

# JOOX

Jakub Kubrynski

[jkubrynski@gmail.com](mailto:jkubrynski@gmail.com)

<http://www.kubrynski.com/>



JOOX

---

Java  
Object  
Oriented  
XML



# CORE PRINCIPLES

---

- Simple wrapper for `org.w3c.dom`
- Inspired JOOQ
- Matches jQuery API
- Uses DOMParser internally



# GETTING JOOX

---

## - Maven

```
<dependency>
```

```
  <groupId>org.jooq</groupId>
```

```
  <artifactId>joox</artifactId>
```

```
  <version>1.1.0</version>
```

```
</dependency>
```

## - Gradle 'org.jooq:joox:1.1.0'



# READING FILE

---

```
import static org.joox.JOOX.*;
```

```
Match document =
```

```
$(new File("pom.xml"));
```



# FINDING

---

Match dependencies =

```
document.find("dependencies");
```



# FILTERING

---

Match dependencies =

```
document.find("dependency")  
  .filter(even());
```

- or(...), and(...), all(), none()
- even(), odd()
- gt(...), idt(...)



# XPATH

---

Match dependencies =

```
document.xpath("//dependency");
```



# ATTRIBUTES

---

String xmlns =

document.getAttribute("xmlns");

String newValue =

document.setAttribute("id", "100");



# MODIFICATION

---

document

.children("version")

.content("1.0.1-SNAPSHOT");

Match trans = \$("trans");

trans.append(\$("id", "102"));

trans.append(\$("client", \$("name", "Jan")));



WHAT  
ELSE?



# JOOQ

---

```
- create.selectFrom(BOOK)
    .where(PUBLISHED_IN.equal(2011))
    .and(LANGUAGE.eq("PL"))
    .orderBy(TITLE)
    .limit(20)
    .offset(100)
```



# JOOR

---

```
String world = on("java.lang.String")  
    .create("Hello World")  
    .call("substring", 6)  
    .call("toString")  
    .get();
```



# LINKS

---

- <http://www.jooq.org/>
- <http://code.google.com/p/joox/>
- <http://code.google.com/p/joor/>



DEMO!!