WELCOME BACK
&
THANK YOU

Advance Java
Crash Course

For TD Bank

MEET YOUR CRASH COURSE TEAM

TANGY F.

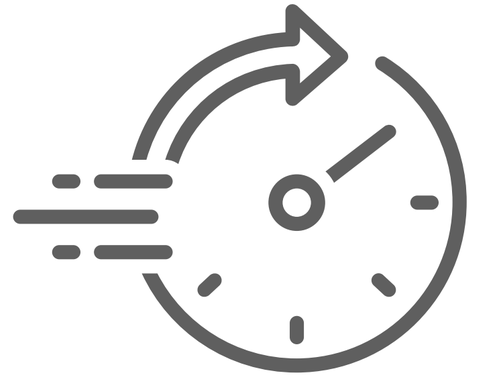
CEO

WILLIAM D.

DEVELOPER

Rapid Review

TRIVIA

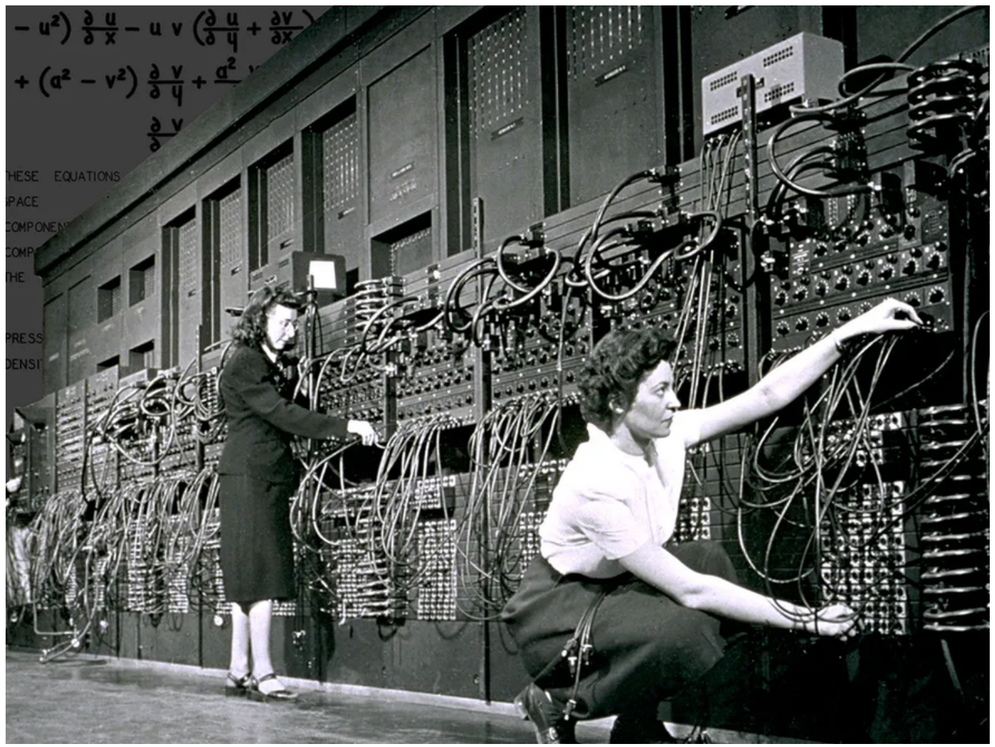


Rapid Trivia

What was the name of the first digital computer?

Rapid Review

TRIVIA



ENIAC Electronic Numerical Integrator and Computer

TRAINING DAY 10

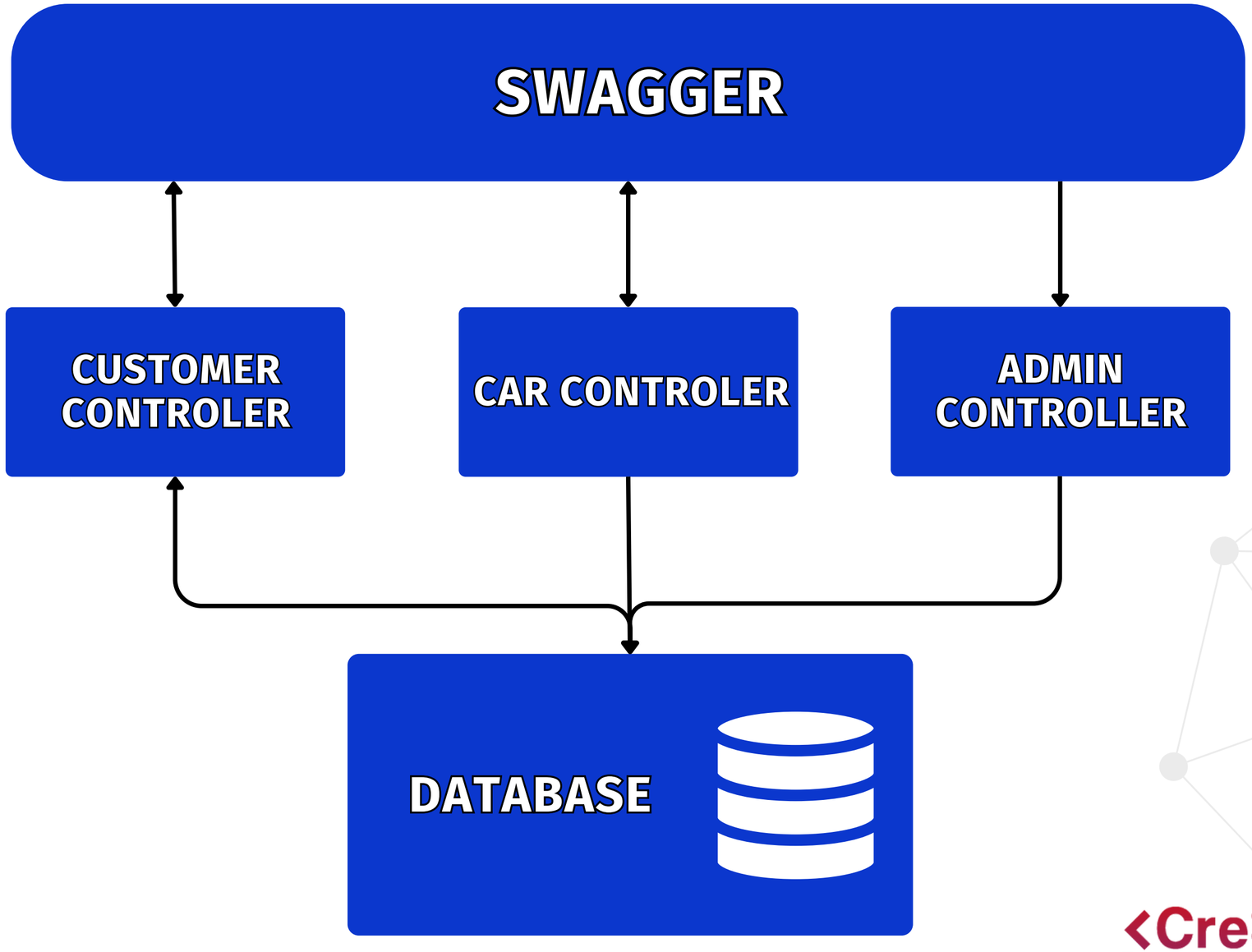
TODAY'S AGENDA

1. Yesterday's Coding Exercise to Go Rapid Review
2. 12 Factor App
3. Case Study. Using Java Reflection in a Corporate EDI Framework
4. Post Assessment Questions

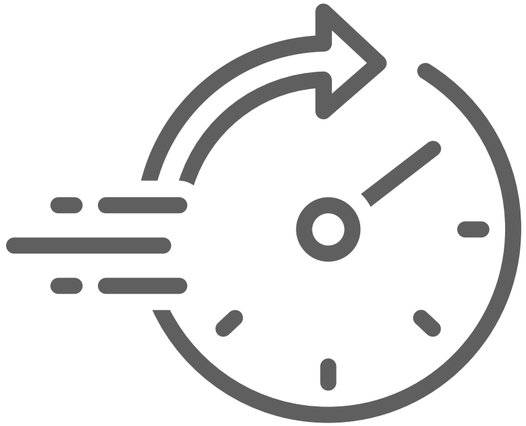
Creative Devs. Software

BITE SIZE
CAR RENTAL APP

WE ARE BUILDING



Rapid Review
Lesson 1



Rapid Review

Add Introspection to car-controller /api/cars/{carId}

CODING EXERCISE TO GO

In the **car-controller /api/cars/{carId}** change the line bellow to call method **getCarById(carId)** via introspection. Use a String variable to store **"getCarById"**

```
CarResponseDto car = carService.getCarById(carId);
```

GET /api/cars/{carId}

YESTERDAY'S CODING TO GO REVIEW

```java
@GetMapping("/{carId}")
public ResponseEntity<CarResponseDto> getCarById(@PathVariable Long carId) throws Exception {
    Class<?> clazz = CarService.class;
    String methodName = "getCarById";
    Method method = clazz.getDeclaredMethod(methodName, Long.class);
    CarResponseDto car = (CarResponseDto) method.invoke(carService, carId);

    //CarResponseDto car = carService.getCarById(carId);
```

DAY 10
LESSON 2

12 Factor App

12 FACTOR APP

LESSON 9

1. **Codebase**

>>Version control best practice

Example: Git

2. Dependencies

>>Using a dependency management system like Maven or Gradle

>>Spring Boot provides dependency management plugins

3. Config

>>Have all configuration in properties or any YAML file instead of hard coding them.

>>Using environmental variable and external config files

12 FACTOR APP

LESSON 9

4. Backing Services

>>DB or external services are considered as backing services

>>Spring boot provides integration modules and libraries such as JPA, Hibernate for connection

5. Build, Release, Run

>>Use Maven or Gradle for build application into executable JAR or WAR files

>>Spring supports these executable file to easily run and deploy.

12 FACTOR APP

LESSON 9

6. Processes

>>Designing a stateless process and having the data store in DB or in Caches

>>Spring supports the DB storage and external cache service

7. Port Binding

>>Bind the services or Rest applications to specific port

>>In spring we can adjust the port number for Tomcat or Jetty

8. Concurrency

>>Scaling the applications horizontally

>>Spring gives the content caching or distributed caching to handle concurrency

12 FACTOR APP

LESSON 9

9. Disposability

>>Ensure fast startup and graceful shut down of application

>>Spring provides lifecycle hooks and shutdown events to cleanup activity.

10. Dev/ Prod Parity

>>Maintaining all the environments as possible.

>>Spring Profiles can be used to manage environment specific configs.

11. Logs

>>Having logging enabled for the application. It helps in monitoring and trouble shooting.

>>Spring Boot has a built-in logging framework like Log4j

12. Admin Processes

>>One-off processes should be performed using the same environment and tools as the production environment.

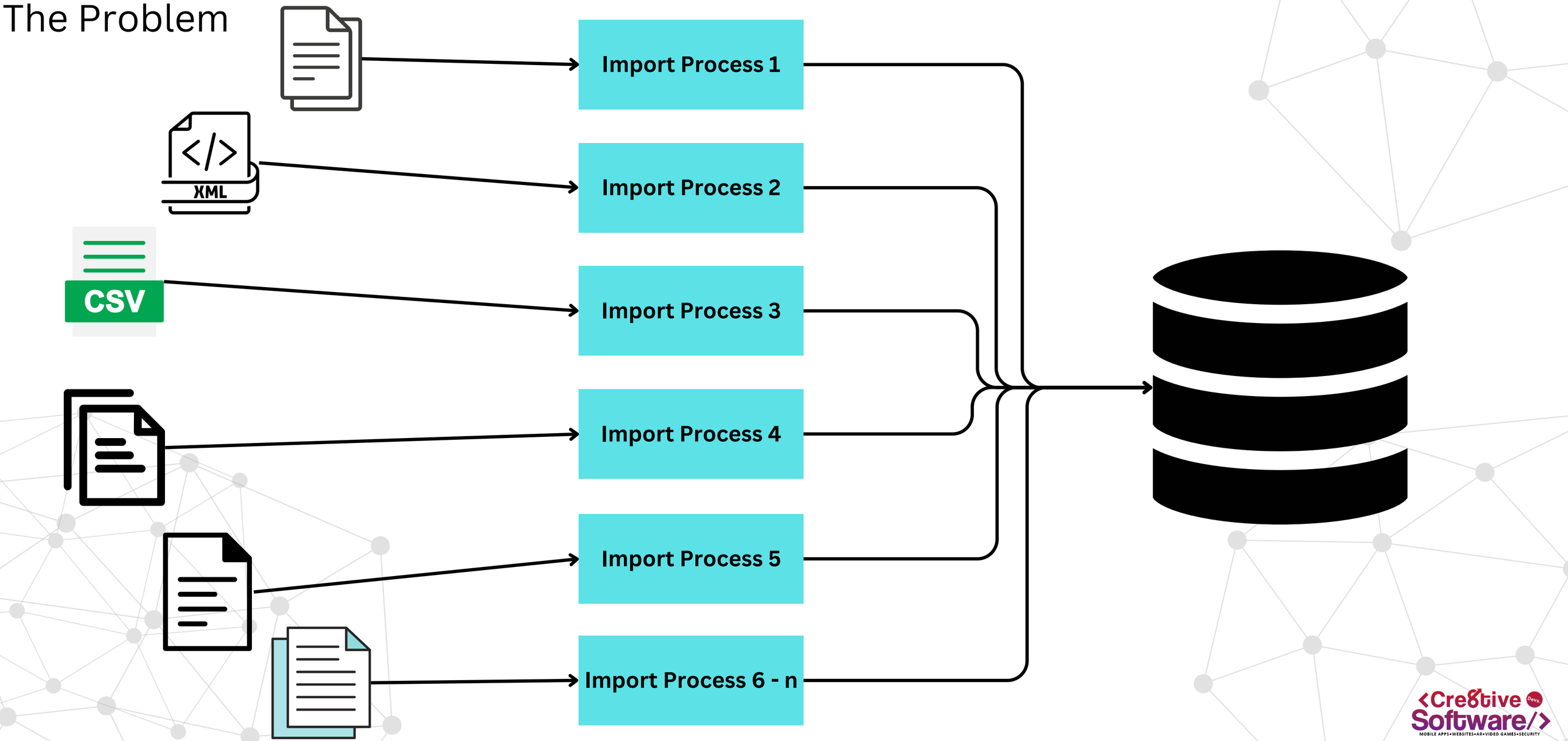
>>Spring boot features like Spring Batch and scheduling tasks to execute the administrative processes.

DAY 10
LESSON 3

Case Study. Using Java Reflection in a Corporate EDI Framework

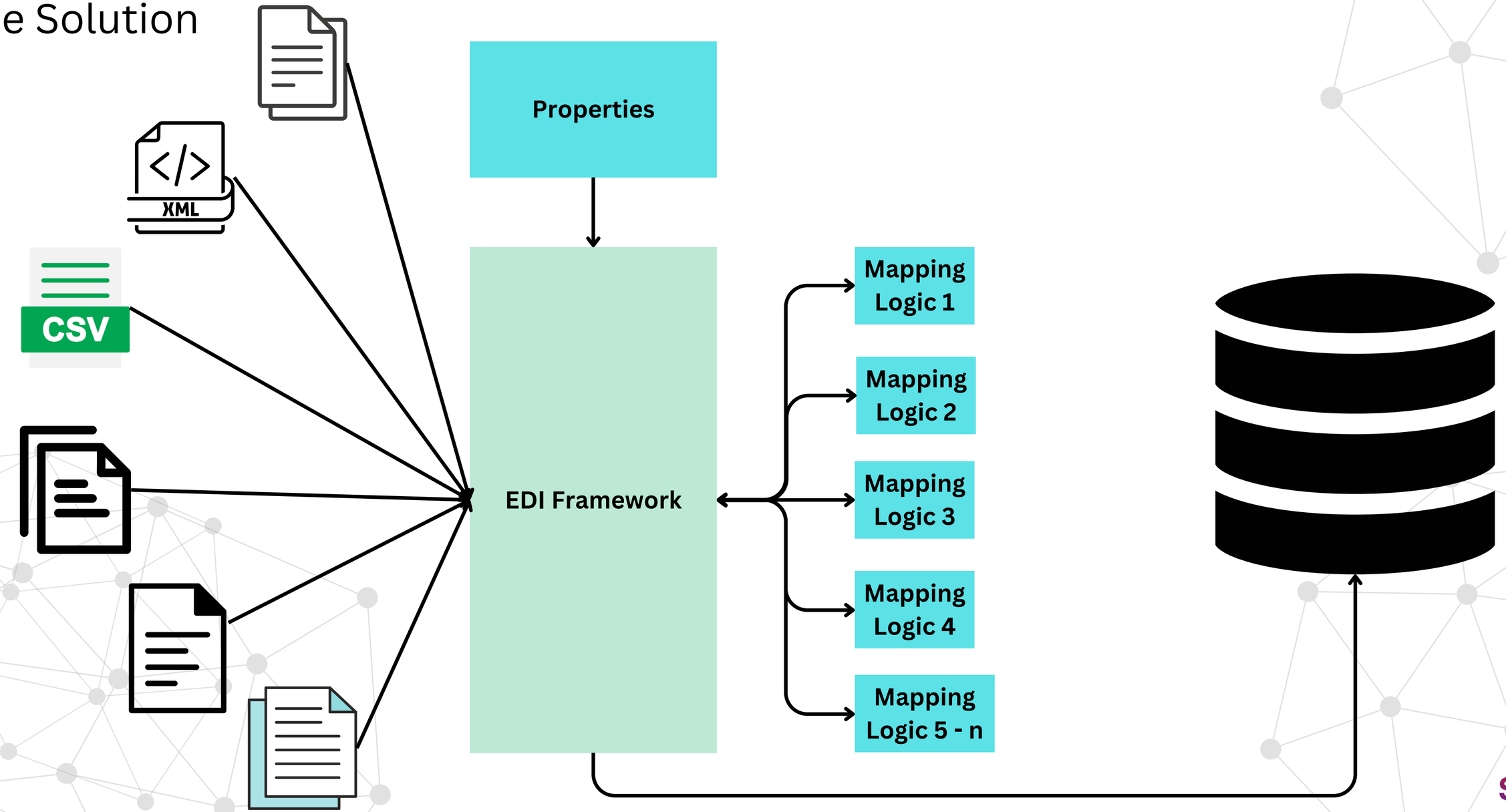
CASE STUDY USING JAVA REFLECTION

The Problem

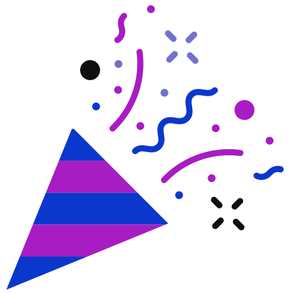


CASE STUDY USING JAVA REFLECTION

The Solution



LESSON 10



Type your answers on a document & email them to us.

QUESTIONS 1 - 3

LESSON 10

1. We are getting a compile error, it seems like a dependency is missing. What is the Gradle command that we should use to know for sure?
2. What should we do with the the **bookVehicleDto** object when we are sure we no longer need it and we want the garbage collector to remove it from memory.
3. Which of the 2 approaches bellow will yield the best performance, the first or the second.

```
// First Logic
String value = "";
 for (int i = 0; i < n; i++) {
    value += i;
 }

// Second Logic
StringBuilder stringBuilder = new StringBuilder();
 for (int i = 0; i < n; i++) {
stringBuilder.append(i);
}
```

QUESTIONS 4

LESSON 10

4. We know **Cat** extends **Domestic**, and **catList** is a **List** of **Cat**. If we have the call **setDomestic(catList)**; which of the implementations bellow will NOT give a compilation error to such call.

```
// Implementation 1
private void setDomestic(List <? extends Domestic> domesticList);

// Implementation 2
private void setDomestic(List <Domestic> domesticList);
```

QUESTIONS 5 - 6

LESSON 10

5.The Lambda expression bellow is not compiling, what is wrong with it?

```
FunctionalCalculator functionalCalculator -> (num1, num2) = num1 * num2;
```

6.We are getting a **ClassNotFoundException** when we run the code bellow despite “**TheClass**” existing in the package. What is missing?

```
String className = “TheClass”;
Class<?> clazz = Class.forName(className);
    ClassInterface instance = (ClassInterface) clazz.getDeclaredConstructor().newInstance();
```

HEARTFELT GRATITUDE
FOR YOUR PARTICIPATION
FROM
CRE8TIVE DEVS

<Cre8tive Devs. Software/>

Crash Course
Spring/ Spring Boot
Cohort 1

CERTIFICATE
OF
COMPLETION

THANK YOU

Crash Course

We will see you tomorrow