开源社
kaiyuanshe

Kotlin 爬虫

抓取多平台播客聆听数据实战

范圣佑
JetBrains 技术布道师

COSCon'23 (10/28-29)

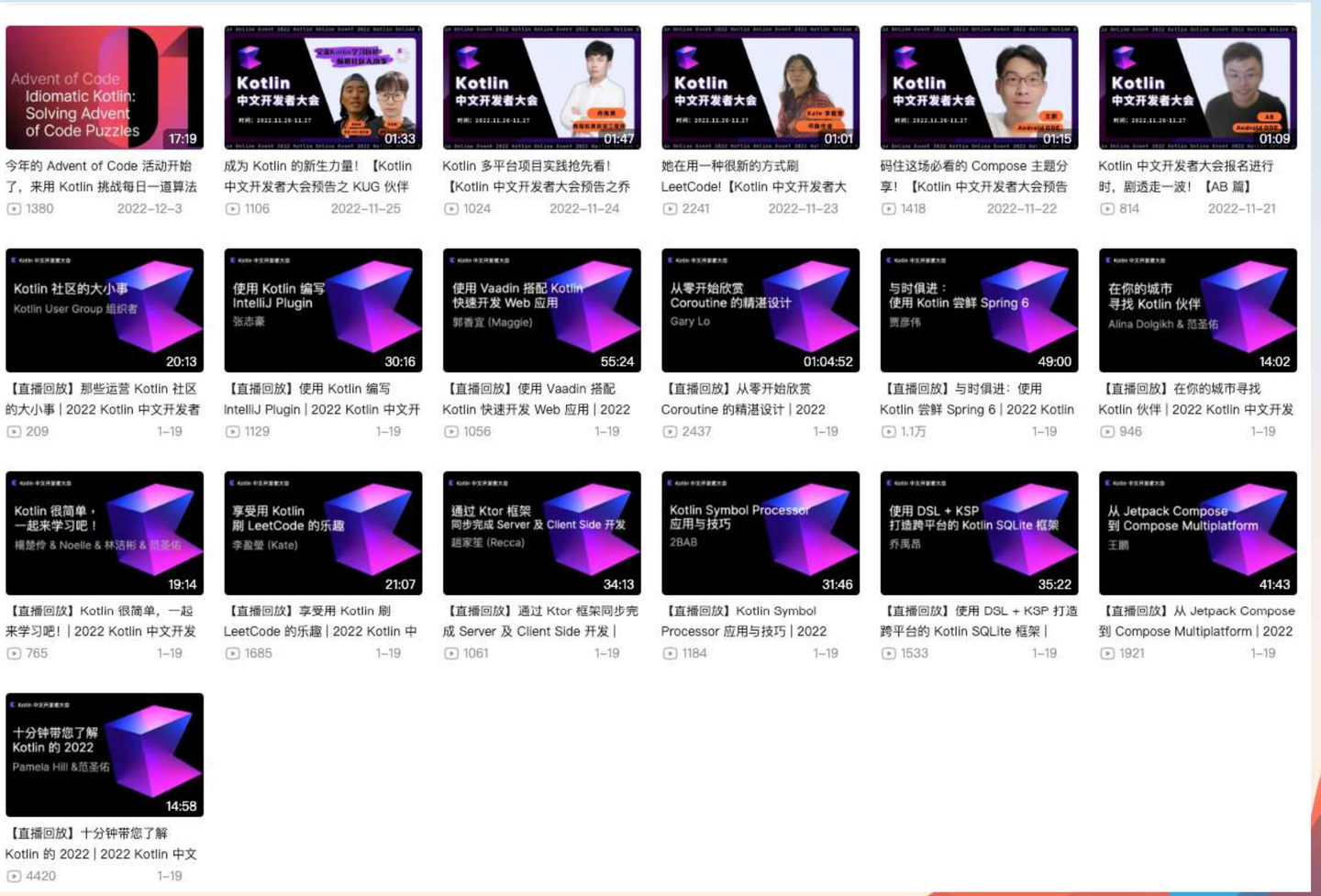
2023 第八届中国开源年会　　开源：川流不息、山海相映

《Kotlin 炉边漫谈》背后的故事

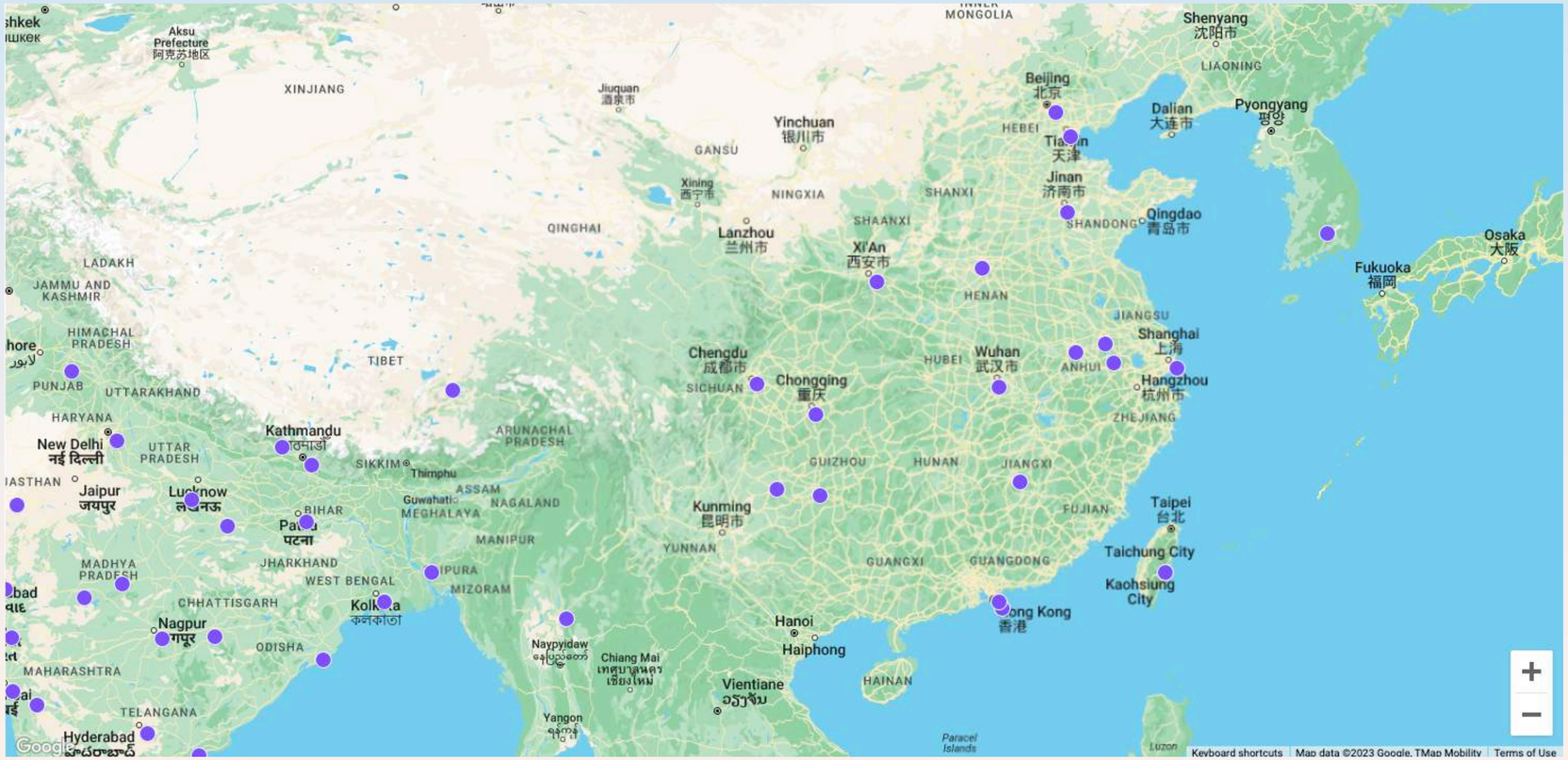
2023 第八届中国开源年会　　开源：川流不息、山海相映

2021 年起举办

Kotlin 中文开发者大会



培育各城市的
Kotlin User Group



China Mainland

Beijing KUG

Chengdu KUG

Chongqing KUG

Guangzhou KUG

Guizhou KUG

Hefei KUG

JiangXi KUG

Jinan KUG

Nanjing KUG

Shanghai KUG

Shenzhen KUG

Suzhou KUG

Tianjin KUG

Wuhan KUG

Xi'An KUG

Zhengzhou KUG

Hong Kong

Hong Kong KUG

Taiwan

Taiwan KUG

目前有 18 个城市有说中文的 Kotlin 社区！

跟 KUG 的小伙伴一时兴起

《Kotlin 炉边漫谈》正式开播！

共同主持

禹昂

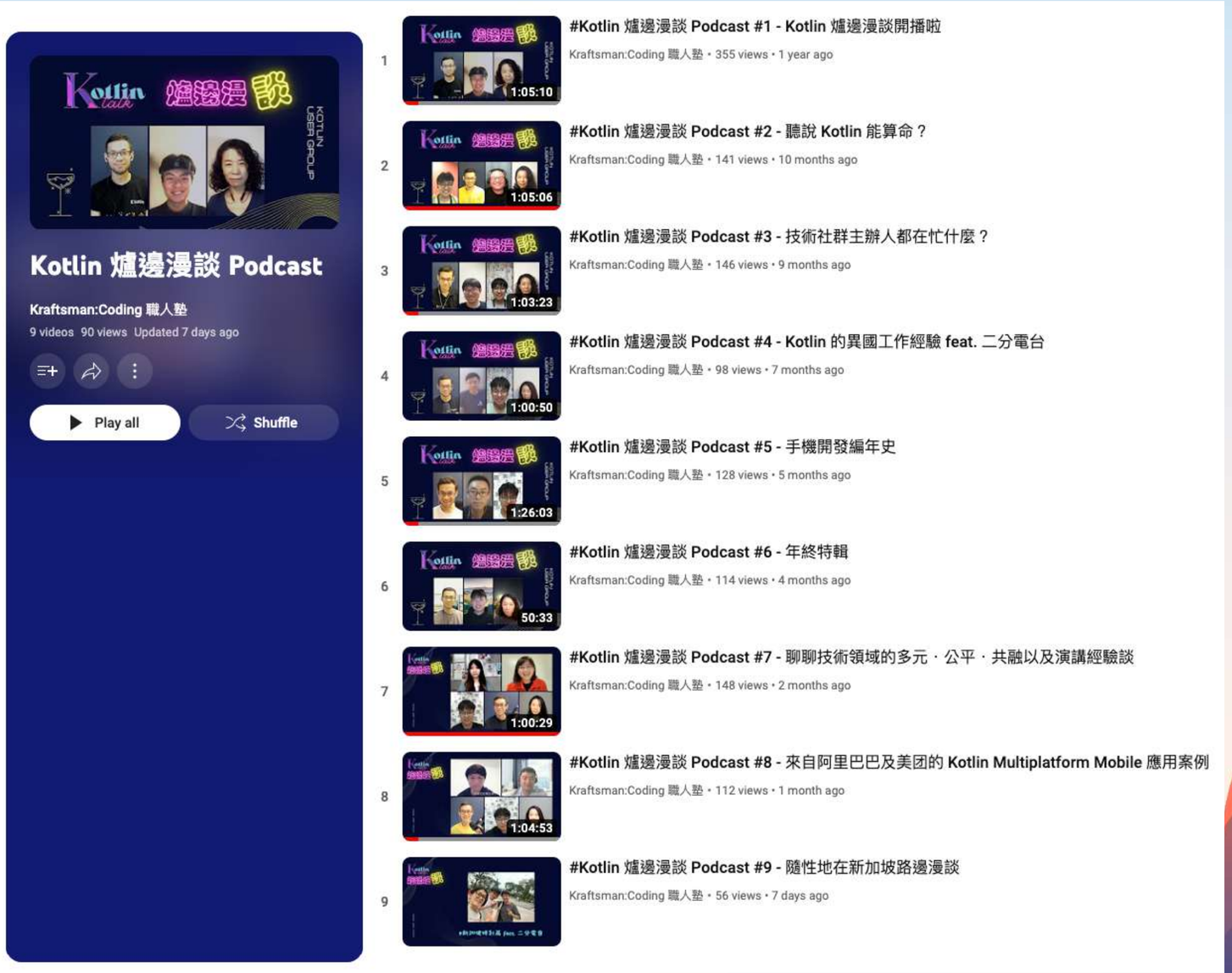
Maggie

现阶段成果

3 位共同主持人

1.5 年

12 期节目



尽力满足各种聆听习惯

2 种语言

10 个平台

YouTube

bilibili

Firstory

小宇宙

Spotify

听

喜马拉雅

蜻蜓FM

Apple Podcasts

Google Podcasts

荔枝

YouTube

bilibili

小宇宙

Firstory

蜻蜓FM

Spotify®

该怎么知道各平台的聆听数据呢？

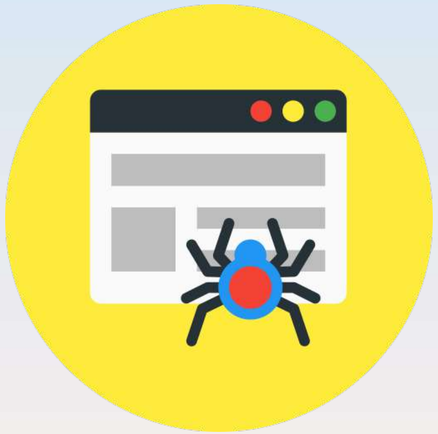
荔枝

喜马拉雅

Apple Podcasts

Google Podcasts

部份平台把数据写在网页上



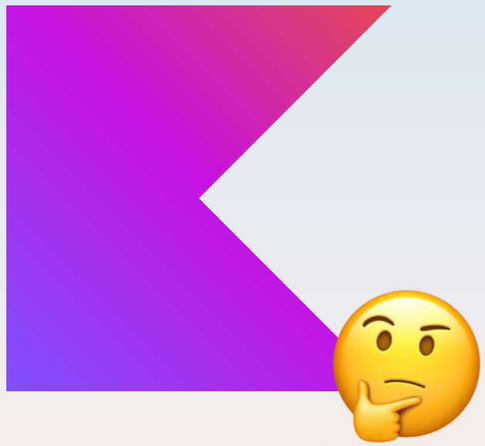
说到网络爬虫，大家想到的编程语言是？

在京东搜索

「爬虫」关键字


爬虫
全部
店铺
口碑
人民邮电出版社京东自营官方...
自营
113万人关注 8年老店
进店
¥69.00
¥68.70
¥77.40
明日科技京东自营官方旗舰店
自营
2.9万人关注 5年老店
进店
¥47.70
¥47.70
¥48.60
机械工业出版社京东自营官方...
自营
40.3万人关注 8年老店
进店
¥100.40
¥68.60
¥65.10
搜书文化图书专营店
26人关注
进店

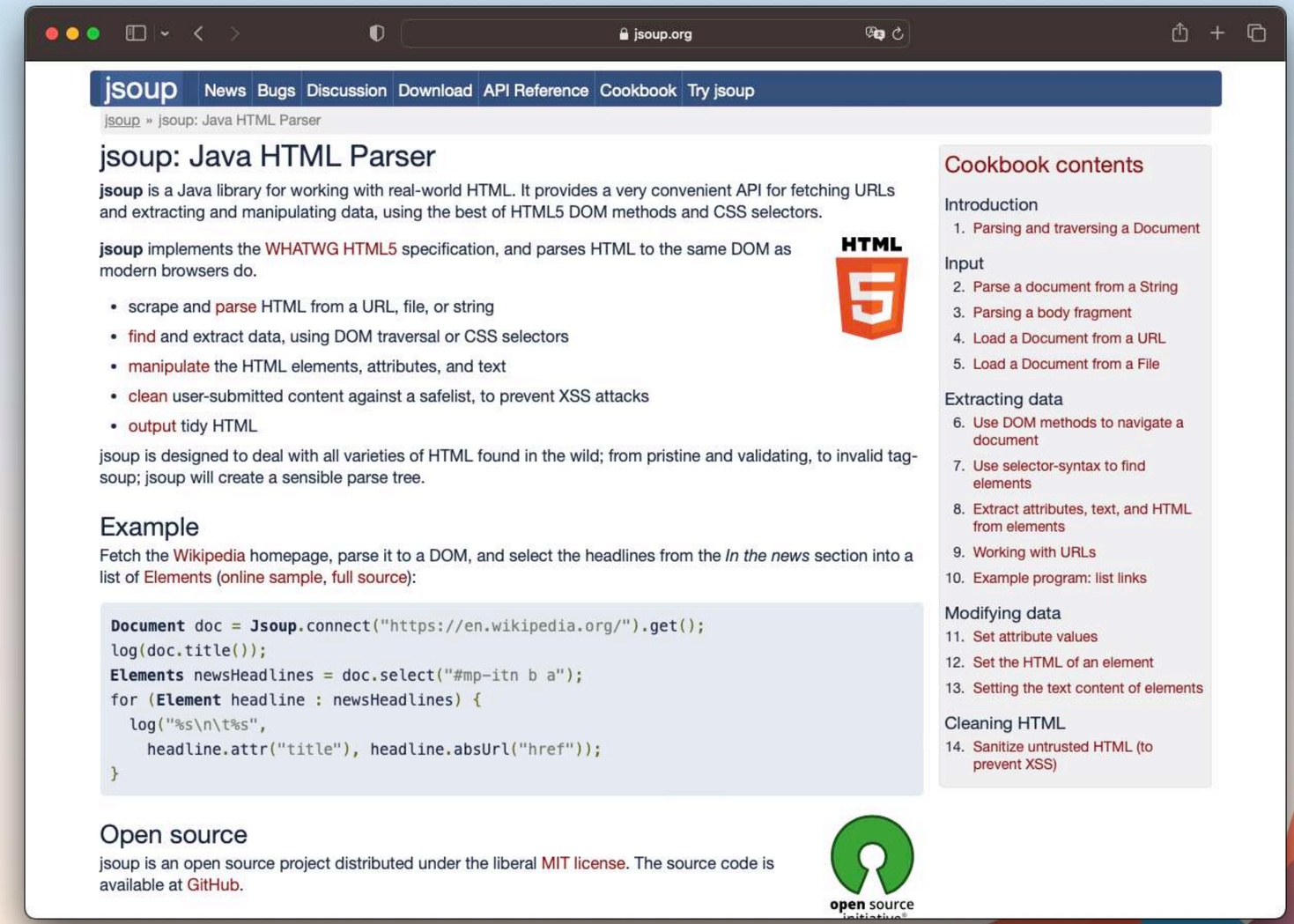

爬虫
全部
店铺
口碑
综合推荐
销量
价格
筛选
京东物流
配送全球
形态
品牌
总净含量
仅剩1件
自营 实战Python网络爬虫
¥87.40 定价¥99.00 8.83折
7天价保
2万+条评价 99%好评
清华大学出版社京东自营官方旗舰店 进店
仅剩2件
自营 Python网络爬虫权威指南 第2版
¥68.70 定价¥79.00 8.70折
7天价保
5000+条评价 98%好评
人民邮电出版社京东自营官方旗舰店 进店
仅剩1件
自营 python网络爬虫与数据可视化应
¥101.70 定价¥109.00 9.34折
7天价保 每满100减50
5万+条评价 98%好评
中国水利水电出版社京东自营官方... 进店
自营 Python网络爬虫从入门到精通
¥88.10 定价¥99.80 8.83折
7天价保
5000+条评价 97%好评
清华大学出版社京东自营官方旗舰店 进店



Python 做得到的事，Kotlin 也行么？

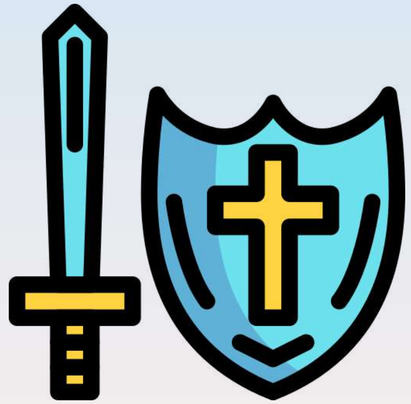
当然行！而且也是 soup！

https://jsoup.org/



网页信息抓取

- 使用 HTTP Client 下载目标网页
- 使用 DOM Parser 套件抓取目标位置的字符串

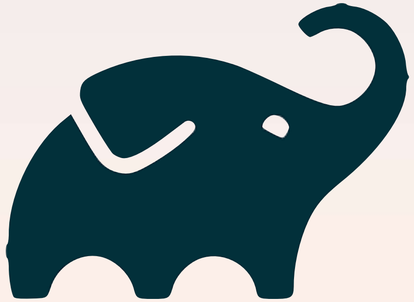


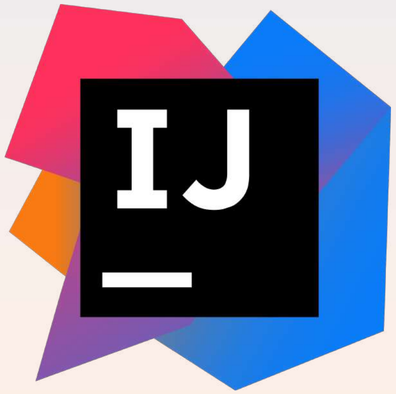
实战 Kotlin 爬虫抓取播客聆听数据

2023 第八届中国开源年会　　开源：川流不息、山海相映

开发环境

- Java LTS 17+
- Gradle (or Maven)
- IntelliJ IDEA





添加 jsoup 库

```
dependencies {
    // ...
    implementation("org.jsoup:jsoup:$version")
    // ...
}
```

jsoup 基本语法

```
// 抓取目标网页
val url = "..."
val doc = Jsoup.connect(url).get()


// 通过 CSS Selector 定位至目标位置
val elements = doc.select("div.info div.category .count")


// 再从目标位置开始抓取其中的字符串
elements.text()
elements.html()
elements.outerHtml()
elements.attr("...")
```

从喜马拉雅抓数据


```
<!-- 页面内容 -->
<div class="info kn_">
    <div class="category kn_">
        <span class="count kn_">
            // ...
        </span>
        <span class="count kn_">
            <i class="xuicon xuicon-erji1 kn_"></i>
            546
        </span>
    </div>
</div>
```

```
// 抓取网页并定位
val url = "https://www.ximalaya.com/sound/$soundId"
val doc = Jsoup.connect(url).get()
val element = doc.select("div.info div.category .count")[1]
// 抓取字符串并转成整数
element.text().trim().toInt()
```

从蜻蜓 FM 抓数据


```
<!-- 页面内容 -->
<div class="info">
    <div>
        <span class=“propTxt">播放: </span>
        <span>11次</span>
    </div>
</div>
```

```
// 抓取网页并定位
val url = "https://www.qingting.fm/.../programs/$programId"
val doc = Jsoup.connect(url).get()
val element = doc.select("div.info div span")
    .first { node -> node.html().contains("次") }

// 抓取字符串并转成整数
element.text().replace("次", "").toInt()
```

等一下！抓回来的画面一片白阿？

从 YouTube 抓数据


```
<!-- 页面内容 -->
<!DOCTYPE html>
<html>
<head>
    <div>
        <meta itemprop="interactionCount" content="128">
    </div>
</head>
</html>
```

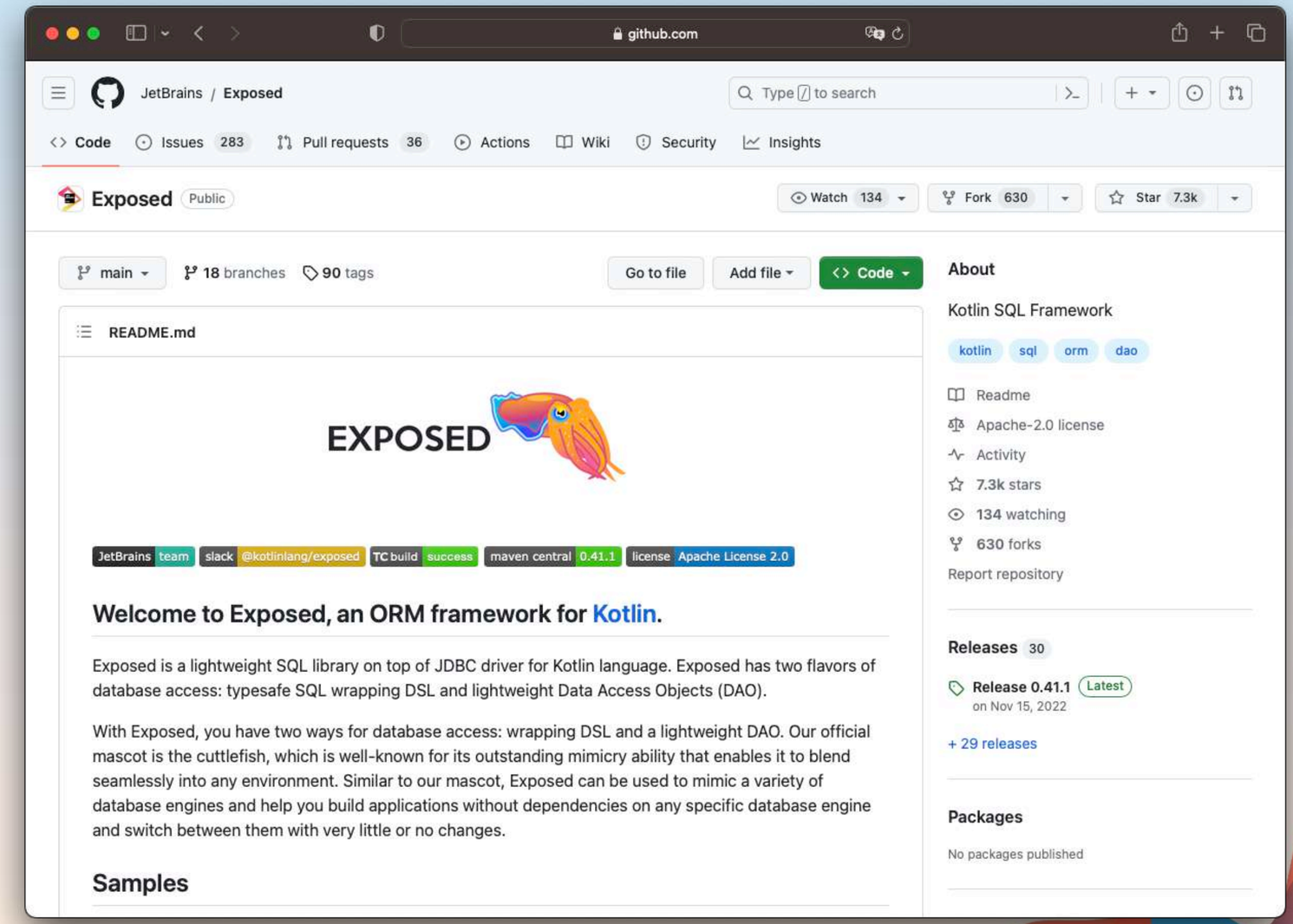
```
// 抓取网页并定位
val url = "https://www.youtube.com/watch?v=$videoId"
val doc = Jsoup.connect(url).get()
val element = doc
        .select("meta[itemprop=interactionCount]").first()

// 抓取字符串并转成整数
element?.attr("content")?.toIntOrNull() ?:0
```

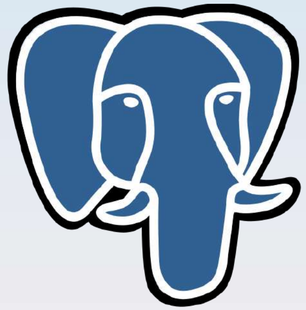
需要将抓取的数据存储至数据库

集成 Exposed ORM 框架

https://github.com/JetBrains/Exposed



支持主流数据库

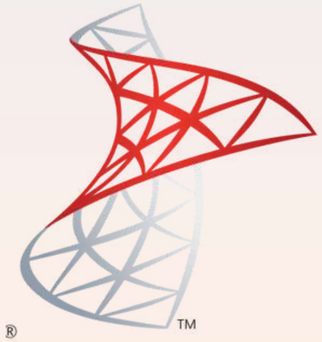


PostgreSQL

SQLite

H2 database

Oracle



SQL Server

添加 Exposed 及 Connector 库

```
dependencies {

    // ...

    implementation("org.jetbrains.exposed:exposed-core:$version")
    implementation("org.jetbrains.exposed:exposed-dao:$version")
    implementation("org.jetbrains.exposed:exposed-java-time:$version")
    implementation("org.jetbrains.exposed:exposed-jdbc:$version")
    implementation("mysql:mysql-connector-java:$version")

    // ...

}
```

两种方式操作数据

typesafe SQL wrapping DSL

lightweight data access objects

设定数据结构

```
// 声明 Table
object Statistics : IntIdTable(name = "statistics") {
    val episode = integer("episode")
    val platform = integer("platform")
    val listenNumber = integer("listen_number")
    val scrapedAt = datetime("scraped_at")
}

// 声明 Entity
class Statistic(id: EntityID<Int>) : IntEntity(id) {
    companion object : IntEntityClass<Statistic>(Statistics)

    var episode by Statistics.episode
    var platform by Statistics.platform
    var listenNumber by Statistics.listenNumber
    var scrapedAt by Statistics.scrapedAt
}
```

将抓取的数据写入

```
// 数据库连线
Database.connect(
    url = "jdbc:mysql://.../...",
    driver = "com.mysql.cj.jdbc.Driver",
    user = "...",
    password = "...",
)


// 写入数据
transaction {
    Statistic.new {
        episode = ...
        platform = ...
        listenNumber = scraper.scrape(…)
        scrapedAt = LocalDateTime.now()
    }
}
```

需要处理不同数据格式？

处理不同数据格式

- Text
- HTML
- JSON
- XML
- CSV

页面上的数据需等 JavaScript 载入后才能抓？

添加 Selenium 及 WebDriverManager 库

```
dependencies {

    // ...

    implementation("io.github.bonigarcia:webdrivermanager:$version")
    implementation("org.seleniumhq.selenium:selenium-java:$version")

    // ...

}
```

通过浏览器抓取网页数据

```
// 取 WebDriverManager 取得 Selenium Driver
WebDriverManager.firefoxdriver().setup()

// 设置浏览器参数
val options = FirefoxOptions()
                    .addArguments("--headless")
val driver = FirefoxDriver(options)

// 以浏览器读取目标网页
driver.get("...")

// 抓取页面上指定位置的数据
val element = driver.findElements(
    By.cssSelector("...")
).first()
element.text.toIntOrNull() ?: 0

// 关闭浏览器
driver.quit()
```

其他支持 Java 的

Headless 浏览器

- HTMLUnit
- Playwright
- Puppeteer (with Jvppeteer)
- 其他第三方服务

框架比较 👉 Playwright vs Selenium vs Cypress: A Detailed Comparison

需要定期抓取数据？

添加 JobRunr 及相依库

```
dependencies {

    // ...

    implementation("org.jobrunr:jobrunr:$version")
    implementation("com.fasterxml.jackson.core:jackson-databind:$ver")

    // ...

}
```

定时抓取数据

```
// 设置 JobRunr
JobRunr.configure()
    .useStorageProvider(...)
    .useBackgroundJobServer()
    .useDashboard(...)
    .initialize()

// 设置爬虫
val scraper = Scraper()

// 设置定时抓取数据
BackgroundJob.scheduleRecurrently(Cron.hourly()) {
    scraper.scrape()
}
```

除了 jsoup 外，还有其他选择吗？

添加 skrape{it} 库

```
dependencies {
    // ...
    implementation("it.skrape:skrapeit:$version")
    // ...
}
```

抓取页面数据

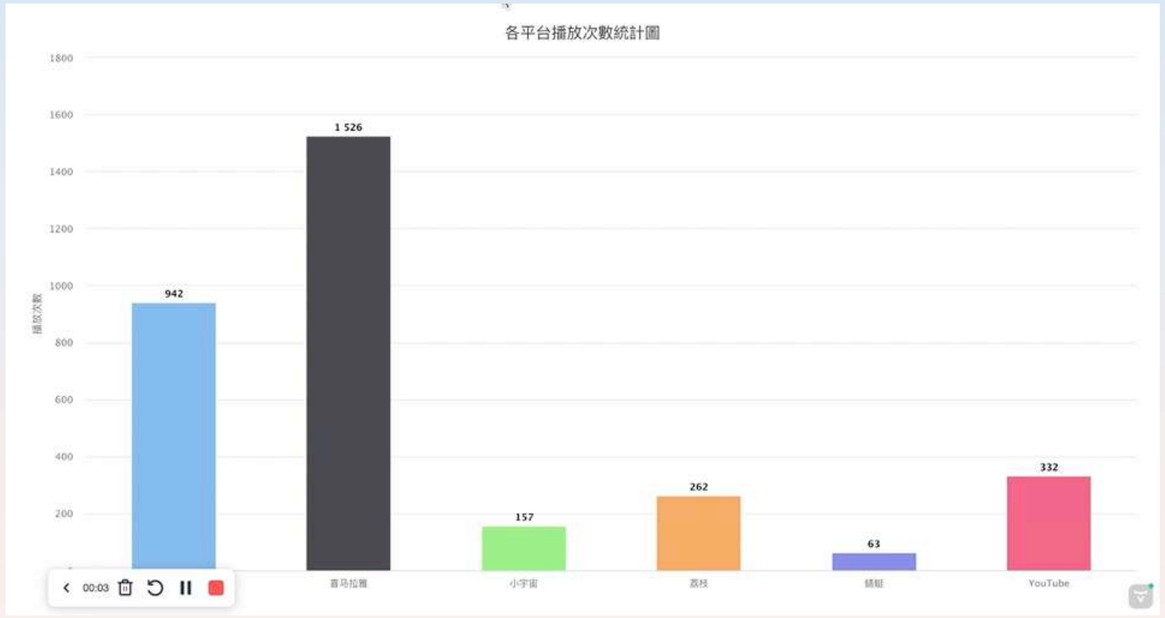
```
fun main() {
    val links: List<String> = skrape(HttpFetcher) {
        request {
            url = "https://kotlinlang.org/docs/reference/"
        }
        response {
            htmlDocument {
                a {
                    findAll {
                        eachHref
                    }
                }
            }
        }
    }
    println(links)
}
```

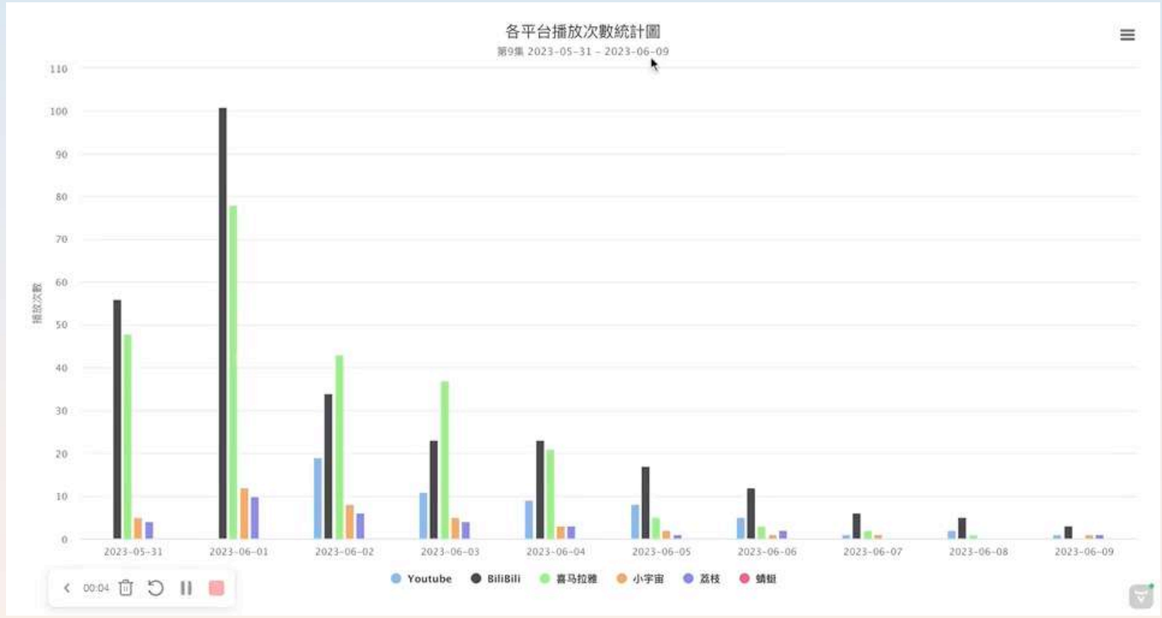
本日分享回顾

Kotlin 也能写爬虫！

- 无需混用多种语言
- 丰富的库、框架及工具
- 成熟的开发生态系

更进一步 (WIP)

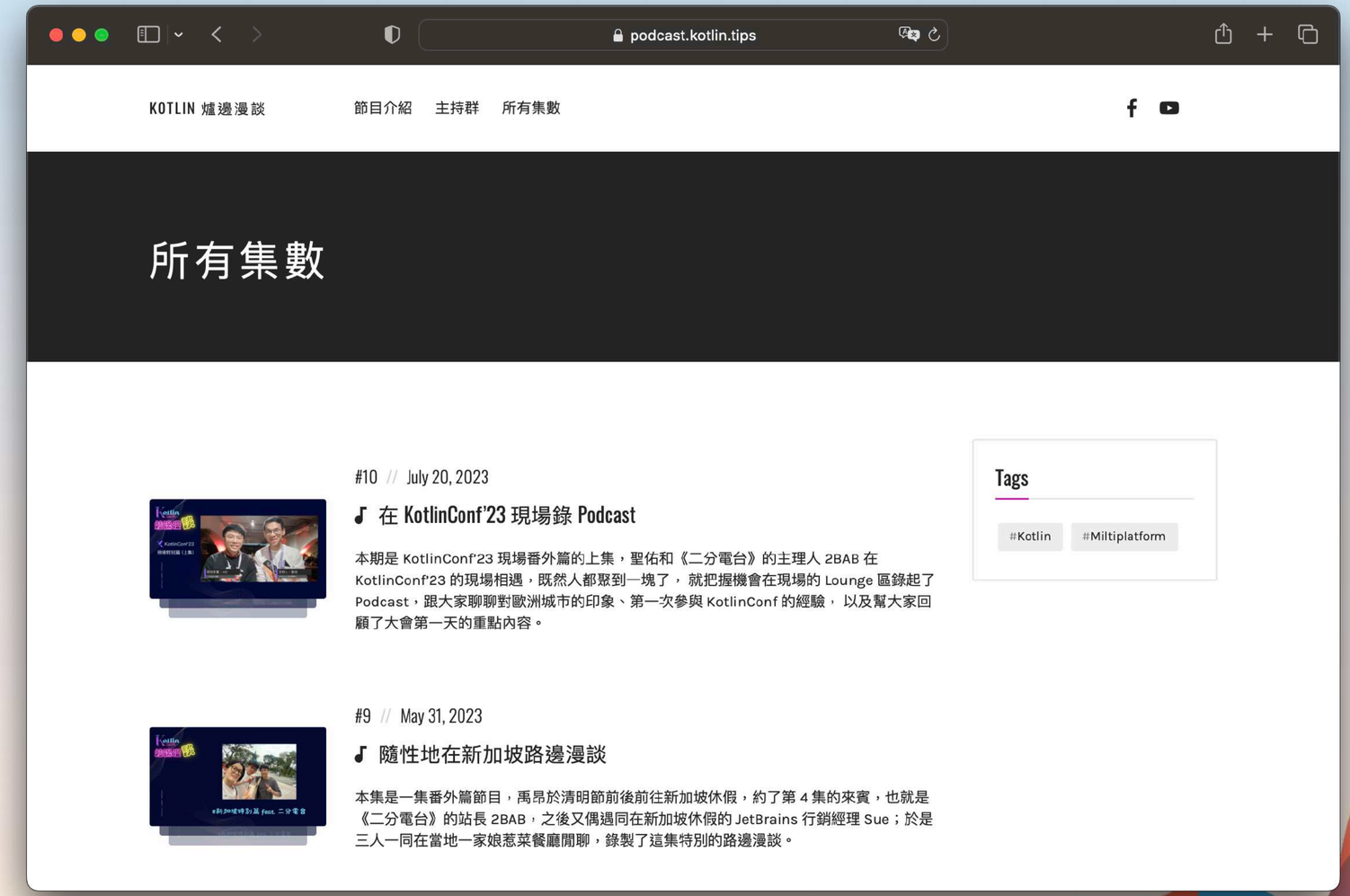

各平台播放次數統計圖
播放次數
1800
1600
1400
1200
1000
800
600
400
200
942
1 526
157
262
63
332
喜马拉雅
小宇宙
荔枝
猜姐
YouTube
00:03


各平台播放次數統計圖
第9集 2023-05-31 - 2023-06-09
播放次數
110
100
90
80
70
60
50
40
30
20
10
0
2023-05-31
2023-06-01
2023-06-02
2023-06-03
2023-06-04
2023-06-05
2023-06-06
2023-06-07
2023-06-08
2023-06-09
Youtube
BiliBili
喜马拉雅
小宇宙
荔枝
猜姐
00:04

欢迎收听并订阅

Kotlin 炉边漫谈播客

https://podcast.kotlin.tips/



想了解更多？

开源市集 (Open Source Bazaar)



开源社 kaiyuanshe

THANK YOU

QUESTIONS?

欢迎扫码打卡
积分可兑换对应礼品哟！

微信公众号：开源社KAIYUANSHE
视频号：开源社KAIYUANSHE
新浪微博：开源社
B站：开源社KAIYUANSHE

简书：开源社
头条：开源社
Facebook：KaiyuansheChina
Twitter：开源社KAIYUANSHE

扫码关注开源社公众号

扫码添加讲师联系方式