

Java Fundamentals

Обработка исключений



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.

Java Fundamentals

Q&A