

# Pourquoi faire une application Android en Kotlin ?



# Plus concis

Java

```
public class Person {  
  
    private int age;  
    private String name;  
  
    public Person(int age, String name) {  
        this.age = age;  
        this.name = name;  
    }  
  
    public int getAge() {  
        return age;  
    }  
  
    public void setAge(int age) {  
        this.age = age;  
    }  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String name) {  
        this.name = name;  
    }  
}
```

Kotlin

```
data class Person(var name: String, var age: Int)
```

# Plus sûr

- Vérification d'erreur à la compilation

```
var text: String = "Hello World!"  
text = null  
print(text.length)
```

# Plus sûr

- Vérification d'erreur à la compilation

```
var text: String = "Hello World!"  
text = null  
print(text.length)
```

# Plus sûr

- Vérification d'erreur à la compilation

```
var text: String = "Hello World!"  
text = null  
print(text.length)
```

# Plus sûr

- Vérification d'erreur à la compilation

```
var text: String = "Hello World!"  
text = null  
print(text.length)
```

# Plus sûr

- Vérification d'erreur à la compilation

```
var text: String = "Hello World!"  
text = null // Compilation error  
print(text.length)
```

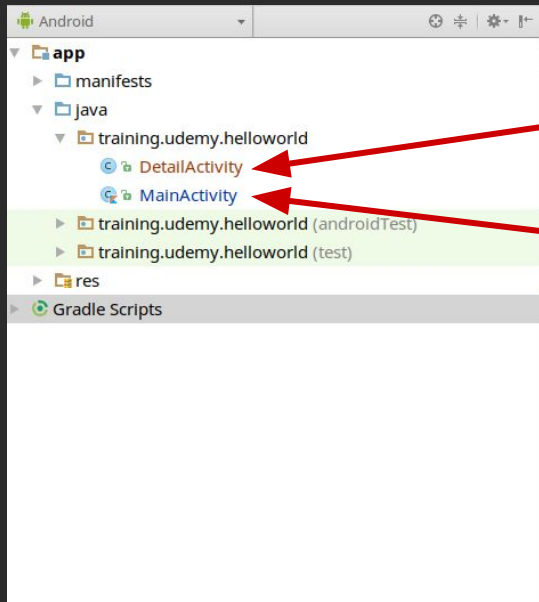
# Compatibilité forte

- Toutes les versions d'Android sont compatibles
- La large majorité des bibliothèques Java sont compatibles



# Compatibilité forte

- Rétro-compatible avec Java dans un même projet



Java

Kotlin

# Support officiel

- Kotlin est officiellement supporté par Google depuis la Google I/O 2017
- Excellent support dans Android Studio

