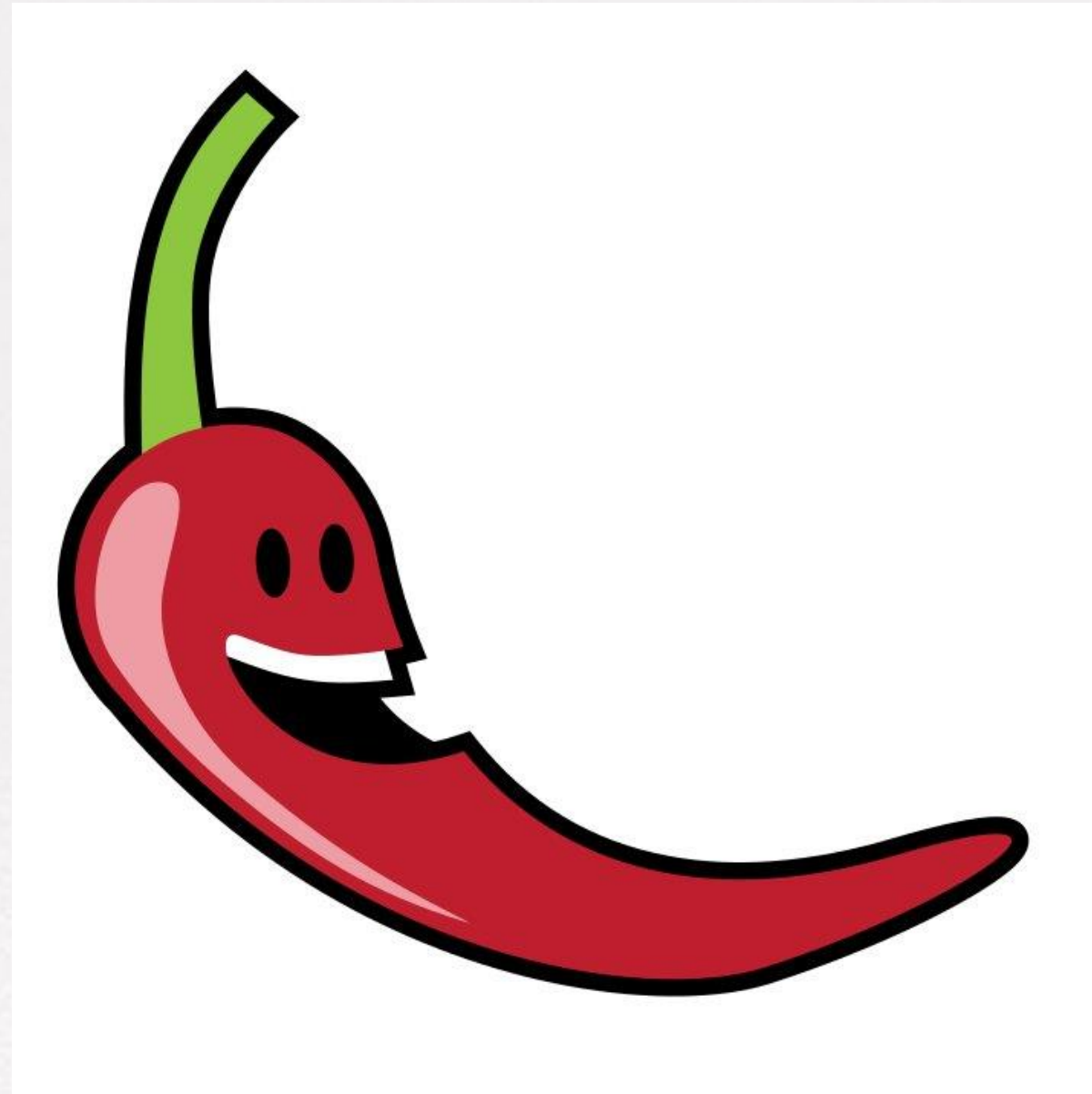
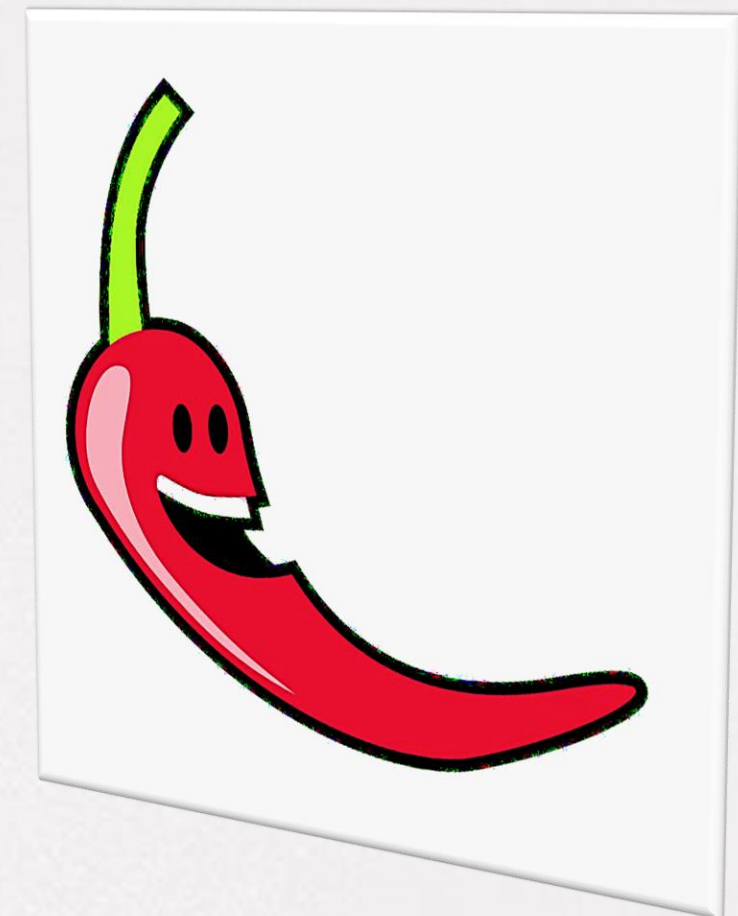


INTRODUCCIÓN A LOMBOK



- **Introducción a Lombok**
- Alternativas a Lombok
- Instalación
- Anotaciones
- Ejercicio práctico
- Cierre



QUÉ ES **LOMBOK**

1. Project Lombok
2. Biblioteca de Java
3. Conecta al editor/compilador
4. Facilita el desarrollo

QUÉ ES **LOMBOK**

5. Evita escribir código:
 - a. Getters/Setters
 - b. Equals
 - c. Constructor(es)
 - d. Variables/constantes

QUÉ ES LOMBOK

6. Elimina código repetitivo
7. Genera automáticamente
8. Integra con el propio IDE
 - a. Inyecta código
 - b. Al instante lo tiene el desarrollador

QUÉ ES **LOMBOK**

- 9. Ejemplo: “@Data”
 - a. Constructor
 - b. Getters/setters
 - c. equals
 - d. hashCode
 - e. toString

QUÉ ES LOMBOK

The image shows a screenshot of an IDE with two panes. The left pane displays the source code of a Java class named `Person` in the package `com.ocicweb.jnb.lombok`. The code uses Lombok annotations: `@Data` and `@AllArgsConstructor`. The class has private fields for `firstName`, `lastName`, `address`, `city`, `state`, `zip`, and `dateOfBirth`. The right pane shows the Outline view, which lists the package, import declarations, and the `Person` class with its methods and fields.

```
package com.ocicweb.jnb.lombok;

import java.util.Date;

import lombok.Data;

@Data
public class Person {
    private final String firstName;
    private final String lastName;
    private String address;
    private String city;
    private String state;
    private String zip;
    private final Date dateOfBirth;
}
```

Outline:

- com.ocicweb.jnb.lombok
 - import declarations
 - java.util.Date
 - lombok.Data
 - Person
 - Person(String, String, Date)
 - getFirstName() : String
 - getLastName() : String
 - getAddress() : String
 - setAddress(String) : void
 - getCity() : String
 - setCity(String) : void
 - getState() : String
 - setState(String) : void
 - getZip() : String
 - setZip(String) : void
 - getDateOfBirth() : Date
 - equals(Object) : boolean
 - hashCode() : int
 - toString() : String
 - firstName : String
 - lastName : String
 - address : String
 - city : String
 - state : String
 - zip : String
 - dateOfBirth : Date

INTRODUCCION A LA **INSTALACIÓN**

1. Tendremos una sección, pero emplearemos más Maven
2. Existen más herramientas
 - a. Maven
 - b. Gradle
 - c. Ivy
 - d. SBT
 - e. Java

INTRODUCCION A LA **INSTALACIÓN**

a. Maven

```
<dependencies>  
  <dependency>  
    <groupId>org.projectlombok</groupId>  
    <artifactId>lombok</artifactId>  
    <version>1.18.12</version>  
    <scope>provided</scope>  
  </dependency>  
</dependencies>
```

INTRODUCCION A LA **INSTALACIÓN**

a. Maven

`<version>1.18.12</version>`

<https://mvnrepository.com/artifact/org.projectlombok/lombok>

INTRODUCCION A LA **INSTALACIÓN**

b. Gradle \geq 2.12

```
compileOnly "org.projectlombok:lombok:1.18.12 "
```

INTRODUCCION A LA **INSTALACIÓN**

b. Gradle < 2.12

```
provided "org.projectlombok:lombok:1.18.12 "
```


INTRODUCCION A LA **INSTALACIÓN**

c. Ivy

```
<dependency  
  org="org.projectlombok"  
  name="lombok"  
  rev="1.18.12"  
  conf="build" />
```

INTRODUCCION A LA **INSTALACIÓN**

d. SBT

```
libraryDependencies += "org.projectlombok" % "lombok" % "1.18.12"
```

INTRODUCCION A LA **INSTALACIÓN**

e. Java

```
java -jar lombok.jar
```

<https://projectlombok.org/downloads/lombok.jar>

ALGUNAS DE LAS ANOTACIONES

- ✓ @NonNull
- ✓ @Cleanup
- ✓ @Getter/@Setter
- ✓ @ToString
- ✓ @EqualsAndHashCode
- ✓ @NoArgsConstructor
- ✓ @RequiredArgsConstructor
- ✓ @AllArgsConstructor

ALGUNAS DE LAS ANOTACIONES

- ✓ @Data
- ✓ @Value
- ✓ @Builder
- ✓ @SneakyThrows
- ✓ @Synchronized
- ✓ @With
- ✓ @Log