
Checked Exceptions



Java Features
Presented by Ramin Zare

مخالف ها



Bruce Eckel

Examination of small programs leads to the conclusion that requiring exception specifications could both enhance developer productivity and enhance code quality, but experience with large software projects suggests a different result – decreased productivity and little or no increase in code quality.

موافق ها



James Moliere

I must confess, checked exceptions help me sleep at night knowing that my code is correct, concise, and fairly clean. I personally prefer a language like Java that helps me enforce rules of good design -- it also prevents project 'runaways'.

Cons & Pros

Pros:

- **Fail-Fast Development**
- **Contract**

Cons:

- **Extensibility**
- **Misuse**
- **Propagation**
 - Throw early, Catch Later

Cons (Extensibility)

```
public interface ApiInterface {  
    void method1() /*ADDED LATER*/ throws RuntimeException;  
}  
  
public static void main(String[] args) {  
    ApiInterface apiImpl1 = new ApiImpl1();  
  
    apiImpl1.method1();//COMPILE ERROR IN FUTURE  
}
```

Cons (Misuse)

| | غیر قابل پیش بینی | قابل پیش بینی |
|--------------------|---|---|
| Contingent (محتمل) | <ul style="list-style-type: none">• FileNotFoundException• PrinterException• SocketTimeoutException | <ul style="list-style-type: none">• IllegalArgumentException• NumberFormatException• MalformedURLException |
| Fault (خطا) | <ul style="list-style-type: none">• OutOfMemoryError• StackOverflowError• NoSuchMethodError | <ul style="list-style-type: none">• ArrayIndexOutOfBoundsException• NullPointerException• ArithmeticException |

Cons (Propagation)

```
public class TestJava {  
    public void callApiInterfaceMethod1() throws RuleException {  
        ApiInterface apiImpl1 = new ApiImpl1();  
        apiImpl1.method1();  
    }  
  
    public static void main(String[] args) {  
        TestJava testJava = new TestJava();  
        try {  
            testJava.callApiInterfaceMethod1();  
        } catch (RuleException e) {  
            e.printStackTrace();  
        }  
    }  
}
```

Cons (Propagation)

```
long calculateBlogDiskSize() {  
    try {  
        return blogEntries  
            .stream()  
            .mapToLong(e -> Files.size(e.sourceFile))  
            .sum();  
    } catch (IOException e) {  
        throw new BlogSizeCalcException(e);  
    }  
}  
//COMPILE ERROR
```


Checked Exception or Unchecked

- آیا این وضعیت واقعا نیاز به Exception دارد؟
- آیا کلاينت میتواند در این موقعيت روشی برای فرار از اون پیدا کند؟
- آیا این وضعیت قابل پیش بینی هست؟

منابع

<http://www.mindview.net/Etc/Discussions/CheckedExceptions>

<https://programming.guide/java/checked-exceptions-good-or-bad.html>

<https://programming.guide/java/choosing-between-checked-and-unchecked-exceptions.html>

<https://kotlinlang.org/docs/reference/exceptions.html>

Thanks!

Presented by :

Ramin Zare

Thanks to:

Mohamad Taghizade

Mahdi Abedi

