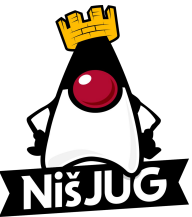




# Kotlin & Spring with Kotlin





# Introduction

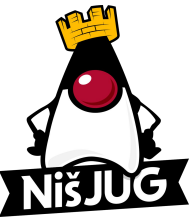
- ▶ Created by JetBrains
- ▶ First appeared in 2011
- ▶ First stable version in 2016
- ▶ Cross platform
- ▶ Supported by Google for Android development
- ▶ Support by most frameworks (Spring, Micronaut, etc) for backend development





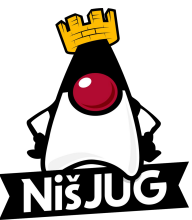
# ► Why was it created?

- ▶ Kotlin was invented to solve developer challenges of JetBrains
- ▶ There was need for simpler tool then Java
- ▶ Needed to work everywhere where Java works





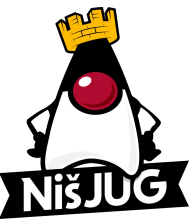
# Syntax introduction





# Syntax example

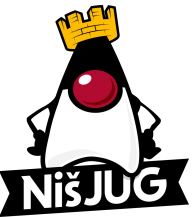
```
class DemoClass {  
    fun sayHello() {  
        println("Hello")  
    }  
    fun add(num1: Int, num2: Int) = num1 + num2  
    fun demo() {  
        val a = 5;  
        var b: Int = 5;  
        for (i in 1..10) {  
            println(i)  
        }  
    }  
}  
  
fun main(args : Array<String>) {  
    val demoClass = DemoClass()  
    demoClass.sayHello()  
    val result = demoClass.add(1,2)  
    println(result)  
    demoClass.demo()  
}
```





# Interchangeability with Java

- ▶ Kotlin and Java are 100% interoperable
- ▶ Call Kotlin code from Java
- ▶ Call Java code from Kotlin
- ▶ Kotlin and Java classes in same project side
- ▶ by side





# ▶ New things

- ▶ Data classes
- ▶ Top level functions
- ▶ Extension functions
- ▶ Smart cast
- ▶ Null safety
- ▶ Coroutines





# Data classes

```
public class PersonJava {  
    private int id;  
    private String name;  
  
    public PersonJava(int id, String name) {  
        this.id = id;  
        this.name = name;  
    }  
  
    public int getId() {  
        return id;  
    }  
  
    public void setId(int id) {  
        this.id = id;  
    }  
}
```

```
public String getName() {  
    return name;  
}  
  
public void setName(String name) {  
    this.name = name;  
}  
  
@Override  
public boolean equals(Object o) {  
    if (this == o) return true;  
    if (o == null || getClass() != o.getClass())  
        return false;  
    PersonJava that = (PersonJava) o;  
    return id == that.id &&  
        Objects.equals(name, that.name);  
}
```

```
@Override  
public int hashCode() {  
    return Objects.hash(id,  
        name);  
}  
  
@Override  
public String toString() {  
    return "PersonJava{" +  
        "id=" + id +  
        ", name='" +  
        name + '\'' +  
        '}';  
}
```

data class Person(var id: Int, var name: String)



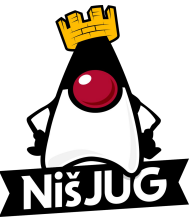




# ▶ Top level functions

```
fun addNumbers(num1: Int, num2: Int) = num1 + num2
```

```
fun sumNumbers(n: Int) : Int {  
    var s = 0  
    for (i in 1..n) {  
        s += i  
    }  
    return s  
}
```

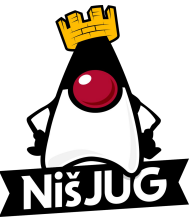




# ▶ Extensions

```
fun Int.percentOf(num: Int) = (num * this) / 100
```

```
fun main(args: Array<String>) {  
    println(7.percentOf(200))  
}
```

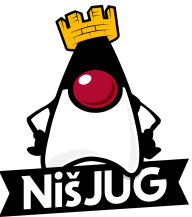




# Smart cast

```
Object obj = "This is some String";  
if(obj instanceof String) {  
    String str = (String) obj;  
    System.out.println("Length is " + str.length());  
}
```

```
val obj: Any = "This is some String"  
if(obj is String) {  
    println("Length is ${obj.length}")  
}
```

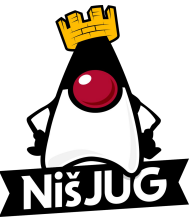




# Null safety

```
data class User(var id: Int, var name: String?)

fun test() {
    val user = User(1, null)
    val user1 = User(null, "User") // Compile error
    println(user.name.length) // Compile error
    println(user.name!!.length) // Throws exception
    println(user.name?.length) // Returns null
}
```

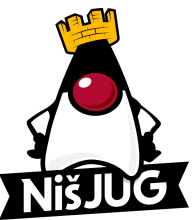




**BADIN**  
SOFT



# DEMO 1





# Kotlin & Spring



# ► Popular misconception

- Kotlin is only for Android

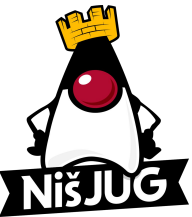
**NO!!!**





# Kotlin for backend

- ▶ Companies started using Kotlin for Backend before Google supported it for Android (Pinterest for example)
- ▶ Many backend frameworks support Kotlin
  - ▶ Micronaut
  - ▶ Spring
  - ▶ Vertx
  - ▶ Spark
  - ▶ .....

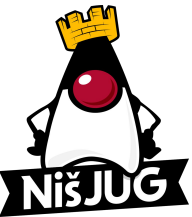






# ▶ Spring and Kotlin

- ▶ Spring supports Kotlin from the beginning
- ▶ Dedicated support since Spring 5



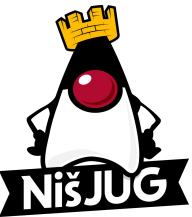


# Kotlin extensions in Spring

- ▶ Spring created extensions for many of its classes
- ▶ RestTemplate, Model....

```
model.addAttribute("title", "Blog");  
model.addAttribute("posts", postRepository.findAll());
```

```
model["title"] = "Blog"  
model["posts"] = repository.findAll()
```

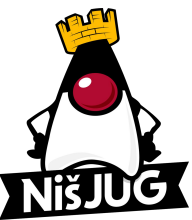




**BADIN**  
SOFT

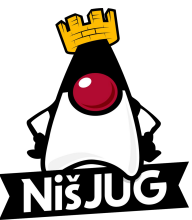


# DEMO 2



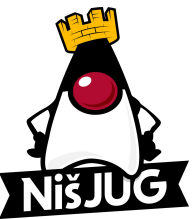
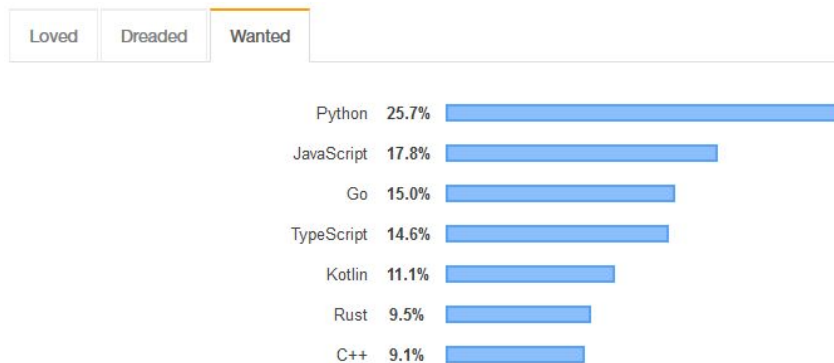
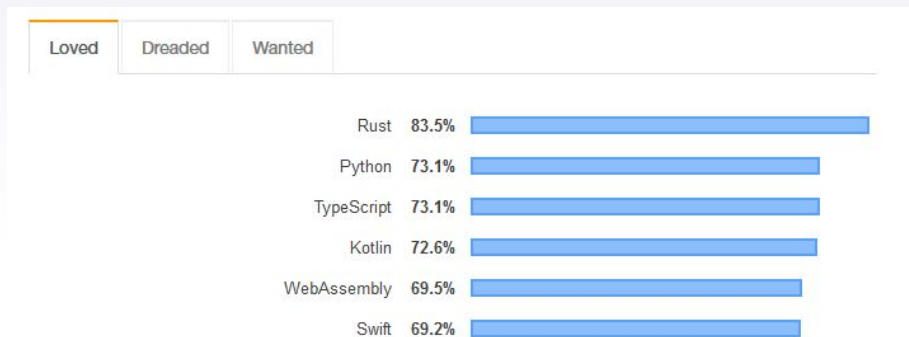


# Some statistics





# Stackoverflow 2019 survey



My linkedin:

- ◇ <https://www.linkedin.com/in/nemanja-petrovic-74b30678>

NisJug:

- ◇ [board@nisjug.org](mailto:board@nisjug.org)
- ◇ Twitter: @nisjug
- ◇ Meetup: NisJug
- ◇ Facebook: Java User Group Niš (J.U.G. Niš)
- ◇ Website: [www.nisjug.org](http://www.nisjug.org)

Badin soft:

- ◇ <https://www.badinsoft.com/>

