

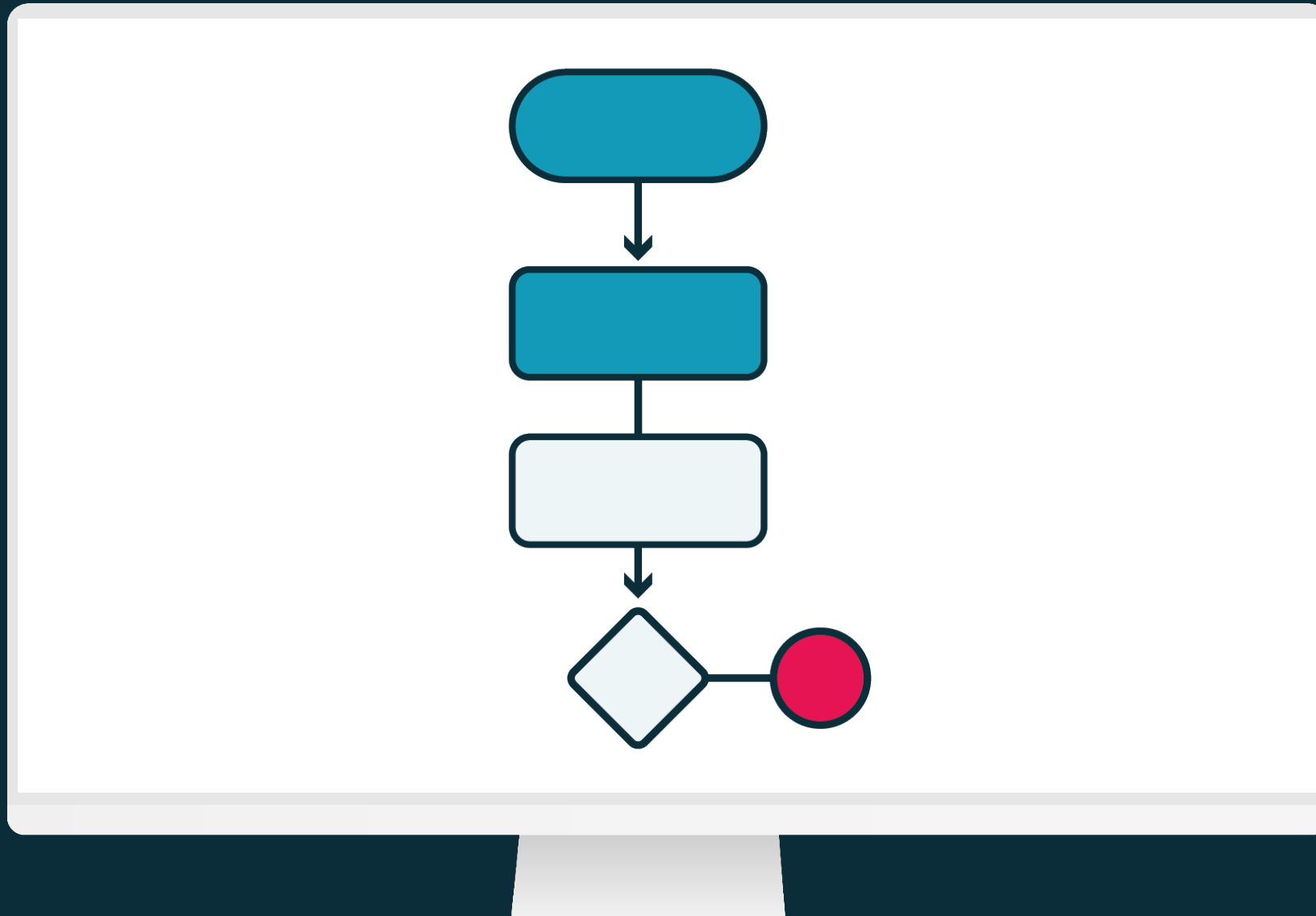
Low-Code trifft High-Code

Robert Loipfinger



**Werkzeuge sind
Optionen**

Low Code



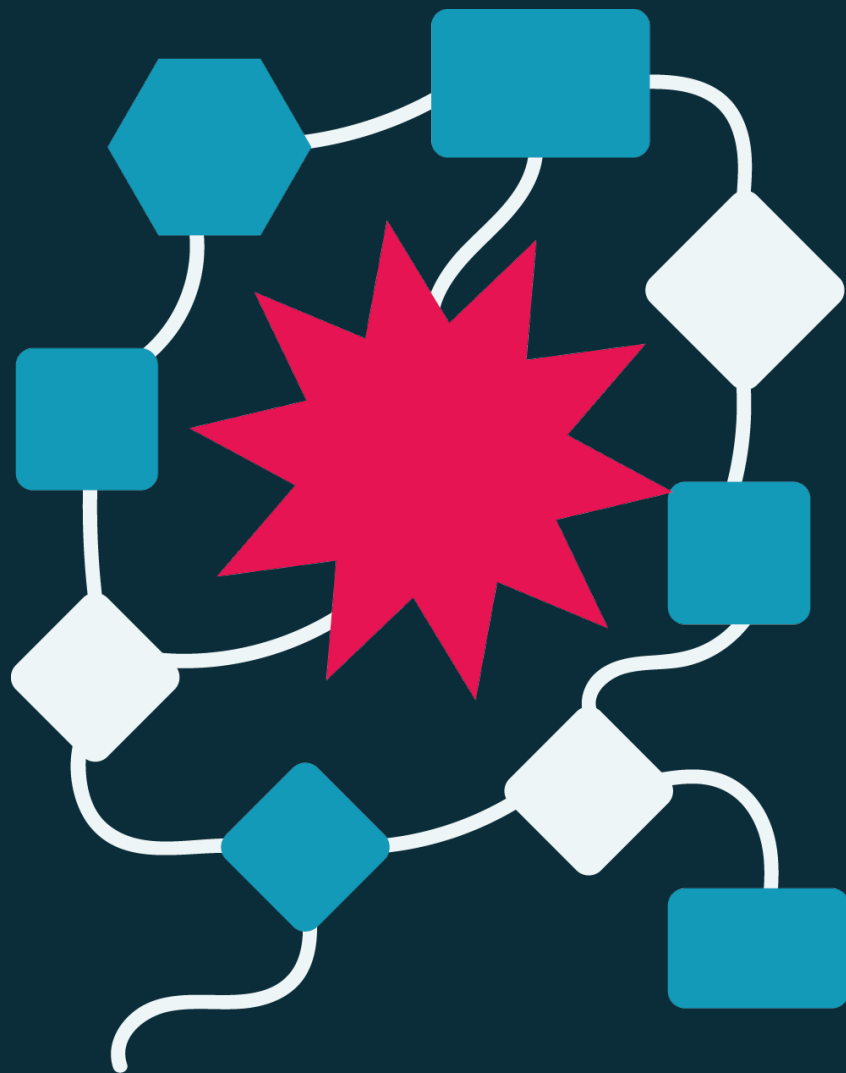
High Code

```
@Bean
public ExecutorService angebotExecutor() {
    ThreadFactory threadFactory = new ThreadFactoryBuilder()
        .setNameFormat("angebot-verifier-%d")
        .setDaemon(true)
        .build();

    return new ThreadPoolExecutor(
        4, 8,
        60L, TimeUnit.SECONDS,
        new LinkedBlockingQueue<>(100),
        threadFactory,
        new ThreadPoolExecutor.CallerRunsPolicy()
    );
}
```

Low-Code High-Code

Du



```
@Configuration
public class AngebotVerifierConfig {

    @Bean
    public ExecutorService anbotExecutor() {
        ThreadFactory threadFactory = new ThreadFactoryBuilder()
            .setNameFormat("angebot-verifier-%d")
            .setDaemon(true)
            .build();

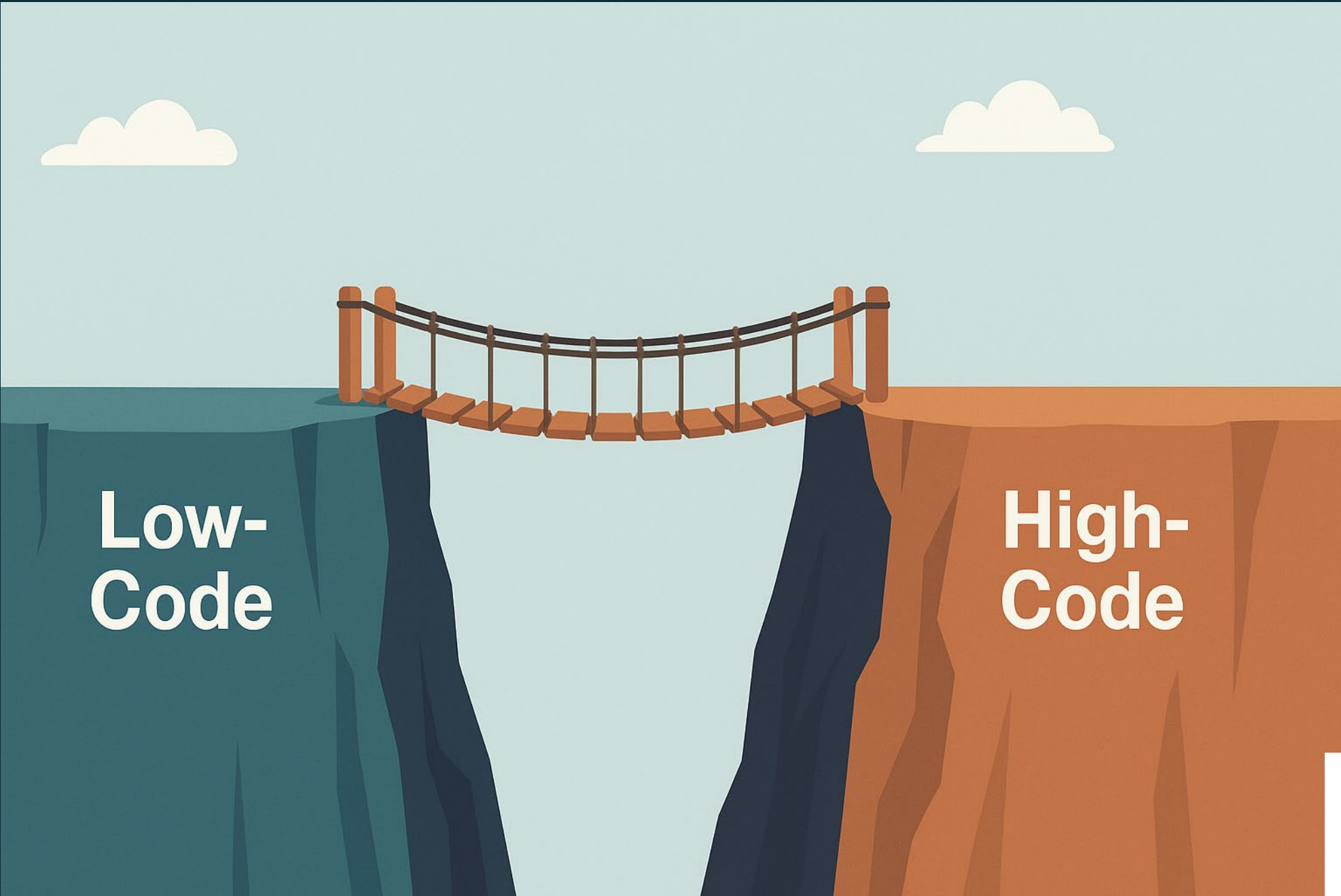
        return new ThreadPoolExecutor(
            4, 8,
            60L, TimeUnit.SECONDS,
            new LinkedBlockingQueue<>(100),
            threadFactory,
            new ThreadPoolExecutor.CallerRunsPolicy()
        );
    }
}

@Service
public class AngebotVerifier {

    private final ExecutorService executor;
    private final OracleClient oracle;
    private final Logger log;

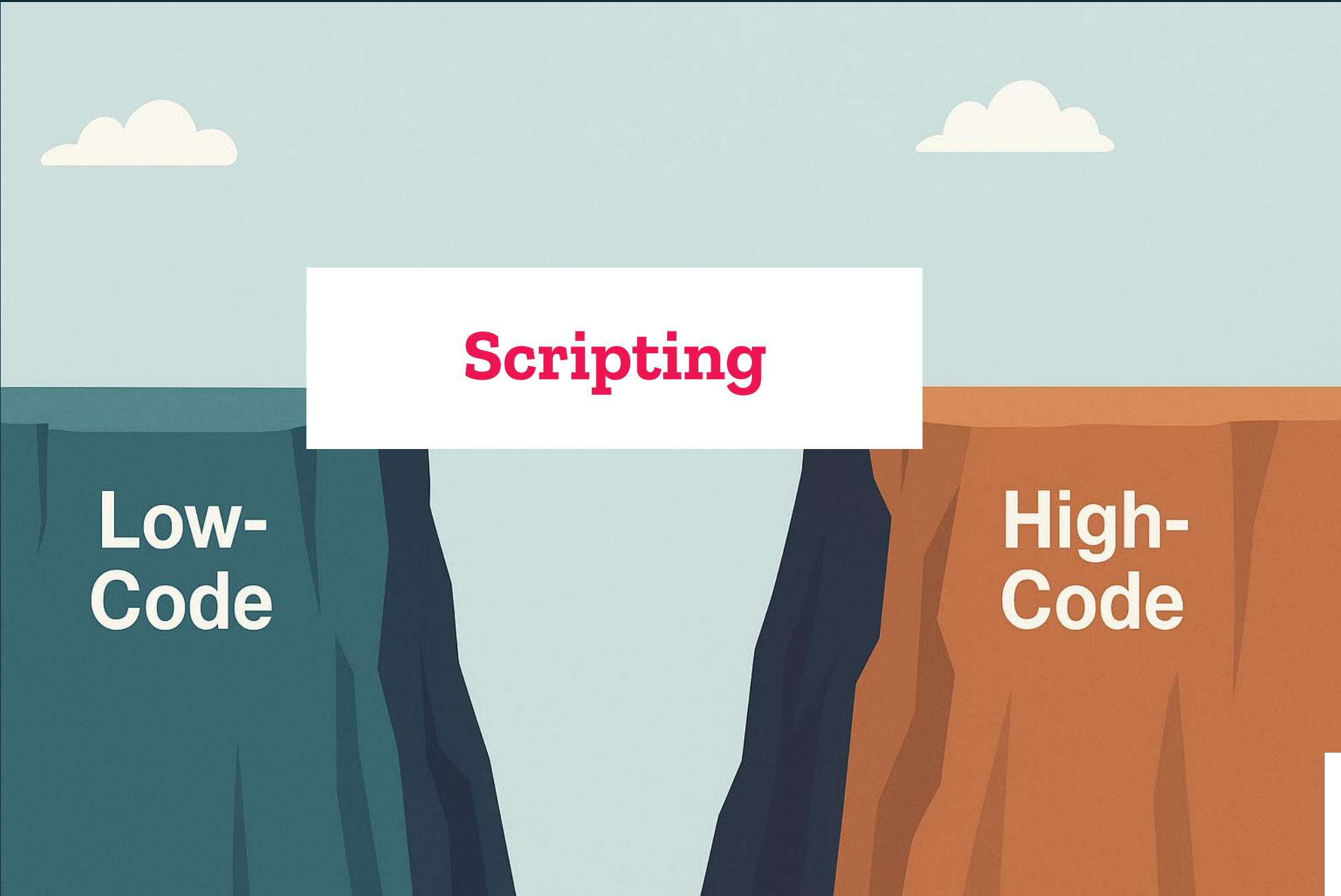
    public AngebotVerifier(ExecutorService executor, OracleClient oracle, Logger log) {
        this.executor = executor;
        this.oracle = oracle;
        this.log = log;
    }

    public void verify(Angebot anbot) {
        executor.submit(() -> {
            try {
                if (anbot.getBetrag() > 10000) {
                    anbot.setStatus(Status.VERIFY);
                    oracle.send("angebote", anbot);
                }
            } catch (Exception e) {
                log.warn("Fehler bei Angebot: " + anbot.getId(), e);
            }
        });
    }
}
```



**Low-
Code**

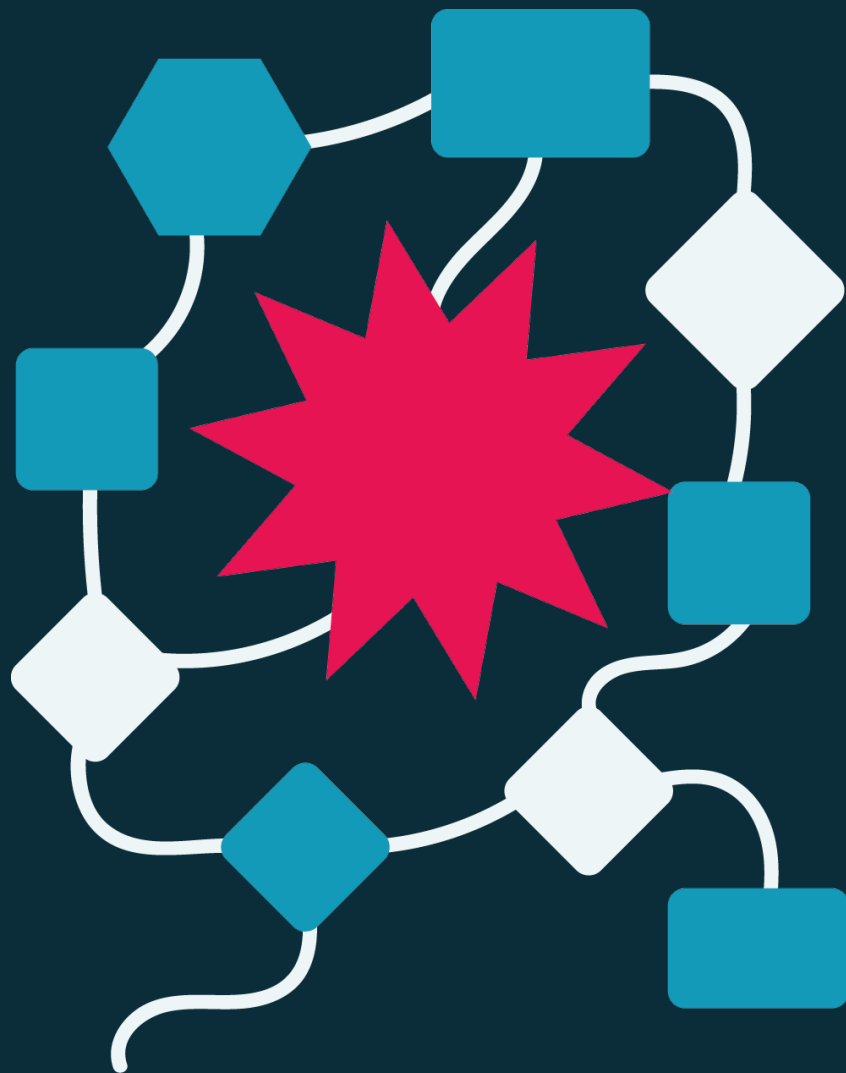
**High-
Code**



Scripting

Low-
Code

High-
Code



```
@Configuration
public class AngebotVerifierConfig {

    @Bean
    public ExecutorService anbotExecutor() {
        ThreadFactory threadFactory = new ThreadFactoryBuilder()
            .setNameFormat("angebot-verifier-%d")
            .setDaemon(true)
            .build();

        return new ThreadPoolExecutor(
            4, 8,
            60L, TimeUnit.SECONDS,
            new LinkedBlockingQueue<>(100),
            threadFactory,
            new ThreadPoolExecutor.CallerRunsPolicy()
        );
    }
}

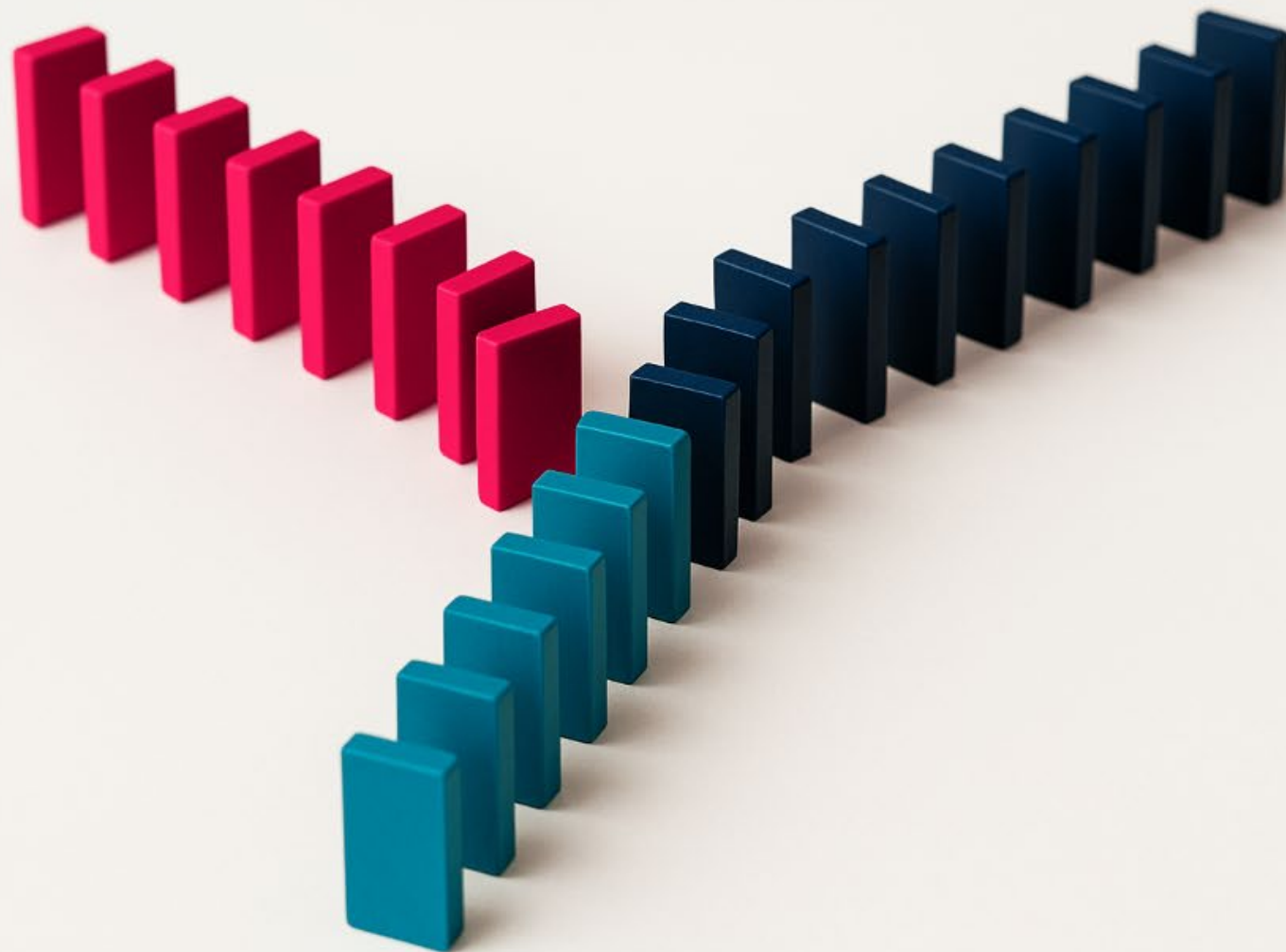
@Service
public class AngebotVerifier {

    private final ExecutorService executor;
    private final OracleClient oracle;
    private final Logger log;

    public AngebotVerifier(ExecutorService executor, OracleClient oracle, Logger log) {
        this.executor = executor;
        this.oracle = oracle;
        this.log = log;
    }

    public void verify(Angebot anbot) {
        executor.submit(() -> {
            try {
                if (anbot.getBetrag() > 10000) {
                    anbot.setStatus(Status.VERIFY);
                    oracle.send("angebote", anbot);
                }
            } catch (Exception e) {
                log.warn("Fehler bei Angebot: " + anbot.getId(), e);
            }
        });
    }
}
```

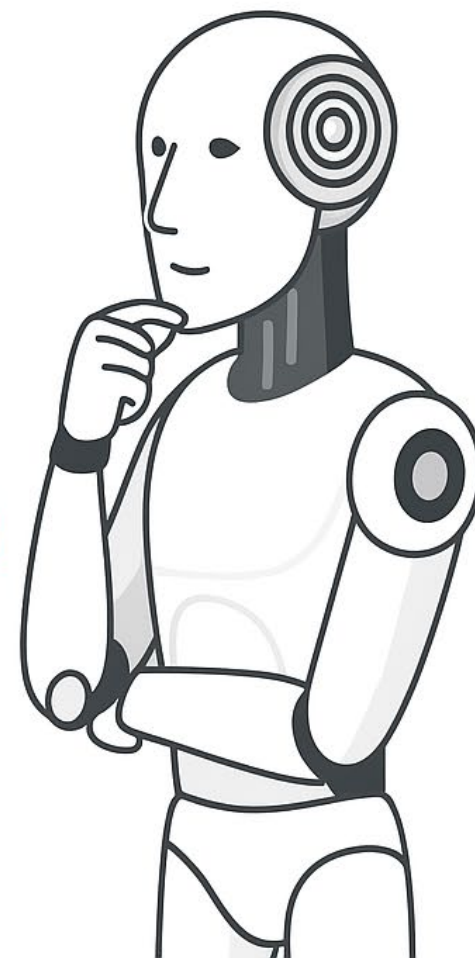
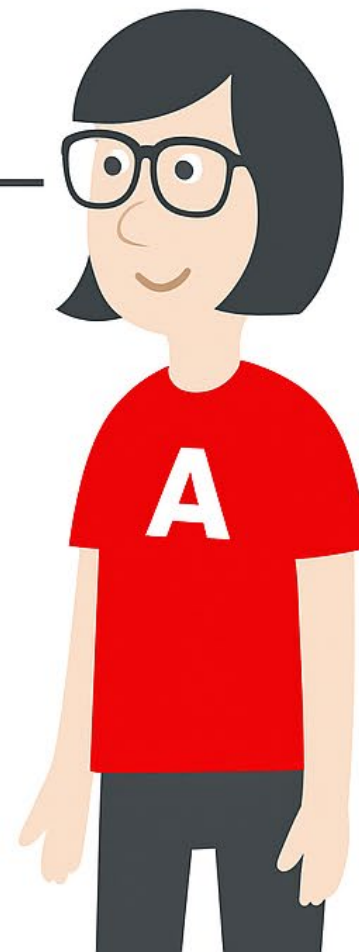
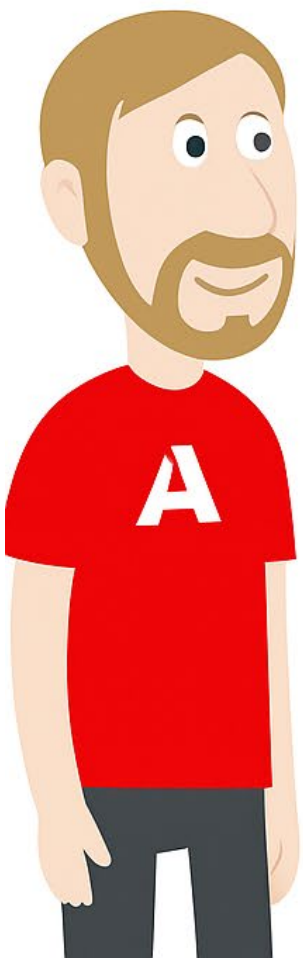
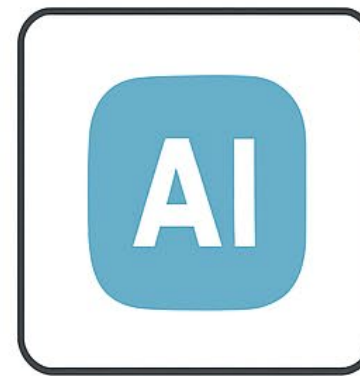
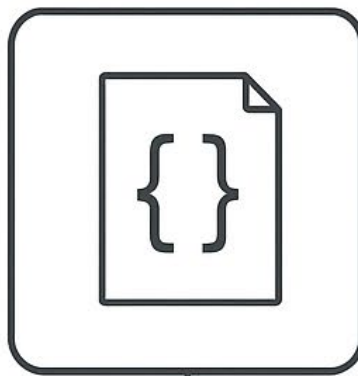
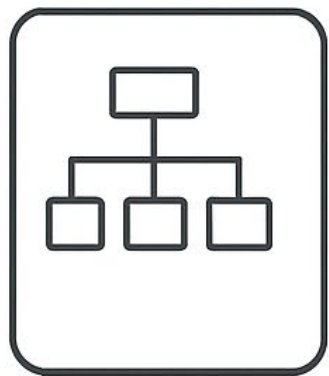
```
if (angebot.typ === 'Sonderfall' && kunde.plz.startsWith("8")) {  
    anbot.status = 'PRÜFEN';  
} else if (angebot.betrag > 10000 && !kunde.vip) {  
    anbot.status = 'ESCALATE';  
} else if (angebot.rabatt > 20) {  
    anbot.status = 'CHECK_RABATT';  
} else {  
    anbot.status = 'OK';  
}
```



**Und was denkt die
KI darüber?**







**Werkzeuge sind
Optionen**

