is a Lombok annotation that automatically generates a logger instance for the class.

### **What does @Slf4j do?**

* It injects a static final logger field named log into your class.
* The logger is of type org.slf4j.Logger.
* You can use this log object to write log messages (info, debug, error, etc.) without manually creating a logger.

Example Usage

If you have:

@Slf4j  
public class HomeController {  
 public void home() {  
 log.info("Home endpoint was called");  
 }  
}

You do not need to write:

private static final Logger log = LoggerFactory.getLogger(HomeController.class);

Lombok does it for you!  
**Why use it?**

* **Less boilerplate**: No need to declare and initialize the logger yourself.
* **Consistency**: Always uses the correct logger for the class.
* **Readability**: Cleaner code.

### **Requirements**

* You must have Lombok in your dependencies (which your project does).
* Your IDE should have Lombok support enabled.

**In summary:**@Slf4j is a Lombok annotation that gives you a ready-to-use log object for logging in your class, saving you from writing repetitive logger setup code.