





**EXCEPTIONS**

THROWS ARE UGLY OTS

MAGNÉT-QUALITÄT S-THOMAS & SÖHN

```
class Weapon {
    Weapon() {
        throw new RuntimeException("NoMoreArrow");
    }
}

class Target {
    Target() {
        throw new RuntimeException("TooMuchFog");
    }
}

class Impacted {}

public class Main {
    Weapon armYouBow = new Weapon();
    Target targetMonster = new Target();
    Impacted hitMonster(Weapon w, Target t) {
        return new Impacted();
    }
}
```

*Exemples en java*

# EXCEPTIONS

THROWS ARE UGLY GOTOS

```
class Weapon {
    Weapon() {
        throw new RuntimeException("NoMoreArrow");
    }
}
class Target {
    Target() {
        throw new RuntimeException("TooMuchFog");
    }
}
class Impacted {}

public class Main {
    Weapon armYouBow = new Weapon();
    Target targetMonster = new Target();
    Impacted hitMonster(Weapon w, Target t) {
        return new Impacted();
    }
}
```

*Exemples en Java*



# EXCEPTIONS

THROW ARE UGLY GOTOS LOST IN TIME OF ASYNC / FUTURE

```
import java.util.concurrent.*;

class Weapon {
    Weapon() { throw new RuntimeException("NoMoreArrow"); }
}

class Target {
    Target() { throw new RuntimeException("TooMuchFog"); }
}

class Impacted {}

public class Main {
    static Impacted hitMonster(Weapon w, Target t) {
        return new Impacted();
    }

    public static FutureTask<Void> attack = new FutureTask<>(new Runnable() {
        @Override
        public void run() {

            Weapon armYouBow = new Weapon();
            Target targetMonster = new Target();
            hitMonster(armYouBow, targetMonster);

        }
    }, (Void) null);

    public static void main(String[] args) {
        ExecutorService es = Executors.newSingleThreadExecutor();
        es.execute(attack);
        es.shutdown();
    }
}
```