

Java: основы языка 2

Введение в анализ данных

Сергей Рыбалкин

Java. Зачем?



+

- Web services
- Databases access application
- Highload systems
- Low latency systems
- Distributed systems
- JVM

Java в качестве основного рабочего инструмента Data Scientist
<https://www.youtube.com/watch?v=1TYIhIVAqQo>

—

- CPU bound applications
- Hardware-access systems

Enum



```
enum SomeNumbers { One, Two, Three }
```

```
enum MoreNumbers extends SomeNumbers { Four, Five }
```

```
// ??? ООП ???
```

```
SomeNumbers five = MoreNumbers.Five;
```

```
MoreNumbers two = SomeNumbers.Two;
```

Инстанцирование объектов

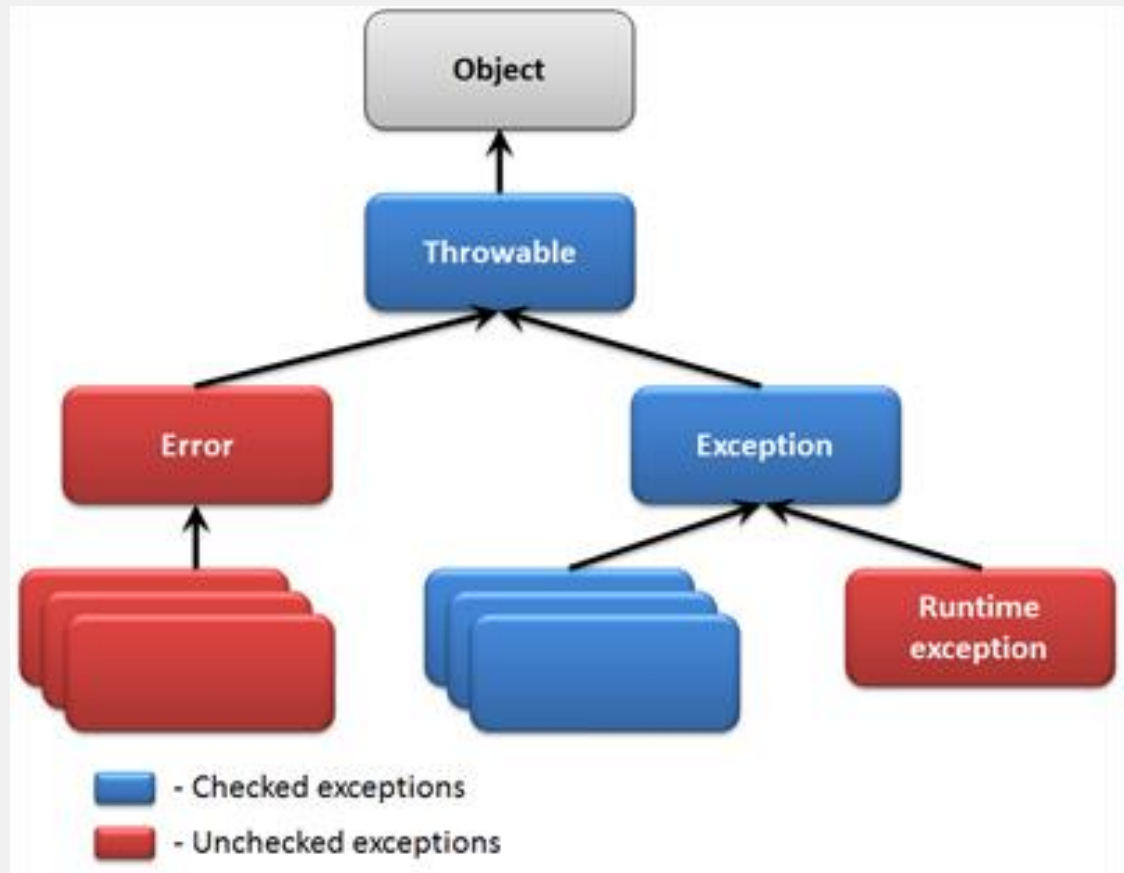


Рассмотрим
`ru.mail.sphere.instantiation`



Иерархия исключений

- Throwable(
 String message,
 Throwable cause)
- getStackTrace()
- getMessage()
- getCause()



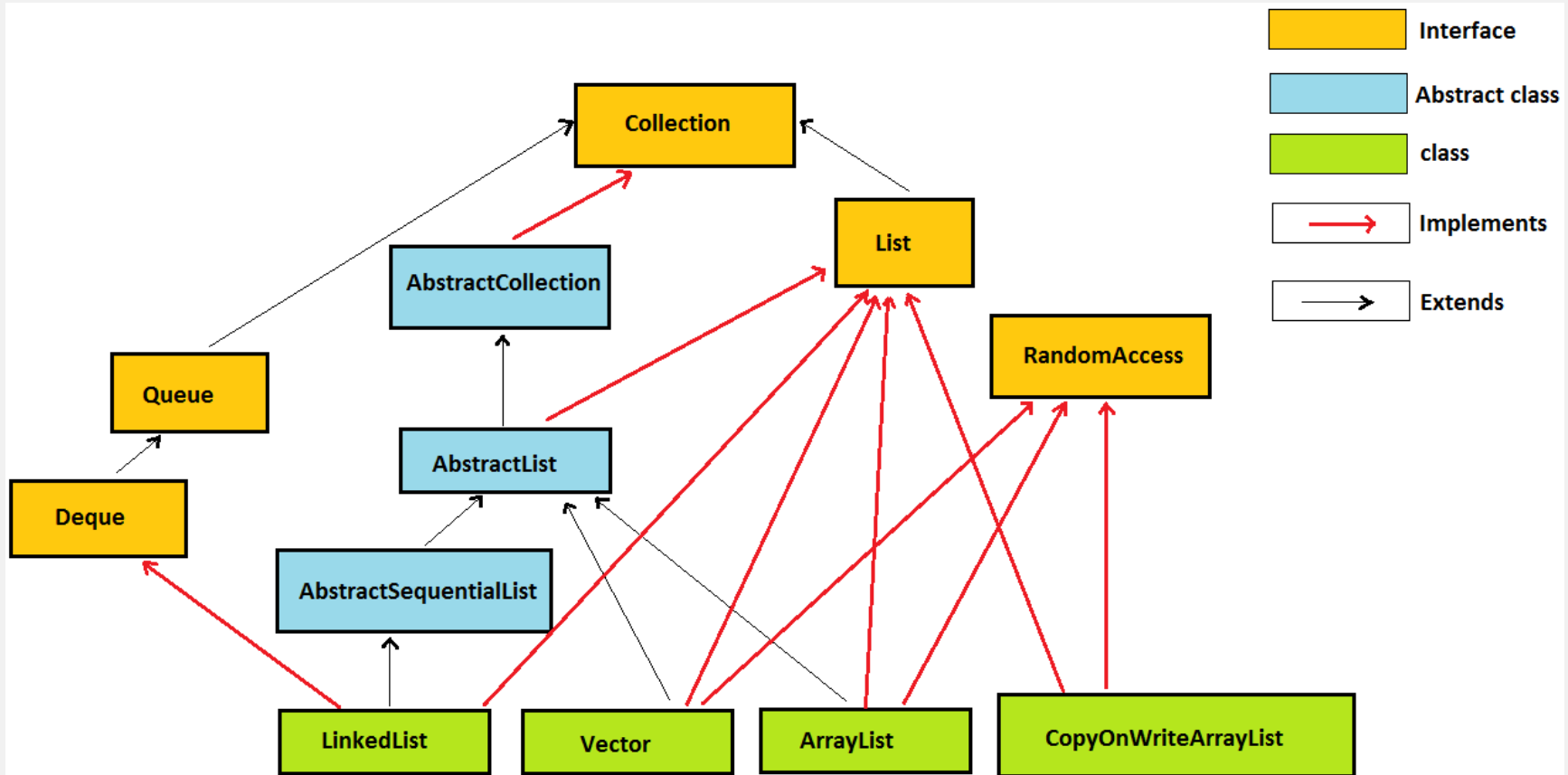
Применение исключений



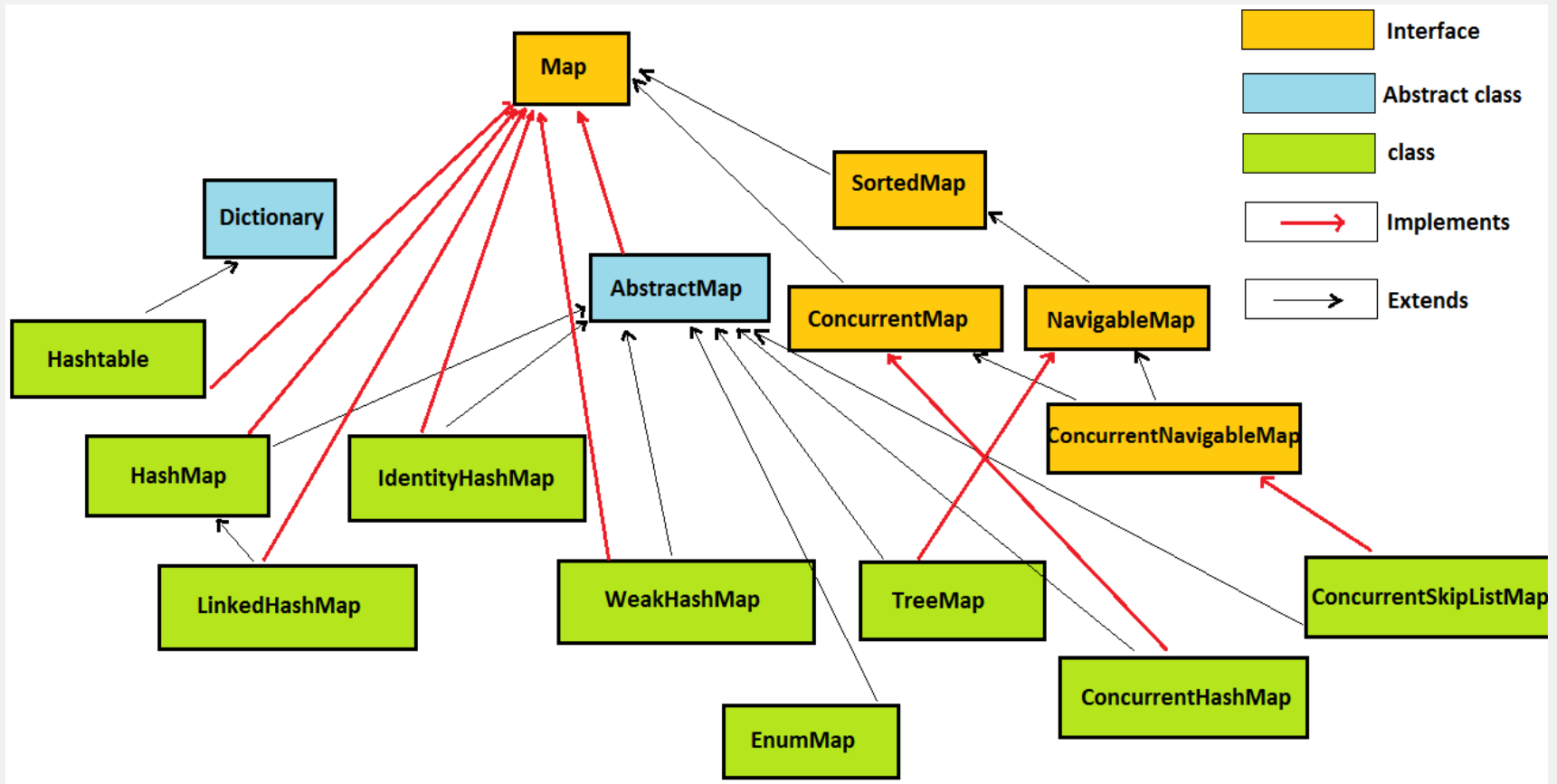
Рассмотрим

`ru.mail.sphere.exception.ExceptionHandler`

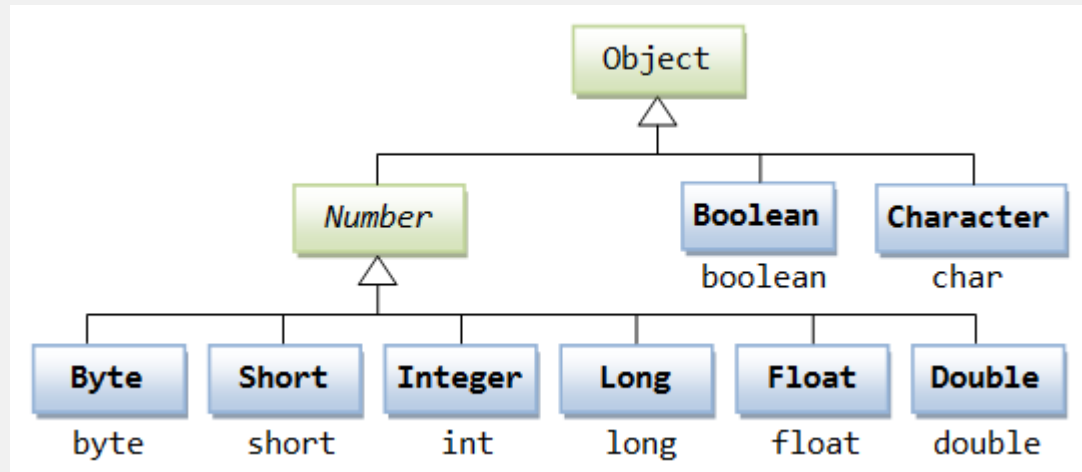
Collection Framework



Collection Framework. Bonus



Collection Framework. Primitive types ?



#Autoboxing #Unboxing #PrimitiveTypes #WrapperClasses

Работа с коллекциями



Рассмотрим

`ru.mail.sphere.collection.CollectionSamples`



1. Позволяют реализовать обобщенные алгоритмы
2. Обеспечивают безопасные приведения типов.
3. Убирают из кода явные приведения типов
4. Обрабатываются в compile time. В runtime отсутствуют*

```
// generic class
class Clazz<T> {
    public void foo(T t) { }
}
```

```
// generic method
<T> void foo(T t);
```

Java 5+

#Wildcards #TypeBounds #TypeErasure #BridgeMethod
#Raw-Type



Ограничения

Есть несколько типов:

- `<T extends A & B & C>`
- `<? extends A>` - wildcard с extends ограничениями
- `<? super A>` - wildcard с super ограничениями
- `<?>` - wildcard без ограничений

`Class<Object>` не является предком `Class<Integer>`

PECS – producer extends consumer super
- Effective Java – J. Bloch



Рассмотрим

`ru.mail.sphere.PositiveNumbersCollection`

И что теперь?



Java

1. JDBC
 2. IO
 3. Concurrency
 4. Lambdas & Stream API
 5. Libs
 1. Guava
 2. Apache Commons
 3. Lombok <https://projectlombok.org/>
- Thinking in Java (4th Edition) – B. Eckel
 - Effective Java (2nd Edition) – J. Bloch
 - Java Concurrency in Practice – B. Goetz

И что теперь?



Web & Frameworks & etc.

1. Web services
2. Spring <http://spring-projects.ru/projects/spring-framework/>
3. ORM <http://hibernate.org/orm/>
4. DB + SQL <http://www.postgresql.org/>
5. Hadoop <http://hadoop.apache.org/>
6. Spark <http://spark.apache.org/>
7. NoSQL



**Спасибо за
внимание!**

Сергей Рыбалкин

s.rybalkin@corp.mail.ru