01 WINS
00 WINS
REPTILE
KUNG LAO
FINISH HIM !!

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

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

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

    if (obj instanceof Rectangle rect) {










    }

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

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

    if (obj instanceof Rectangle rect) {









        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
    }

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

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

    if (obj instanceof Rectangle rect) {
        var topLeft = rect.topLeft();
        var bottomRight = rect.bottomRight();







        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
    }

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

```java
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();





        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
    }

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

```java
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;




        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
    }

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

```java
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");
}
```

```java
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

Co powiecie na 4 linie?

Record patterns

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

    if (obj instanceof Rectangle                                              ) {


        System.out.println("Width: " + width);
        System.out.println("Height: " + height);
    }

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

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

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

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

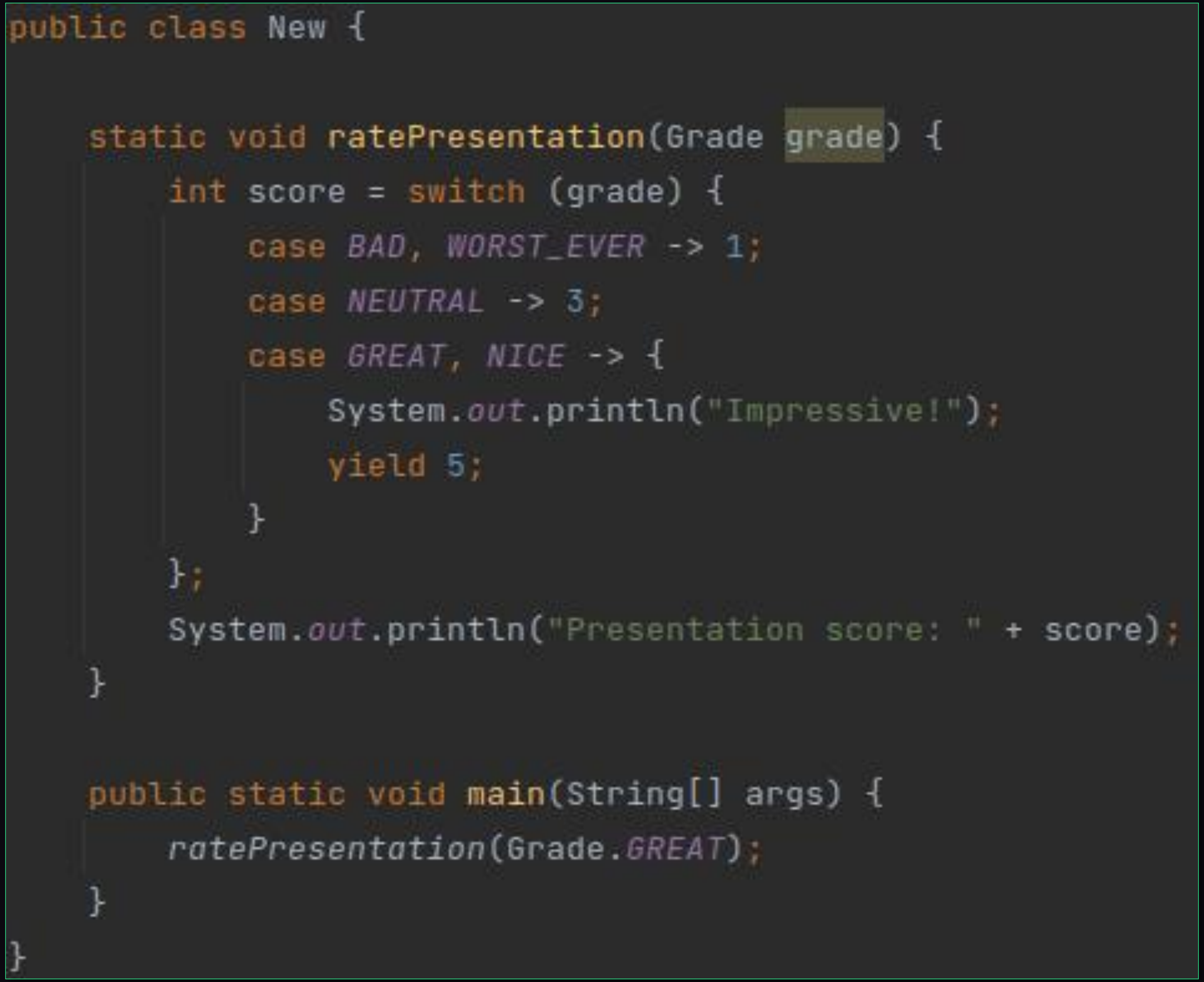
```
public static void main(String[] args) {
    Object obj = blackbox();

    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");
}
```

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

    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



```java
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);
    }
}
```

```
public class Old {

    static void ratePresentation(Grade grade) {
        if (grade == null) {
            System.out.println("Unfortunately grade is null");
            return;
        }
        int score = switch (grade) {
            case 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);
    }
}
```

```
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));




    }
}
```

```
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());



    }
}
```

JDK 17-19

```java
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!");
    }
}
```

```
static void testTriangle(Shape s) {
    switch (s) {
        case null:
            break;








         default:
            System.out.println("A shape, possibly a small triangle");
    }
}
```

```
static void testTriangle(Shape s) {
    switch (s) {
        case null:
            break;
        case Triangle t:




         default:
            System.out.println("A shape, possibly a small triangle");
    }
}
```

```
static void testTriangle(Shape s) {
    switch (s) {
        case null:
            break;
        case Triangle t:
            if (t.calculateArea() > 100) {


            }

         default:
            System.out.println("A shape, possibly a small triangle");
    }
}
```

```
static void testTriangle(Shape s) {
    switch (s) {
        case null:
            break;
        case Triangle t:
            if (t.calculateArea() > 100) {
                System.out.println("Large triangle");
                break;
            }

         default:
            System.out.println("A shape, possibly a small triangle");
    }
}
```

```
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

JDK 17-19

```
static void error(Object o) {
   switch (o) {





   }
}
```

```
static void error(Object o) {
  switch (o) {




    default -> {}
  }
}
```

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


        default -> {}
    }
}
```

```
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
                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 -> {}
    }
}
```

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

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

JDK 17-19

```
static void checkAnimal(Animal animal) {
    switch (animal) {



    }
}
```

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


   }
}
```

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

   }
}
```

```
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");
   }
}
```

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

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

```java
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");
   }
}
```

Tradycyjny wątek

Tradycyjny wątek

# wątków w Javie

=

# wątków systemowych

Tradycyjny wątek

# wątków w Javie

=

# wątków systemowych

<=

MAX # wątków systemowych

Virtual threads

Wirtualny wątek

# wątków w Javie

>

MAX # wątków systemowych

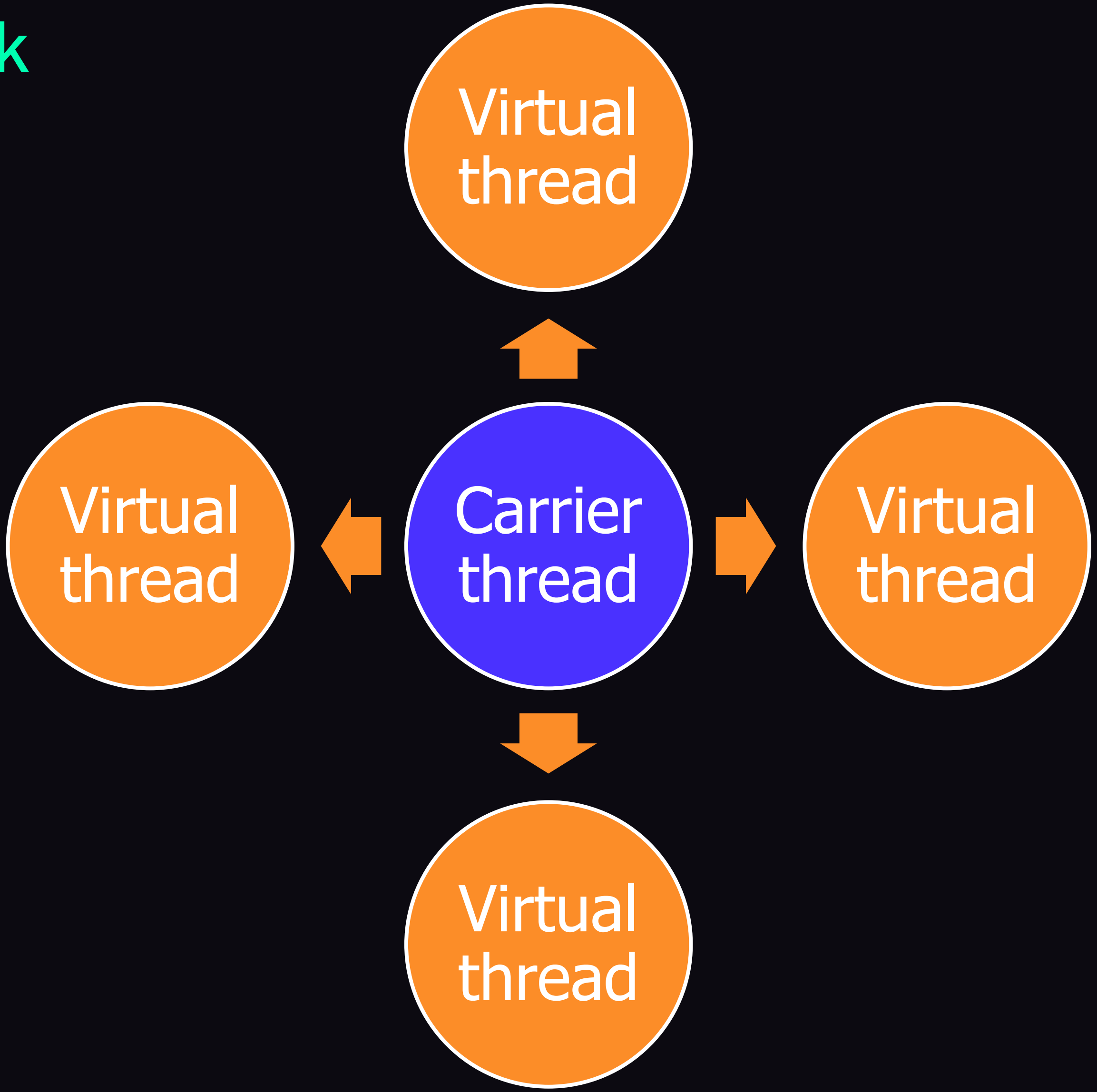
Wirtualny wątek

# wątków w Javie

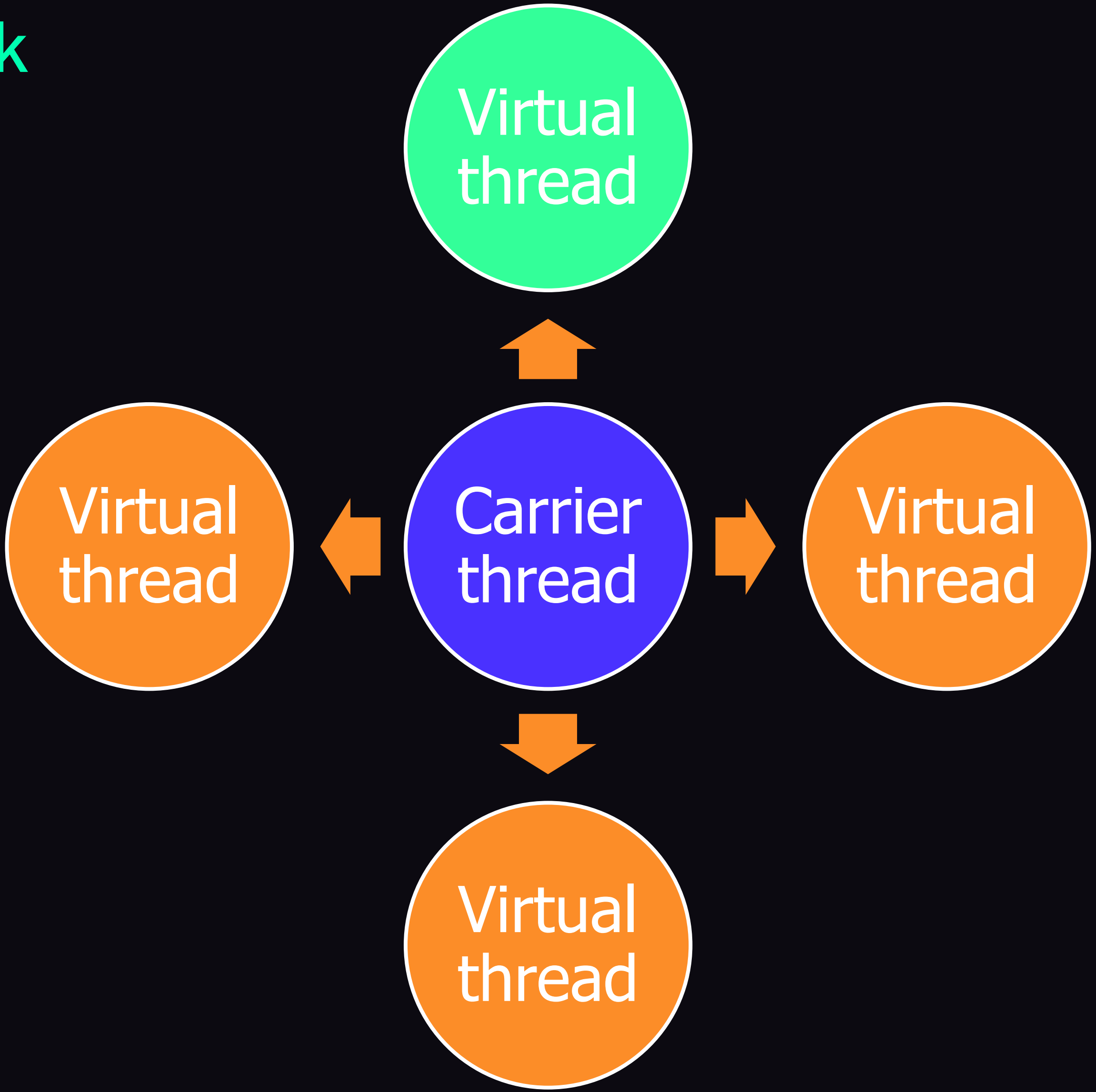
>>

MAX # wątków systemowych

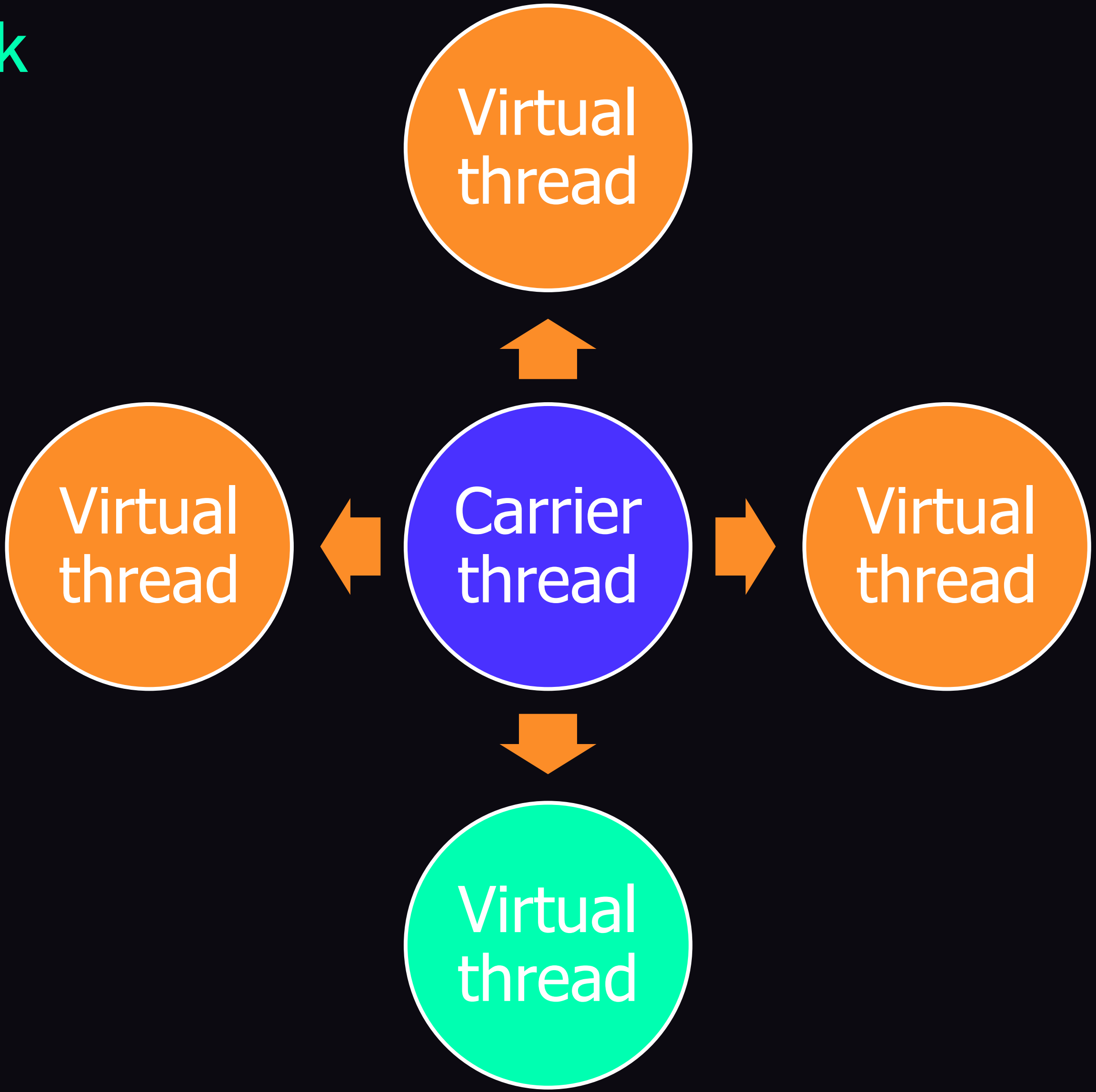
Wirtualny wątek



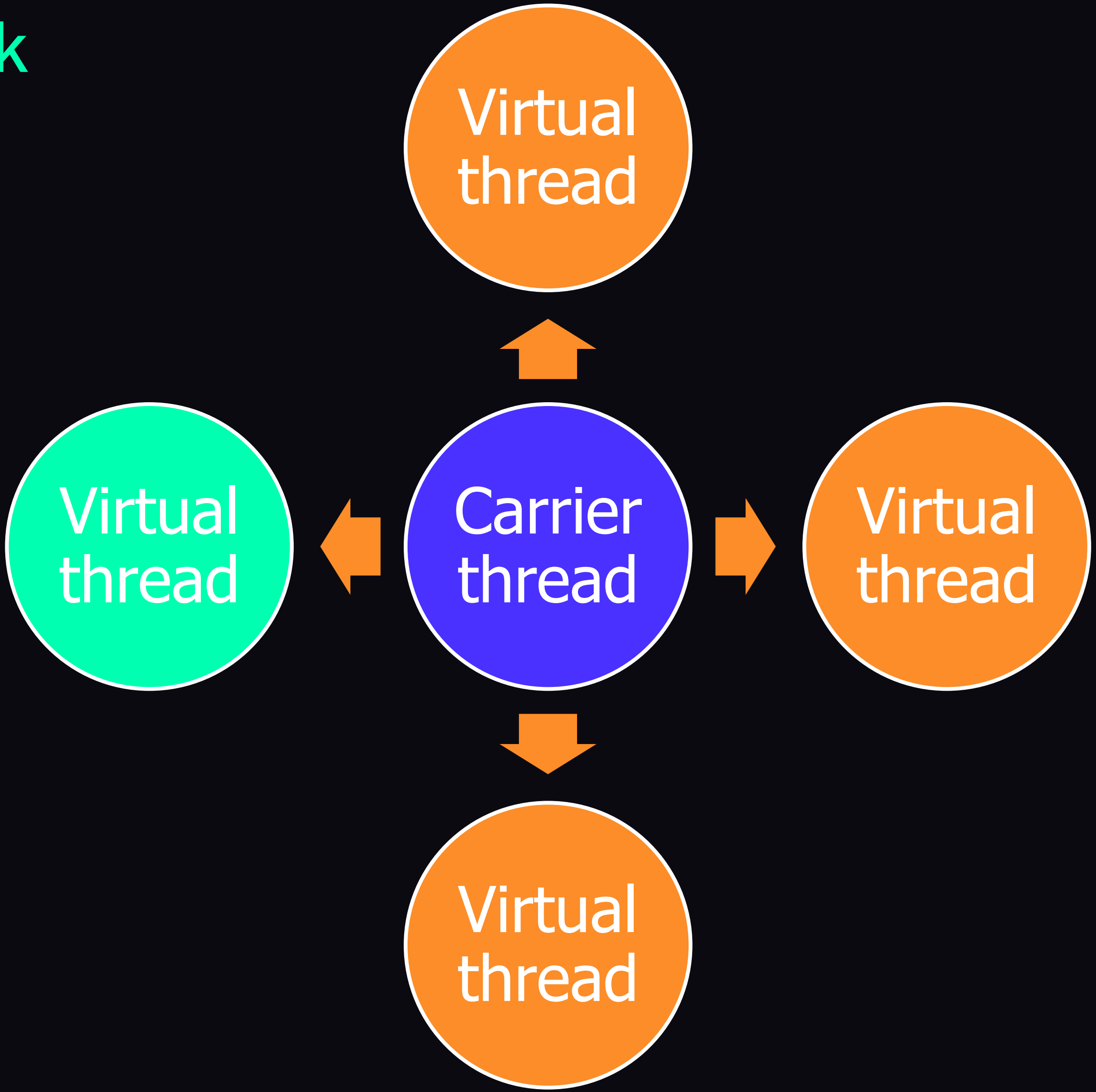
Wirtualny wątek



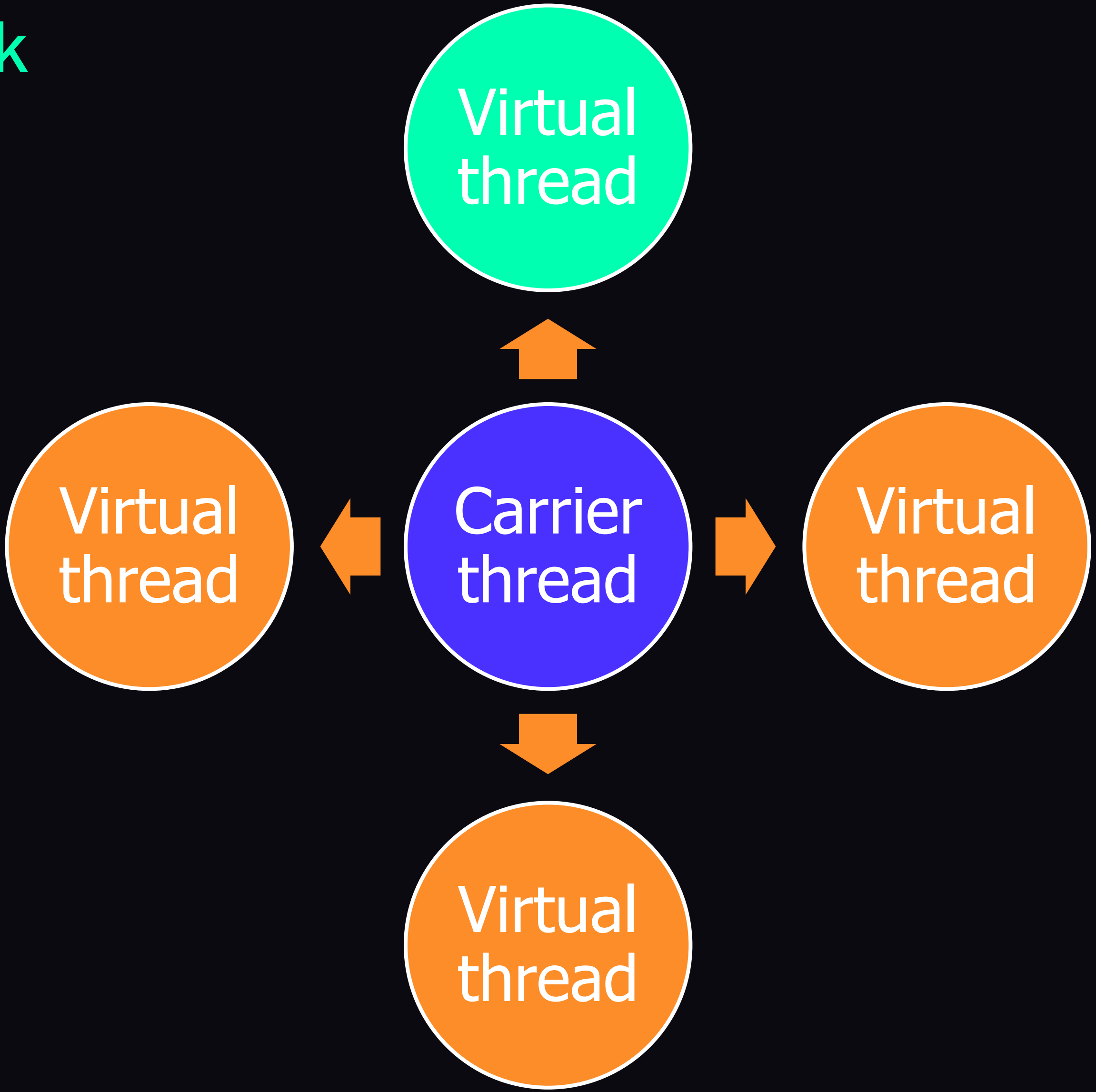
Wirtualny wątek



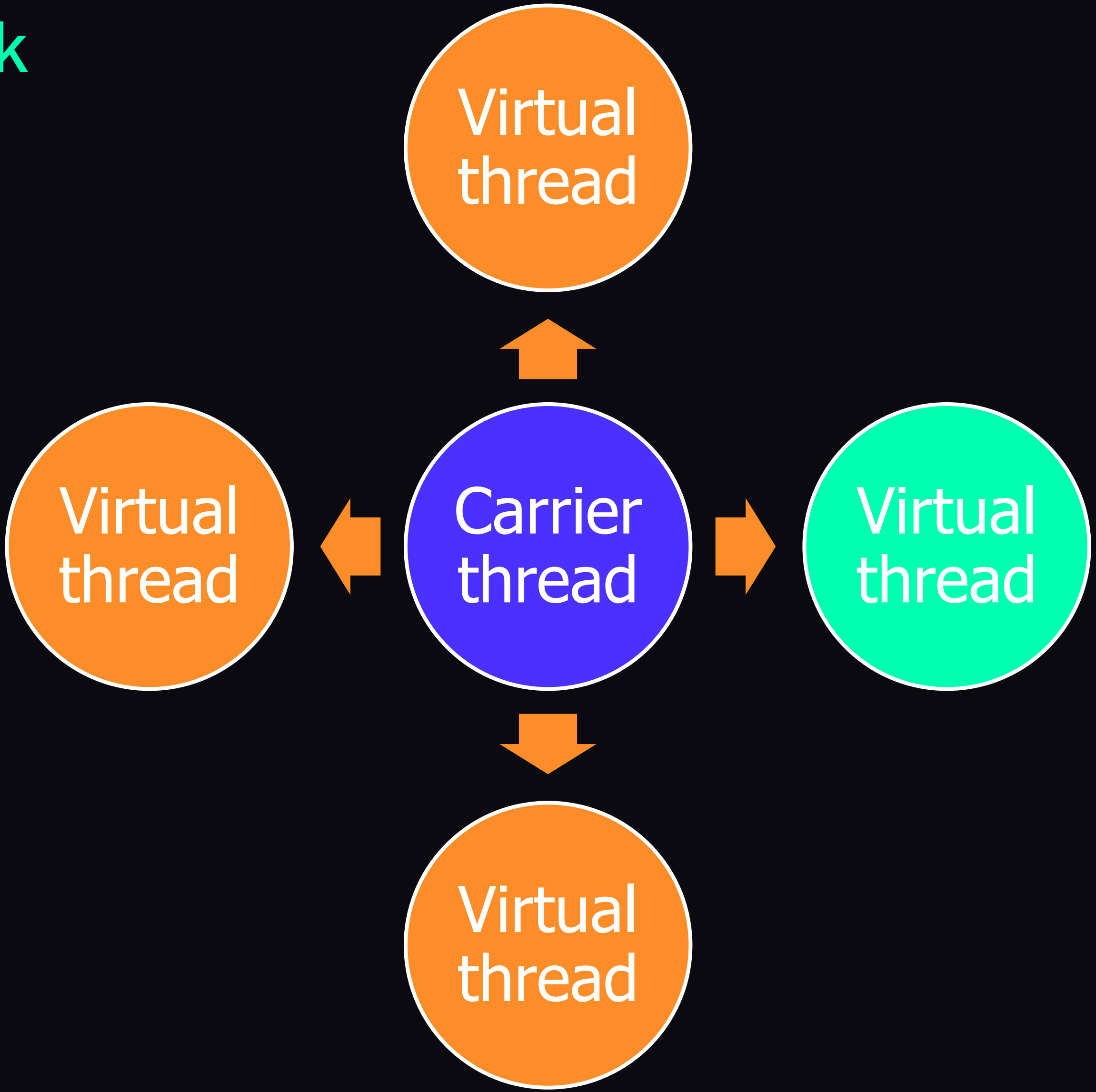
Wirtualny wątek



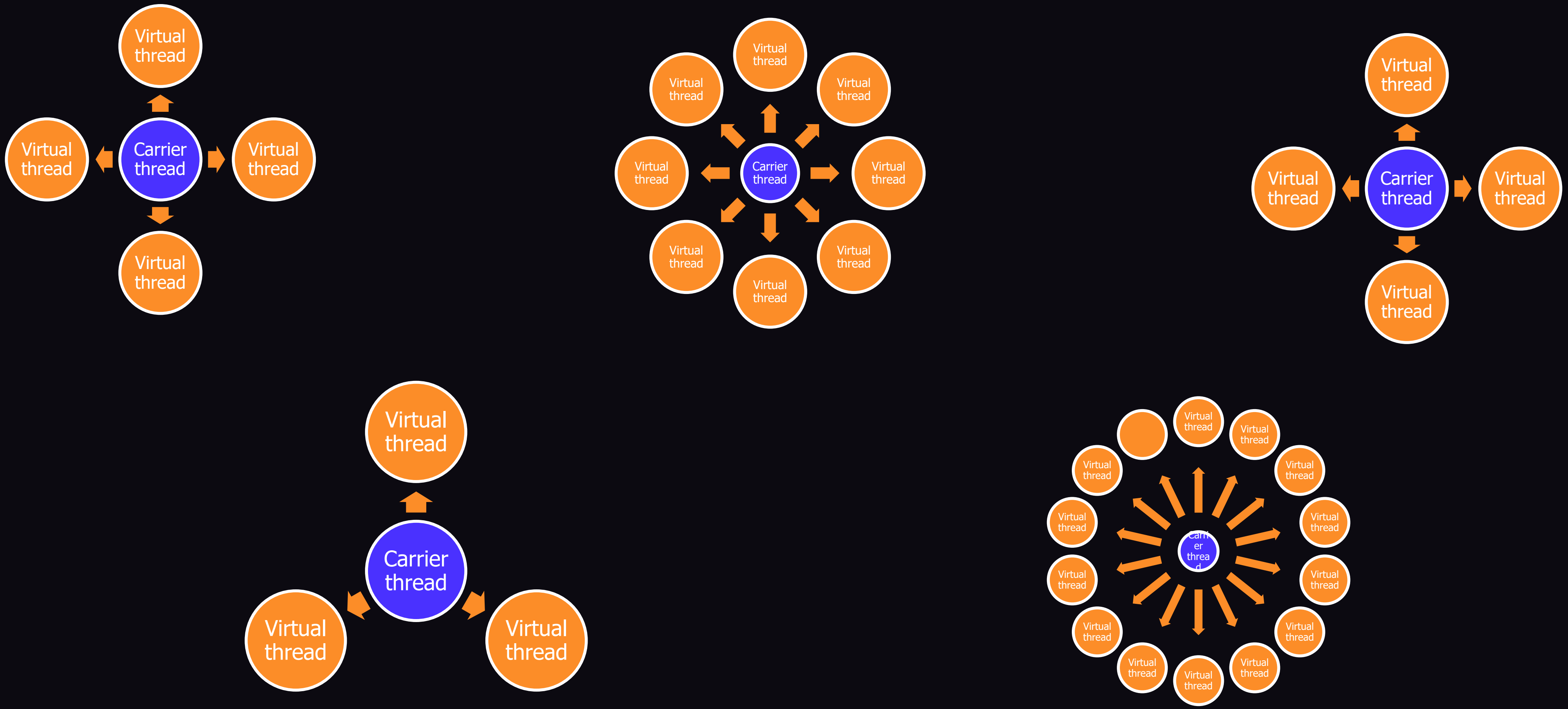
Wirtualny wątek



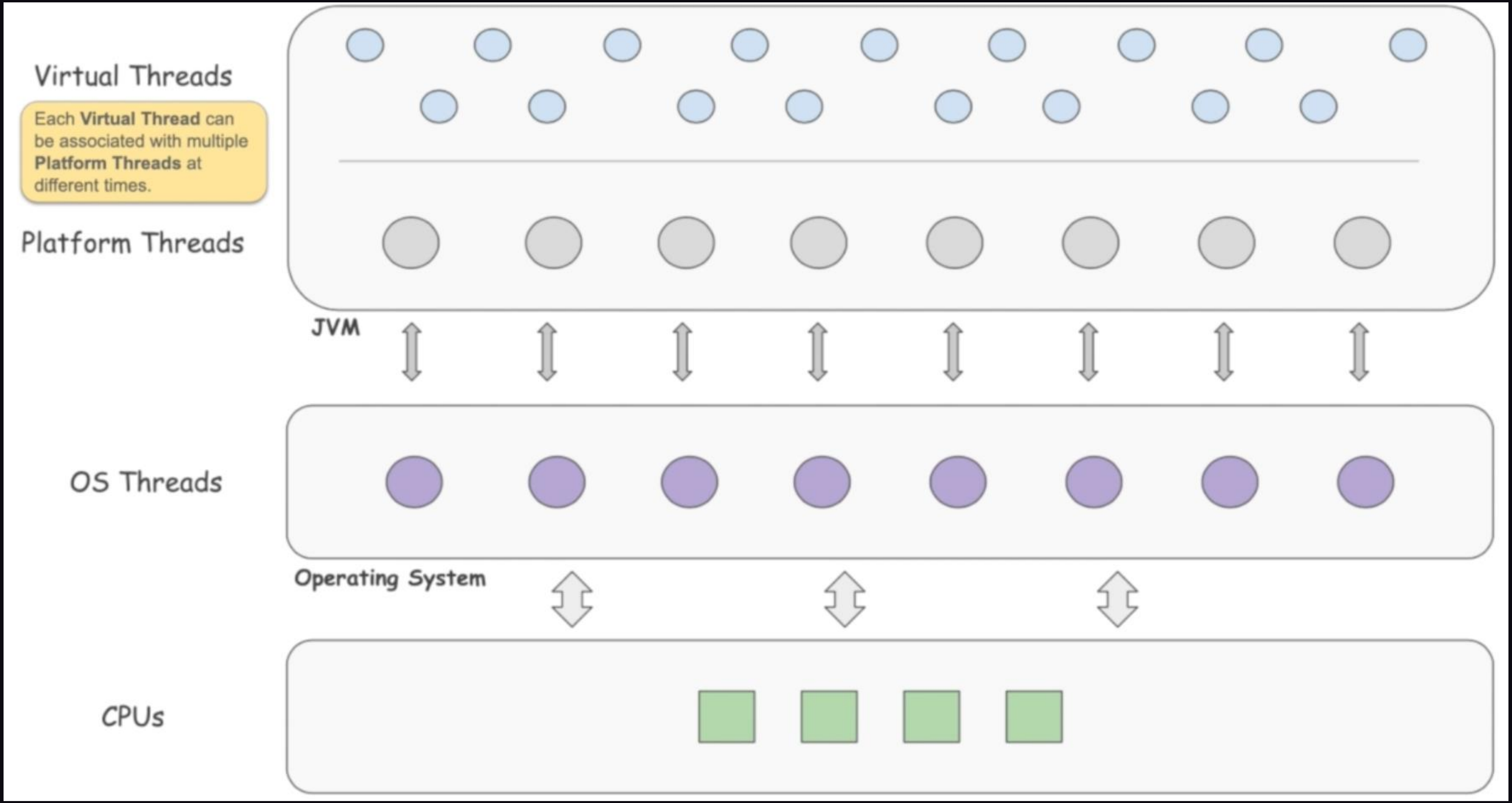
Wirtualny wątek



Wirtualny wątek



Wirtualny wątek


Virtual Threads
Each **Virtual Thread** can be associated with multiple **Platform Threads** at different times.
Platform Threads
JVM
OS Threads
Operating System
CPUs

Wirtualny wątek

Jakie operacje blokują 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 w pulą wątków

! Można zapchać „pulę” używając **synchronized** (nie sprawdzałem)

! Brak priorytetów

! Nowinka (nadal preview)

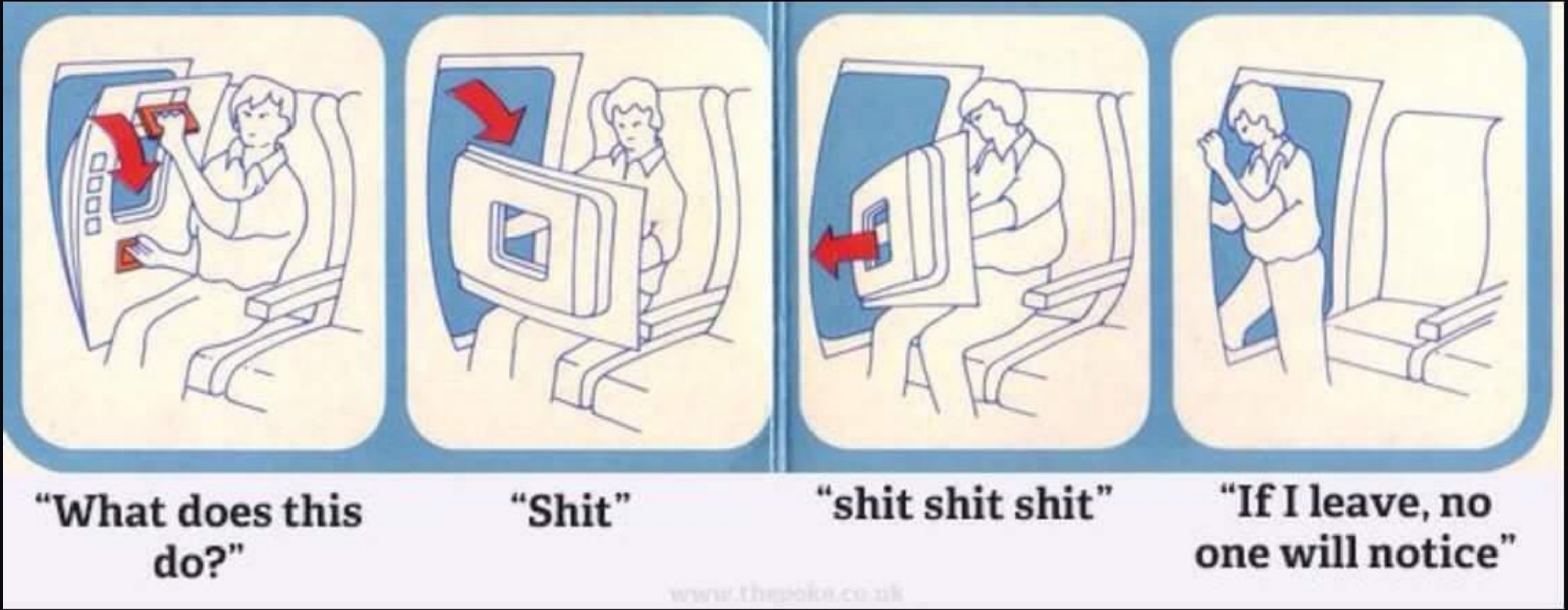
To wszystkie zjavy na dziś

ZWILLING
J.A. HENCKELS
GERMANY

Pracujcie nad długiem technologicznym

Niech biznes wie,
że to ważne

Testujcie zmiany


"What does this do?"
"Shit"
"shit shit shit"
"If I leave, no one will notice"

mbaranowski.pl

ANY QUESTIONS?
memecrunch.com