**What is an Exception?**

An exception is an unwanted or unexpected event, which occurs during the execution of a program i.e at run time, that disrupts the normal flow of the program’s instructions

**Error vs Exception**

Error: An Error indicates serious problem that a reasonable application should not try to catch.

Exception: Exception indicates conditions that a reasonable application might try to catch.

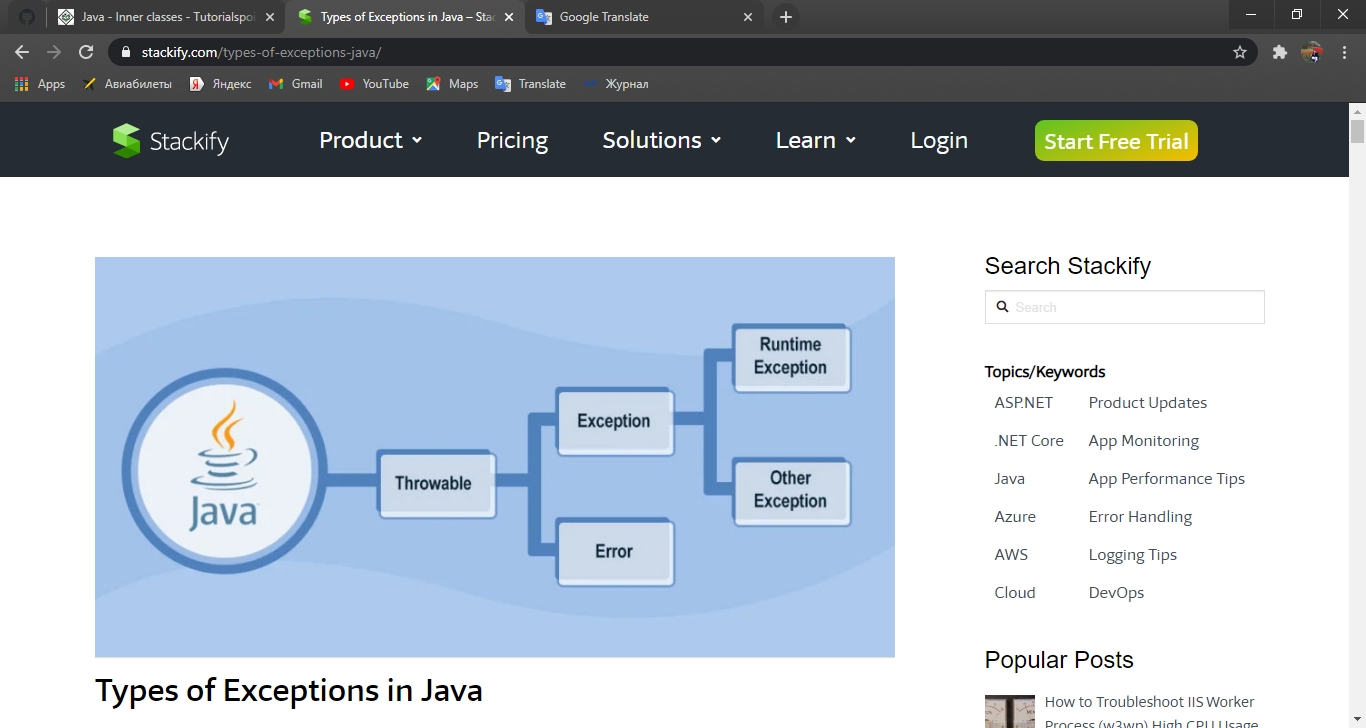
**Что такое исключение?**

Исключением является нежелательное или неожиданное событие, которое происходит во время выполнения программы, то есть во время выполнения, которое нарушает нормальный поток инструкций программы.

**Ошибка против исключения**

Ошибка: ошибка указывает на серьезную проблему, который не может быть решено by try and catch

Исключение: исключение указывает условия, которые разумное приложение может try and catch.



**Unchecked exception**

The unchecked exceptions are those exceptions that occur during the execution of the program. Hence they are also referred to as **Runtime exceptions**. These exceptions are generally ignored during the compilation process.

**Непроверенное исключение**

Непроверенные исключения - это те исключения, которые возникают во время выполнения программы. Следовательно, они также называются исключениями времени выполнения. Эти исключения обычно игнорируются в процессе компиляции.

**Examples:**

**NullPointerException:** Этот тип исключения возникает, когда вы пытаетесь получить доступ к объекту с помощью ссылочной переменной, текущее значение которой равно нулю или пусто.

This type of exception occurs when you try to access an object with the help of a reference variable whose current value is null or empty.

**ArrayindexoutofBond:**

**aruthmeticException**

### Checked exception

Checked exceptions are also known as compile-time exceptions as these exceptions are checked by the compiler during the compilation process to confirm whether the exception is handled by the programmer or not.

**Проверено исключение**

Проверенные исключения также известны как исключения времени компиляции, поскольку эти исключения проверяются компилятором во время процесса компиляции, чтобы подтвердить, обработано ли исключение программистом или нет.

**IOExceotion:** This type of exception occurs while using file I/O stream operations.

Этот тип исключения возникает при использовании операций с файловым потоком ввода-вывода.

Keywords which are used for handling exception:

Try

Catch

Finally

Throw

Throws

**Anonymous Class:** It is an inner class without a name and for which only a single object is created. Anonymous classes enable you to make your code more concise. They enable you to declare and instantiate a class at the same time. They are like local classes except that they do not have a name.

**Анонимный класс:** это внутренний класс без имени, для которого создается только один объект. Анонимные классы позволяют сделать код более лаконичным. Они позволяют вам одновременно объявлять и создавать экземпляры класса. Они похожи на местные классы, за исключением того, что у них нет названия.

**Local Classes:** are the inner classes that are defined inside a block. Generally, this block is a method body. Sometimes this block can be a for loop, or an if clause.it is a kind of nested inner class.

**Локальные классы:** внутренние классы, которые определены внутри блока. Обычно этот блок является телом метода. Иногда этот блок может быть циклом for или предложением if. Это своего рода вложенный внутренний класс.

**Nested classes:** In Java, it is possible to define a class within another class, such classes are known as nested classes. They enable you to logically group classes that are only used in one place.

**Вложенные классы:** в Java можно определить класс внутри другого класса, такие классы известны как вложенные классы. Они позволяют вам логически группировать классы, которые используются только в одном месте.