

ORACLE®

# Java Essential

Обработка исключений

# Java Essential

Автор курса



Евгений Тихонов

# Java Essential

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

## Обработка исключений

# Exceptions

## Исключения

**Исключения** – неожиданные события, прерывающие нормальную работу кода и возникающие во время исполнения программы.

# try, catch

## Обработка исключений

Обработка исключительных ситуаций в Java организуется с помощью трех ключевых слов: `try`, `catch` и `finally`.

```
try {  
    // Попытка деления на ноль.  
    a = a / (2 - n);  
    System.out.println("a = " + a);  
}  
catch (Exception e) {  
    System.out.println(e.getMessage());  
}
```

Конструкция `try-catch` позволяет выделить небезопасные участки кода, в которых могут возникать исключения, и предоставляет код для их обработки, если они происходят.

Если в конструкции `try` возникает исключение, то выполняется конструкция `catch`.

# Исключения

## Обработка исключений

```
static void main(String args[]) {  
    Exception ex = new Exception("Мое Исключение");  
    try {  
        throw ex;  
    }  
    catch(Exception e) {  
        System.out.println("Обработка исключения.");  
        System.out.println(e.getMessage());  
    }  
}
```

Ключевое слово **throw** – генерирует исключение.

# Исключения

## Обработка исключений

Для создания пользовательского исключения, требуется наследование от класса `java.lang.Exception`.

```
class UserException extends Exception {  
    public void method() {  
        System.out.println("Exception");  
    }  
}
```



# Исключения

## Обработка исключений

Блок `finally` используется зачастую для установки настроек в первоначальное значение и выполняется всегда. Является не обязательным.

```
try {  
    throw new Exception("МоеИсключение");  
} catch (Exception e) {  
    System.out.println("Обработка исключения.");  
} finally {  
    System.out.println("Выведется всегда");  
}
```

# Исключения

## Обработка нескольких конструкций catch

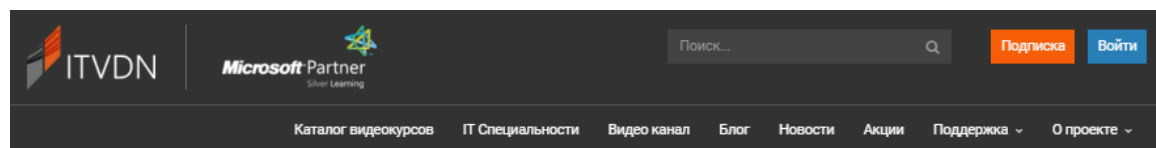
Блок **finally** используется зачастую для установки настроек в первоначальное значение и выполняется всегда. Является не обязательным.

```
try {  
    //throw new Exception("Exception");  
    throw new MyExceptionA("MyExceptionA");  
    //throw new MyExceptionB("MyExceptionB");  
}  
catch(MyExceptionB e) {  
    System.out.println(e.getMessage());  
}  
catch(MyExceptionA e) {  
    System.out.println(e.getMessage());  
}  
catch(Exception e)  
{  
    System.out.println(e.getMessage());  
}
```

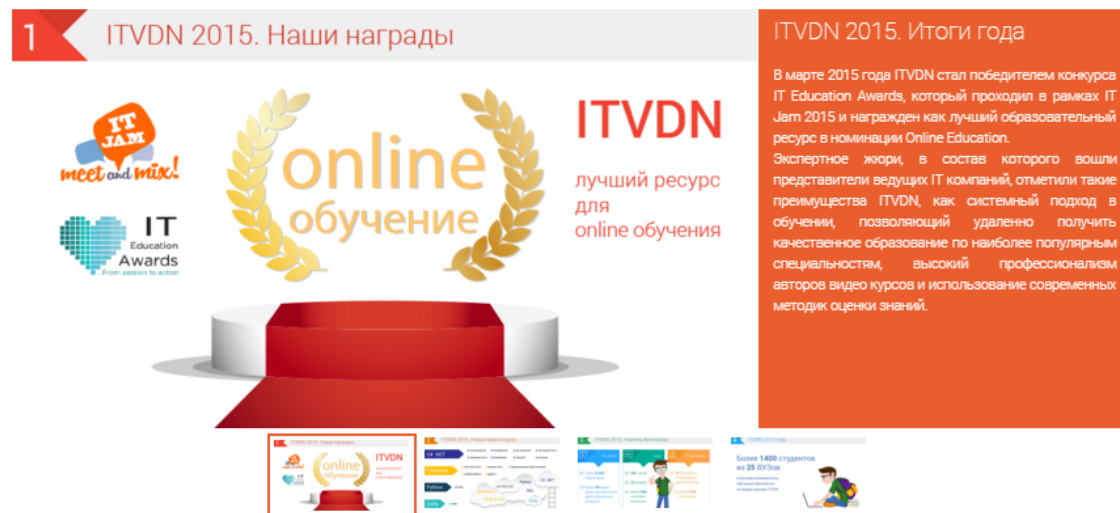
Несколько конструкций **catch** используются для отлавливания нескольких видов ошибок.

# Смотрите наши уроки в видео формате

ITVDN.com



Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://itvdn.com) для закрепления пройденного материала.



Все курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics

## Новые видео

Исключения	0
Итераторы и генераторы	0

## Популярные видео курсы

Видео курс C# Стартовый (для начинающих)	9 уроков (16 ч. 3 мин.)
Видео курс по шаблонам проектирования	29 уроков (16 ч. 7 мин.)

## Теги

.NET Developer
Frontend Developer



# Проверка знаний

## TestProvider.com

TestProvider | Мы помогаем людям оценить себя

Регистрация Войти

Главная Каталог Сертификация Microsoft Поддержка О нас

### Тестирование

Языки программирования и информационные технологии

**Microsoft**

C# ASP.NET MVC JavaScript Patterns Of Design SQL Architecture Guide WCF HTML&CSS XML SEO WPF HTML5&CSS3 JQuery XNA SharePoint GUI for Android Windows Azure Platform Microsoft Patterns&Practices TFS SCRUM ReSharper TDD WWF LINQ Entity Framework Windows Forms Refactoring Microsoft Expression Blend 4 Windows Phone 8 Windows 8 AppStore Visual Studio Tips&Tricks MSF MEF SilverLight AJAX MEF Service Oriented Architecture

Пройти тест

Наши партнеры

Microsoft Partner CyberBionic ITVDN PROMETRIC TEST CENTER PEARSON VUE Authorized Test Center Windows Azure Cloud Partner EBA

Дополнительные ресурсы:

Очное обучение On-line обучение Видео обучение

TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](http://TestProvider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



# Java Essential

Q&A

# Информационный видеосервис для разработчиков программного обеспечения

