Исключения Java

В мире программирования возникновение ошибок и непредвиденных ситуаций при выполнении программы называют **исключением**. В программе исключения могут возникать в результате неправильных действий пользователя, отсутствии необходимого ресурса на диске, или потери соединения с сервером по сети. Причинами исключений при выполнении программы также могут быть ошибки программирования или неправильное использование API. В отличие от нашего мира, программа должна четко знать, как поступать в такой ситуации. Для этого в Java предусмотрен механизм исключений.  
  
Какие бывают исключения:

ArithmeticException - выбрасывается, когда возникает исключительная арифметическая ситуация.

ArrayIndexOutOfBoundsException - выбрасывается при попытке обратиться к массиву по недействительному индексу.

ArrayStoreException - выбрасывается при попытке сохранить объект несоответствующего типа в массиве объектов.

ClassCastException - выбрасывается, когда код совершает попытку привести тип объекта к подклассу, экземпляром которого он не является.

ClassNotFoundException - выбрасывается, когда приложение совершает попытку загрузить класс по его имени в строковом представлении с использованием метода forName в классе Class.

CloneNotSupportedException - выбрасывается, когда для клонирования объекта определенного класса вызван метод clone, но класс этого объекта не реализует интерфейс Cloneable.

EnumConstantNotPresentException - выбрасывается, когда приложение производит попытку обратиться к константе из перечисления по имени, но тип этого перечисления не содержит константу с указанным именем.

Exception - класс Exception и его подклассы являются подклассами Throwable и указывают на ситуации, которые могут быть перехвачены приложением.

IllegalAccessException - выбрасывается, когда приложение совершает попытку применить рефлексию, чтобы создать экземпляр (отличный от массива), присвоить полю значение или получить значение поля, вызвать метод, но текущий выполняемый метод не имеет доступа к определению указанного класса, поля, метода или конструктора.

IllegalArgumentException - выбрасывается, когда методу передан недопустимый или неприемлемый аргумент.

IllegalMonitorStateException - выбрасывается, когда нить пытается ожидать монитор объекта или отправить оповещение другим нитям, ожидающим монитор объекта, но указанный монитор не принадлежит ей.

IllegalStateException - выбрасывается, когда метод вызван в недопустимое или неприемлемое время.

IllegalThreadStateException - выбрасывается, когда нить находится в неприемлемом состоянии для выполнения запрошенной операции.

IndexOutOfBoundsException - выбрасывается, когда индекс определенного типа (например, для массива, строки или вектора) находится вне допустимого диапазона.

InstantiationException - выбрасывается, когда приложение совершает попытку создать экземпляр класса с использованием метода newInstance класса Class, но создать экземпляр указанного класса невозможно.

InterruptedException - выбрасывается, когда нить находится в состоянии ожидания, сна или занята каким-либо другим образом, но ее работа прервана либо перед выполнением действия, либо после его выполнения.

NegativeArraySizeException - выбрасывается, когда приложение совершает попытку создать массив отрицательного размера.

NoSuchMethodException - выбрасывается, когда невозможно найти указанный метод.

NullPointerException - выбрасывается, когда приложение совершает попытку использовать значение null, но требуется указать объект.

NumberFormatException - выбрасывается, когда приложение совершает попытку преобразовать строку в один из числовых типов, но строка имеет недопустимый формат.

RuntimeException - это родительский класс для тех исключений, которые могут быть выброшены при нормальной работе виртуальной машины Java.

SecurityException - выбрасывается менеджером безопасности при нарушении безопасности.

StringIndexOutOfBoundsException - выбрасывается методами класса String при попытке использовать отрицательный индекс или индекс, превышающий размер строки.

TypeNotPresentException - выбрасывается, когда приложение совершает попытку доступа к типу с указанием его имени в виде строки, но не удается найти определение типа с указанным именем.

UnsupportedOperationException - выбрасывается, когда запрошенная операция не поддерживается.

Пример исключения с конструктором по умолчанию:

**class** MyException **extends** Exception{

// Конструктор по умолчанию

MyException(){}

}

**class** Main{

**public** **static** **void** main(**String**[] args){

**try**{

MyException e = new MyException();

**throw** e;

}

**catch**(MyException ex){

System.out.println("Перехвачено пользовательское исключение");

}

}

}

Пример исключения с параметром в конструкторе:

**class** MyException **extends** Exception{

MyException(**String** msg){

**super**(msg);

}

}

**class** Main{

**public** **static** **void** main(**String**[] args){

**try**{

MyException e = new MyException("Перехвачено пользовательское исключение с информацией");

**throw** e;

}

**catch**(MyException ex){

System.out.println(ex.getMessage());

}

}

}