
DIASOFT
всё по-настоящему

VI. Исключения

4. Antipatterns

Замена проверки исключением

Замена простой проверки

DIASOFT
всё по-настоящему

```
public class TestReplacementDemo {

   public static void main(String[] args) {

      String[] names = { "Harry Hacker", "Tonny Tester", "Eve Engineer" };

      long now = System.currentTimeMillis();

      for (int i = 0; i < 100000000; i++) {
         if (3 < names.length) {

            System.out.println(names[3]);
         }
      }

      System.out.println("Time using length check: " + (System.currentTimeMillis() - now) + " ms");

      now = System.currentTimeMillis();

      for (int i = 0; i < 100000000; i++) {
         try {
            System.out.println(names[3]);
         } catch (ArrayIndexOutOfBoundsException e) {

         }
      }
      System.out.println("Time using exceptions: " + (System.currentTimeMillis() - now) + " ms");
   }
}
```

```
Time using length check: 141 ms
Time using exceptions: 232765 ms
```

Перехватывание и игнорирование

Перехватывание и игнорирование

DIASOFT
всё по-настоящему

```
public class CatchAndIgnore {

    public static void main(String[] args) {

        String[] names = { "I:\\FileIO\\lineFile.txt", "I:\\noSuchDir\\noSuchFile" };

        for (String name : names) {
            printFirstLine(name);
        }
    }

    private static void printFirstLine(String fileName) {

        try {
            BufferedReader br = new BufferedReader(new FileReader(fileName));
            System.out.print(br.readLine());
        }

        catch (IOException e) {
        }
    }
}
```

Деструктивная обёртка

```java
class MyException extends RuntimeException {

    public MyException(String message) {
        super(message);
    }

    public MyException(String message, Throwable cause) {
        super(message, cause);
    }
}
```

Деструктивная обёртка

DIASOFT
всё по-настоящему

```
public class DestructiveWrappingDemo {

    public static void main(String[] args) {

        String[] names = { "I:\\FileIO\\lineFile.txt", "I:\\noSuchDir\\noSuchFile" };

        for (String name : names) {
            try {
                writeLine(name, "Hello World!");
            } catch (IOException e) {
                throw new MyException("Hello: something bad has happened ..... ");
            }
        }
    }

    private static void writeLine(String fileName, String line)
            throws IOException {

        Writer wr = new FileWriter(fileName);
        wr.write(line);
        wr.flush();
    }
}
```

```
Exception in thread "main" antipatterns.MyException: Hello: something bad has happened .....
at antipatterns.DestructiveWrappingDemo.main(DestructiveWrappingDemo.java:18)
```

Недеструктивная обёртка

DIASOFT
всё по-настоящему

```
public class DestructiveWrappingDemo {

    public static void main(String[] args) {

        String[] names = { "I:\\FileIO\\lineFile.txt", "I:\\noSuchDir\\noSuchFile" };

        for (String name : names) {
            try {
                writeLine(name, "Hello World!");
            } catch (IOException e) {
                throw new MyException("Hello: something bad has happened ..... ", e);
            }
        }
    }

    private static void writeLine(String fileName, String line)
            throws IOException {

        Writer wr = new FileWriter(fileName);
        wr.write(line);
        wr.flush();
    }
}
```

```
Exception in thread "main" antipatterns.MyException: Hello: something bad has happened .....
at antipatterns.DestructiveWrappingDemo.main(DestructiveWrappingDemo.java:19)
Caused by: java.io.FileNotFoundException: I:\noSuchDir\noSuchFile (The system cannot find the path specified)
at java.io.FileOutputStream.open(Native Method)
at java.io.FileOutputStream.<init>(Unknown Source)
at java.io.FileOutputStream.<init>(Unknown Source)
at java.io.FileWriter.<init>(Unknown Source)
at antipatterns.DestructiveWrappingDemo.writeLine(DestructiveWrappingDemo.java:27)
at antipatterns.DestructiveWrappingDemo.main(DestructiveWrappingDemo.java:16)
```

Перехватывание и возвращение null

Возвращение null

DIASOFT
всё по-настоящему

```
public class CatchReturnNull {

    public static void main(String[] args) {

        String line = readFirstLine("I:\\noSuchDir\\noSuchFile");

        if (line == null) {
            System.out.println("File is empty");
        }
        else {
            System.out.println("First line in file: " + line);
        }
    }

    private static String readFirstLine(String fileName) {
        try {

            BufferedReader br = new BufferedReader(new FileReader(fileName));
            return br.readLine();
        }

        catch (IOException e) {
           return null;
        }
    }
}
```

```
File is empty
```

Возвращение null для неподдерживаемой операции

Возвращение null

DIASOFT
всё по-настоящему

```
public class UnsupOpReturnNull {

    public static void main(String[] args) {

        SingletonStringMap map= new SingletonStringMap("firstName", "Alice");
        System.out.println(map);
        map.put("lastName", "Coder");
        System.out.println(map.get("lastName").trim());
    }
}

class SingletonStringMap {

    private final String k;
    private final String v;

    SingletonStringMap(String key, String value) {
        k = key;
        v = value;
    }

    public int size() {return 1;}
    public boolean isEmpty() {return false;}
    public boolean containsKey(Object key) {return eq(key, k);}
    public boolean containsValue(Object value) {return eq(value, v);}
    public String get(String key) {return (eq(key, k) ? v : null);}
    public String put(String key, String value) {return null;}
    public String toString() {return "[" + k  + "," + v +"]";}

    private static boolean eq(Object o1, Object o2) {
        return (o1==null ? o2==null : o1.equals(o2));
    }
}
```

Возвращение null

```
[firstName,Alice]
Exception in thread "main" java.lang.NullPointerException
at antipatterns.UnsupOpReturnNull.main(UnsupOpReturnNull.java:10)
```

Логирование и перебрасывание

Логирование и перебрасывание

DIASOFT
всё по-настоящему

```
class SomeClass {

    private static Logger logger = Logger.getLogger(SomeClass.class);

    public void methodA() throws IOException {
        try {
            throw new IOException();
        } catch (IOException e) {
            logger.error("Error!", e);
            throw e;
        }
    }

    public void methodB() throws IOException {
        try {
            methodA();
        } catch (IOException e) {
            logger.error("Error!", e);
            throw e;
        }
    }

    public void methodC() throws IOException {
        try {
            methodB();
        } catch (IOException e) {
            logger.error("Error!", e);
            System.out.println("Handle Exception here.....");
        }
    }
}
```

Логирование и перебрасывание

DIASOFT всё по-настоящему

```
0 [main] ERROR antipatterns.SomeClass  - Error!
java.io.IOException
at antipatterns.SomeClass.methodA(LogAndRethrowDemo.java:23)
at antipatterns.SomeClass.methodB(LogAndRethrowDemo.java:32)
at antipatterns.SomeClass.methodC(LogAndRethrowDemo.java:41)
at antipatterns.LogAndRethrowDemo.main(LogAndRethrowDemo.java:13)
0 [main] ERROR antipatterns.SomeClass  - Error!
java.io.IOException
at antipatterns.SomeClass.methodA(LogAndRethrowDemo.java:23)
at antipatterns.SomeClass.methodB(LogAndRethrowDemo.java:32)
at antipatterns.SomeClass.methodC(LogAndRethrowDemo.java:41)
at antipatterns.LogAndRethrowDemo.main(LogAndRethrowDemo.java:13)
0 [main] ERROR antipatterns.SomeClass  - Error!
java.io.IOException
at antipatterns.SomeClass.methodA(LogAndRethrowDemo.java:23)
at antipatterns.SomeClass.methodB(LogAndRethrowDemo.java:32)
at antipatterns.SomeClass.methodC(LogAndRethrowDemo.java:41)
at antipatterns.LogAndRethrowDemo.main(LogAndRethrowDemo.java:13)
Handle Exception here.....
```

Спасибо

Россия, 127018,
Москва, ул. Полковая 3, стр. 14
Тел.: +7(495) 780 7575, 789 9339
Факс: +7(495) 780 7576, 789 9338
info@diasoft.ru, www.diasoft.ru