

# C O D E \_ F A C T Ø R Y ( D K B ) ;

## Bank Account Toy Coding Challenge

Thanks for taking the time and looking into DKB Code Factory as your next potential employer. Here's a small code challenge that should give you a rough idea on what you'd might work on in the banking industry.

### ***The Task:***

Create a small application and design a REST API with the following prerequisites and use cases. Some tasks are optional, feel free to decide if you want to work on them.

### ***Prerequisites:***

We have an existing bank customer with money in the following accounts:

- Checking account (functions as a reference account for the savings account)
- Savings account
- Private loan account

Every account has an IBAN assigned and should be referenced by this.

All accounts are having following capabilities:

**Checking account** - transferring money from and to any account is possible

**Savings account** - transferring money from any account is possible. Only transferring money from the savings account to the reference account (checking account) is possible.

**Private loan account** - transferring money from any account is possible. Withdrawal is not possible

### ***Use cases:***

#### ▸ **Deposit money into a specified bank account**

- Enable adding some amount to a specified bank account

#### ▸ **Transfer some money across two bank accounts**

- Imagine this like a regular bank transfer. You should be able to withdraw money from one account and deposit it to another account

#### ▸ **Show current balance of the specific bank account**

- Show the balance of a specified bank account

#### ▸ **Filter accounts by account type**

- Request account information by account type (could be multiple)

#### ▸ **Show a transaction history**

- For an account, specified by IBAN, show the transaction history
- **\*Bonus - account creation**
  - To open an account and assign it with an IBAN an endpoint should be provided.
- **\*Bonus - account locking**
  - For security reasons, it might be a good idea to be able to lock a specific account. For example, if an internal fraud management system spots something suspicious, it should be able to lock that account. Naturally, if the account can be locked, there should be an unlock functionality in place

#### **Extra information:**

- Think about how to organise account data, and be prepared to explain why you've chosen this approach. Basically the application should be stateless, except that.
- You should implement your application in one of the JVM languages - preferred are Java or Kotlin.
- All the specified functionality must be available through a REST API.
- Spring-Boot framework should be used.
- You have your hands free, meaning you choose what build tools, libraries, etc. you want to use.
- Although it's a simple API, it should be production-ready. Everybody has a different opinion on what that means so we're looking forward to speak about that with you.

#### **Deliverables:**

Your coding challenge solution as a zip archive or a link to a public repository with all the necessary files you think need to be provided when thinking about a production-ready API solution.

**In case you have additional questions, feel free to contact us. We're looking forward to receive your solution.**