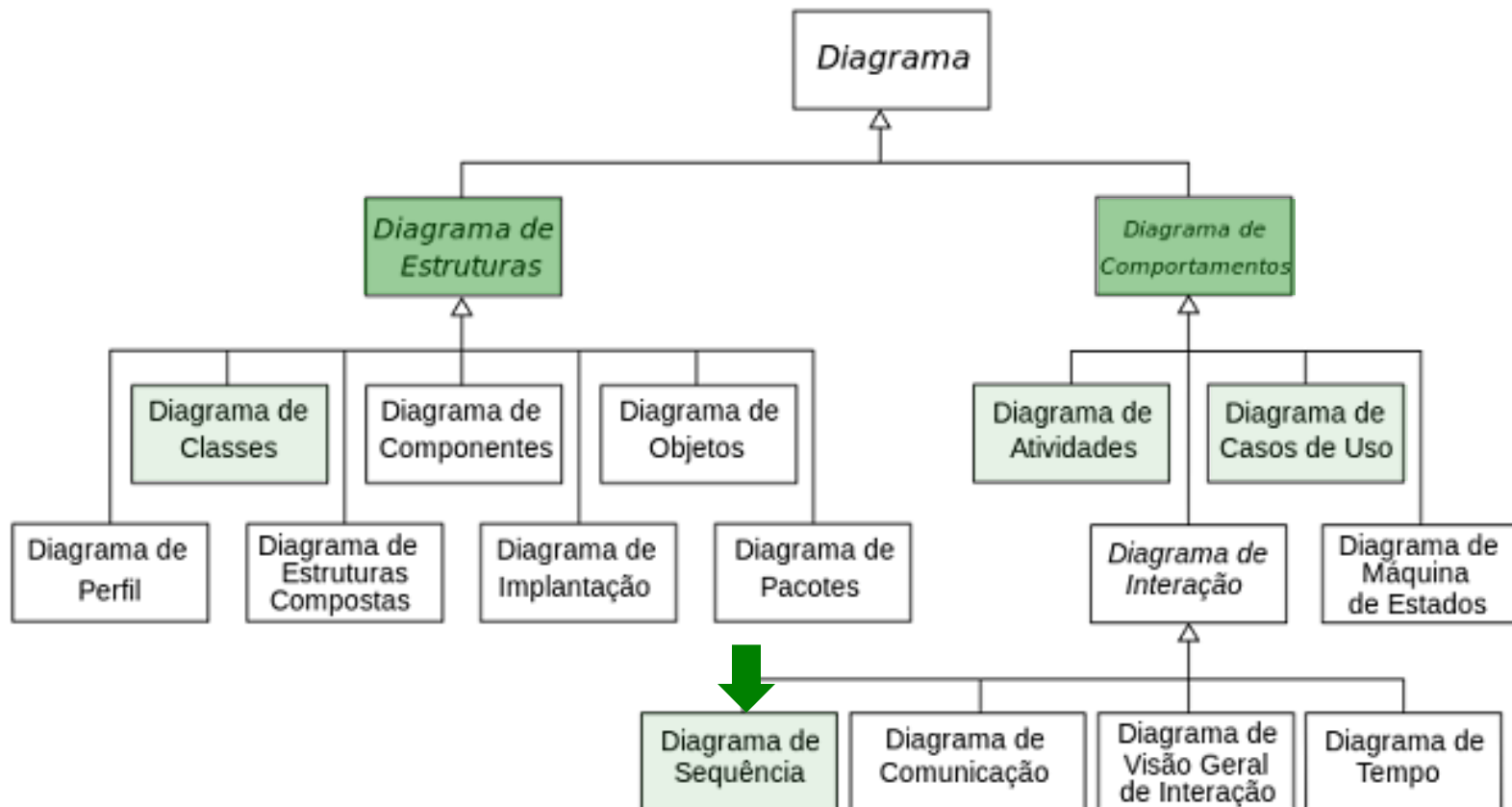


Curso: Análise e Desenvolvimento de Sistemas
Disciplina: Análise e Projeto Orientado a Objetos

Profª Drª Marcia Cassitas Hino

1º Semestre/2024

● ONDE ESTAMOS



Categoria	Diagramas	Descrição
Diagramas Dinâmicos	Casos de uso	Expressam a funcionalidade de um sistema
	Atividades	Representam o fluxo de atividades dos processos de negócio
	Interatividade	Apresenta um fluxo de atividades, mostrando como elas trabalham em uma sequência de eventos
	Sequência	Define a ordem e a troca das mensagens entre objetos
	Comunicação	Representa o diagrama anterior de colaboração
	Máquina de estados	Representa as ações ocorridas em resposta ao recebimento de eventos
	Temporal	Mostra mudança de estado de um objeto
Diagramas Estruturais	Classes	Apresenta elementos conectados por relacionamentos
	Objetos	Apresenta objetos e valores de dados
	Componentes	Mostra dependências entre componentes de software
	Pacotes	Usado para organizar elementos de modelos e mostrar dependências entre eles
	Implantação	Mostra a arquitetura do sistema em tempo de execução, as plataformas de software, etc.
	Estrutura composta	Usado para mostrar a composição de uma estrutura complexa

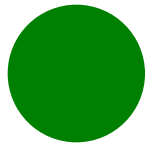
● Diagrama de Sequência



**Que
informações
faltam no
diagrama de
classes?**



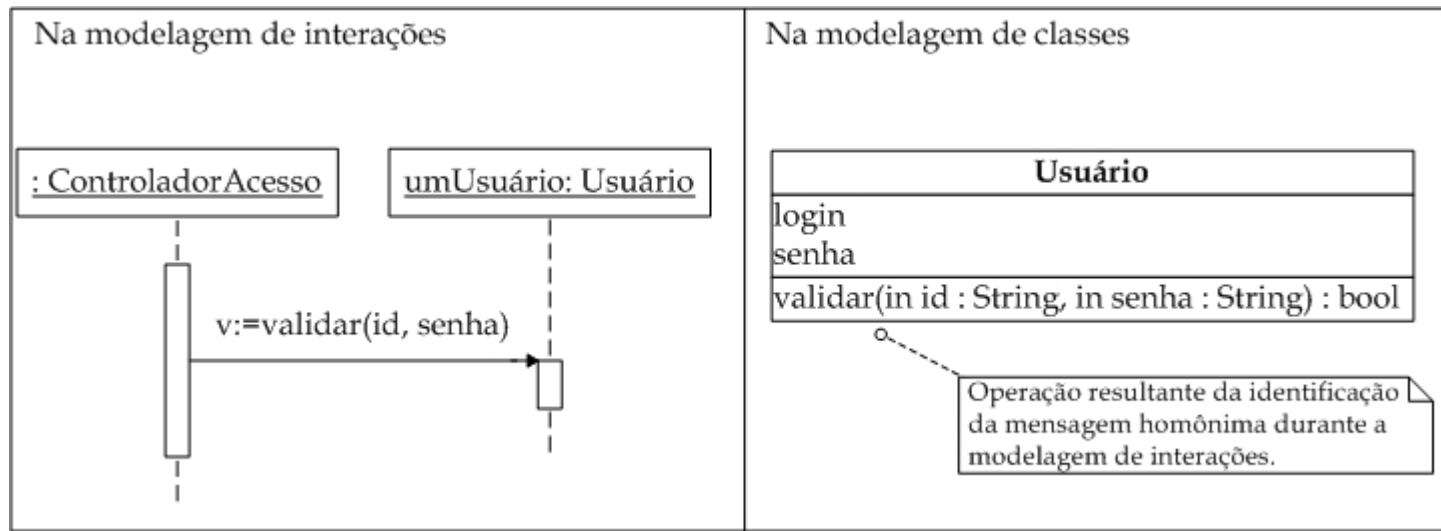
- ◎ De que forma os objetos colaboram para que um determinado caso de uso seja realizado?
- ◎ Em que ordem as mensagens são enviadas?
- ◎ Que informações precisam ser enviadas em uma mensagem de um objeto a outro?

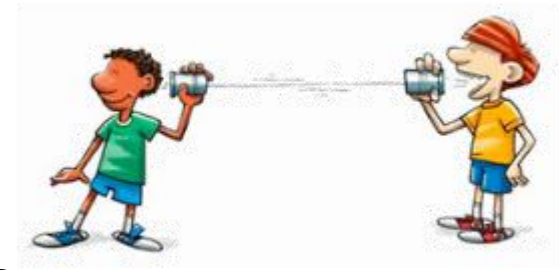


MODELO DE INTERAÇÃO

- Representa mensagens trocadas entre objetos para a execução de cenários dos casos de uso do sistema
- E tem como objetivos:
 - Obter informações adicionais para completar e aprimorar outros modelos (principalmente o modelo de classes)
 - Fornecer uma visão detalhada dos objetos e mensagens envolvidos na realização dos casos de uso

Os diagramas de interação buscam identificar mensagens trocadas entre as classes.





- Uma mensagem implica na existência de uma operação no objeto receptor. É a especificação de uma comunicação entre objetos, onde são passadas informações, com a esperança de que ocorra alguma atividade.
- A resposta do objeto receptor ao recebimento de uma mensagem é a execução da operação correspondente.
- Na maioria das vezes, uma mensagem resulta na execução de uma operação.



Há três tipos de diagrama de interação:

- Diagrama de sequência
- Diagrama de comunicação
- Diagrama de visão geral da interação

Diagrama de Sequência:

- 
- Têm foco nas mensagens enviadas no decorrer do tempo
- 
- 

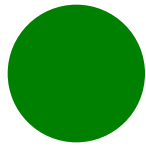


DIAGRAMA DE SEQUÊNCIA

- Mostra um **conjunto de objetos, seus relacionamentos e as mensagens** que podem ser enviadas entre eles
- Baseia-se nos casos de uso e diagrama de classes
- São utilizados para representar e modelar o **fluxo de mensagens**, eventos e ações entre objetos e componentes de um sistema. Descreve a **sequência de mensagens** que devem ser trocadas para realizar um determinado cenário

São úteis para modelar os seguintes cenários:

- ❑ Complexas interações entre os sistemas: bastante útil quando componentes estão sendo desenvolvidas em paralelo
- ❑ Validação dos casos de uso: auxílio visual que permite uma maior clareza do que está escrito
- ❑ Modelar sistemas distribuídos: troca de mensagens entre os diferentes sistemas
- ❑ Classes: valida se todas as operações das classes foram declaradas
- ❑ Documentação: permite documentar a dinamicidade do sistema, o que é difícil em diagramas estáticos como o de classe

- Diagramas de Sequência enfatizam a ordenação das mensagens trocadas entre os objetos.
- Um **cenário** é uma sequência específica de ações que ilustra um comportamento.
- Diagramas de Sequência podem modelar **apenas um cenário** ou um **conjunto de cenários**.
- Diagramas de Sequência podem mostrar decisões simples e iterações.

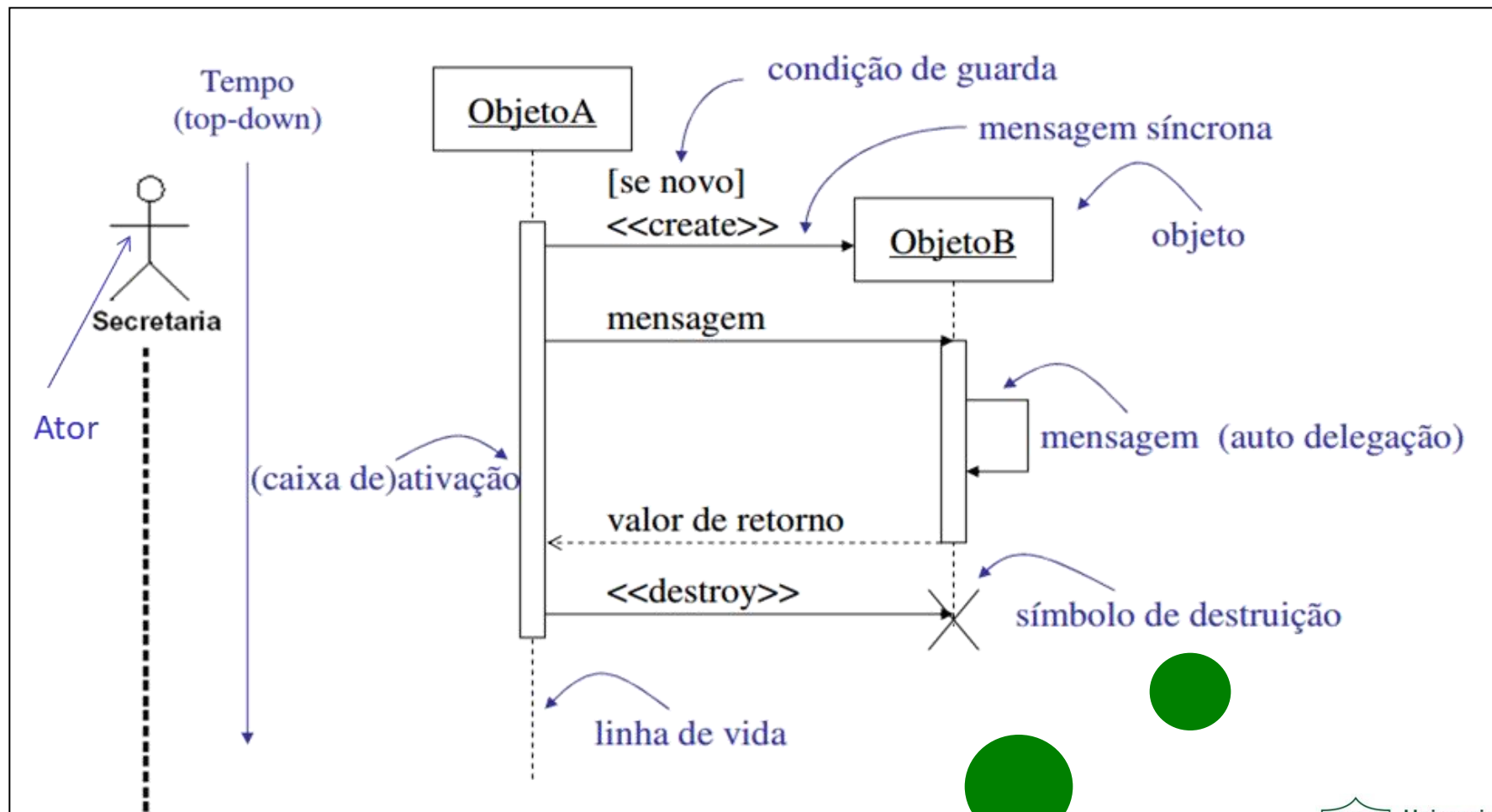
Sequenciamento

- Quando um objeto envia uma mensagem para outro objeto, o objeto que recebe a mensagem pode enviar outras mensagens e assim por diante, formando uma **sequência** de mensagens

O Sequenciamento pode ser:

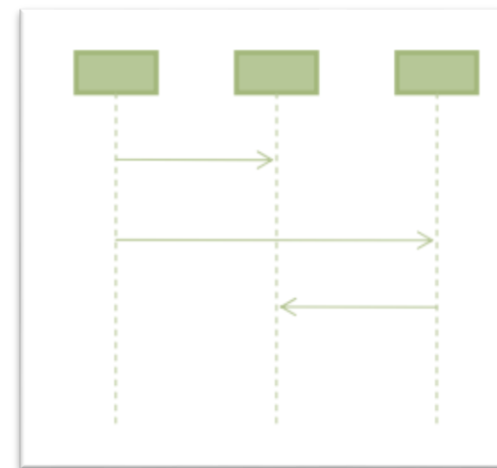
- *Procedural*:
mensagens síncronas (alinhamento)
- *Plano*:
mensagens assíncronas (sem alinhamento)

Notação



Elementos básicos:

- Atores
- Objetos e classes - organizados na horizontal e representados **por caixas**
- Linhas de vida - Abaixo de cada objeto existe uma linha vertical representando a **vida do objeto**
- Mensagens - representadas com **linhas horizontais** rotuladas, partindo da linha de vida do objeto remetente e chegando a linha de vida do objeto receptor



carla : Pessoa

● Objetos e classes

nomes definidos no diagrama de classes

- nome genérico (ex.: alguém : Pessoa)
- nome específico (ex.: carla : Pessoa)
- Nome omitido (ex.: : Pessoa)

● Linhas de vida

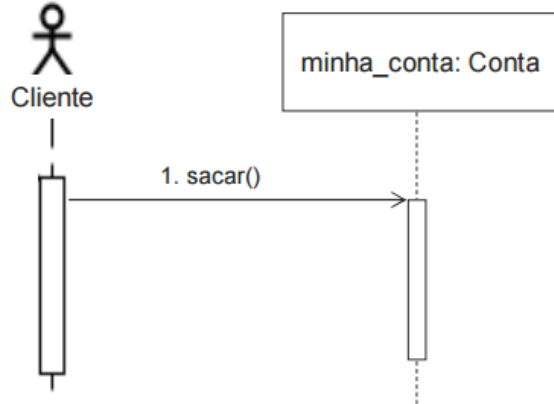
sai do objeto e representa o momento da sua criação. Quanto mais baixo, mais o tempo passou.

● Mensagens

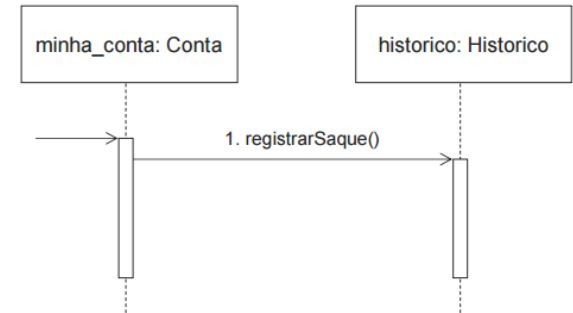
representa a interação entre objetos e contém a assinatura do método chamado. Pode ser:

- Entre objetos ou entre atores
- Para o mesmo objeto (reflexiva)
- De ator para objeto e vice-versa

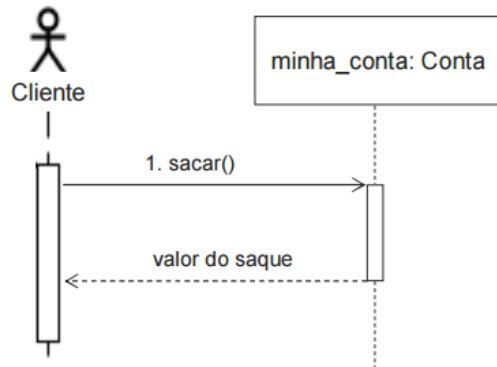
Atores para objeto



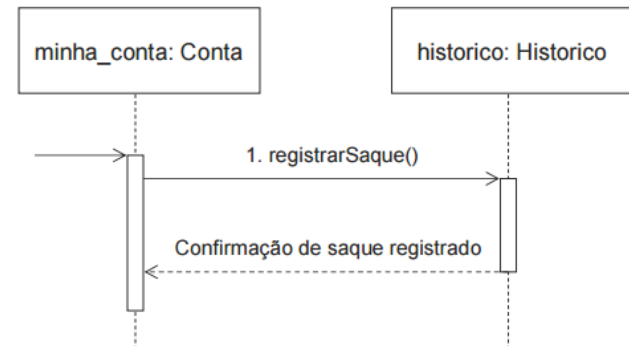
Objeto para objeto






Objeto para ator



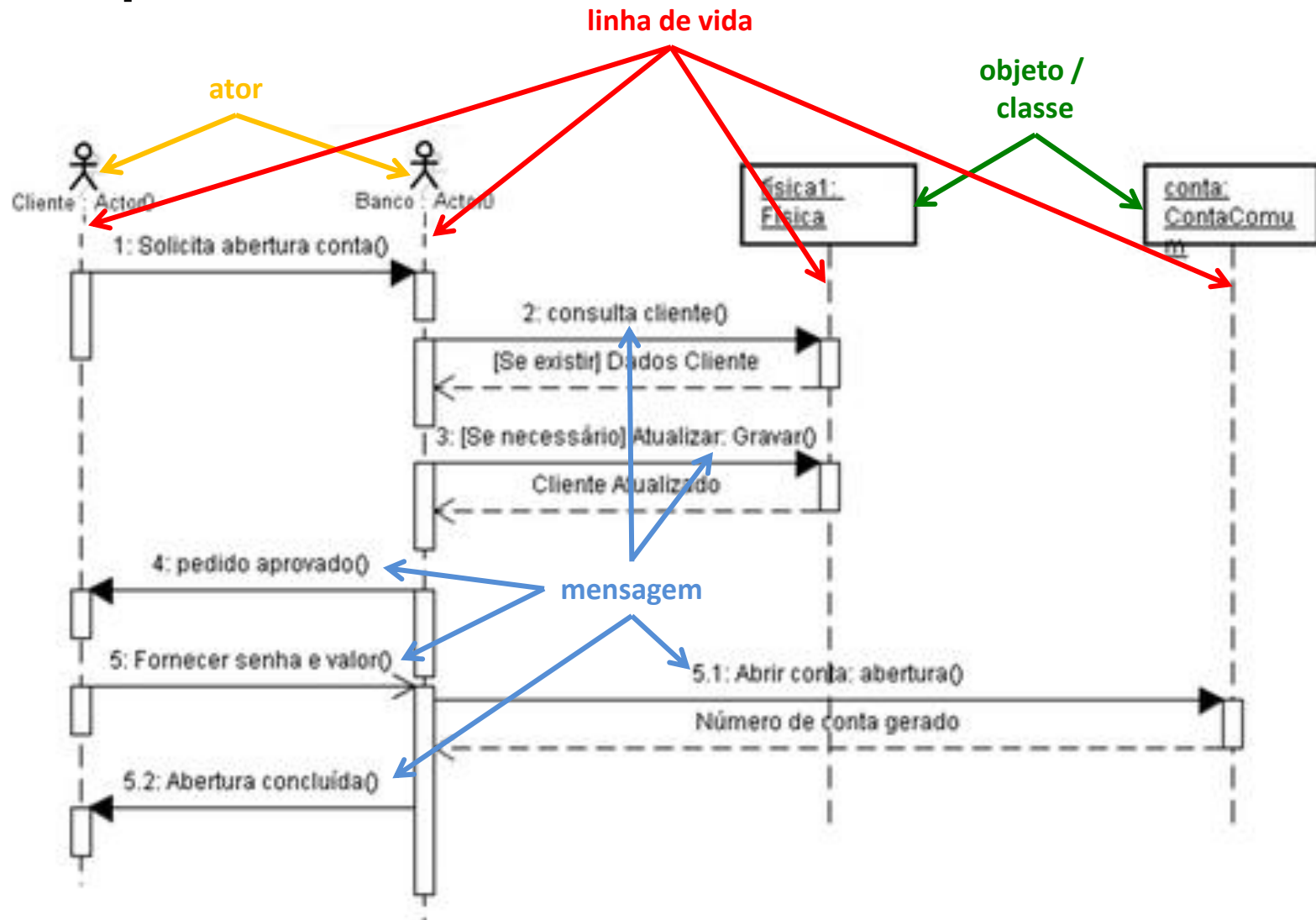
Mensagem de retorno



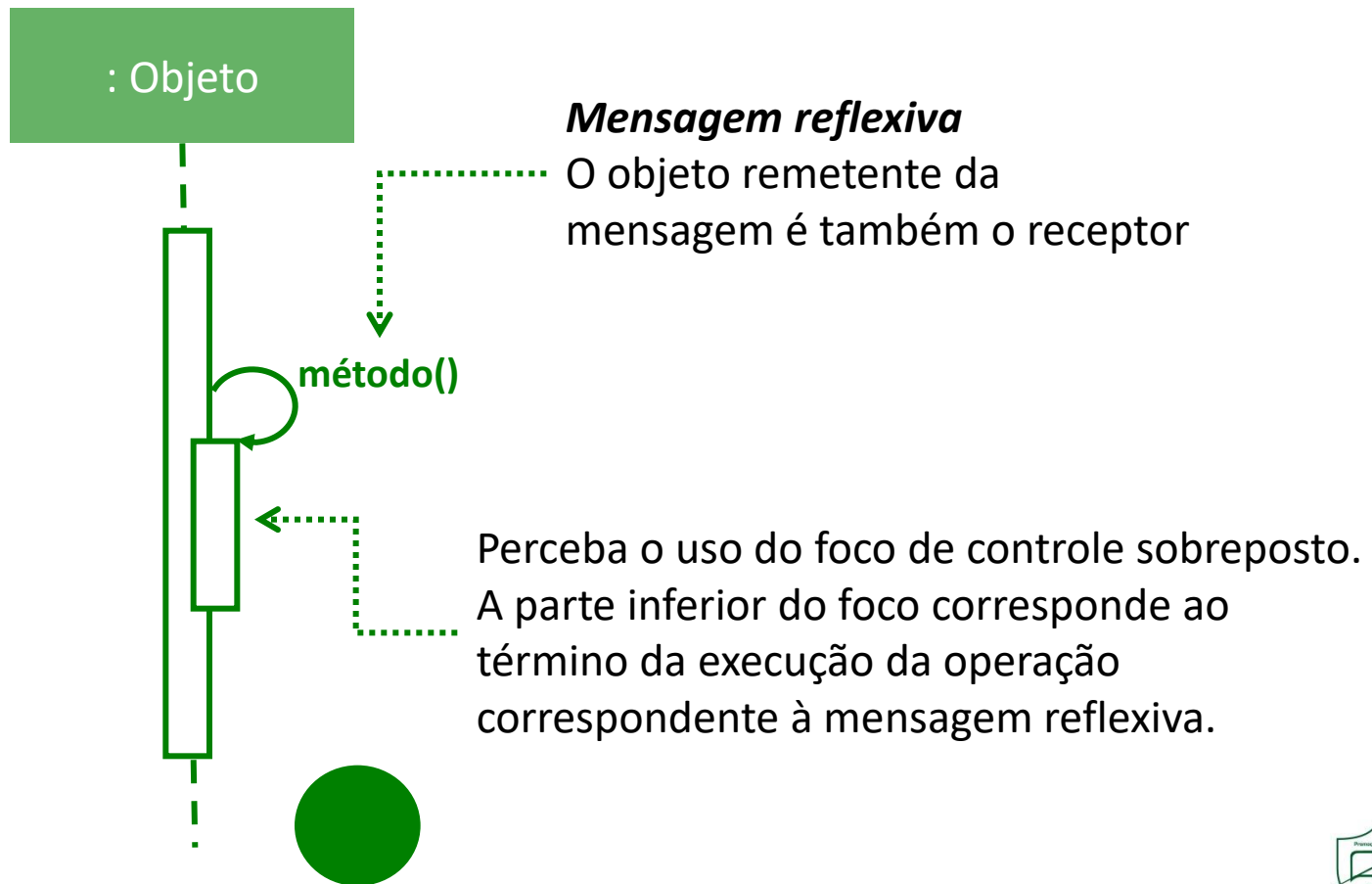
Representações

Símbolo	Significado
	Mensagem síncrona
	Mensagem assíncrona
	Mensagem de retorno (opcional)

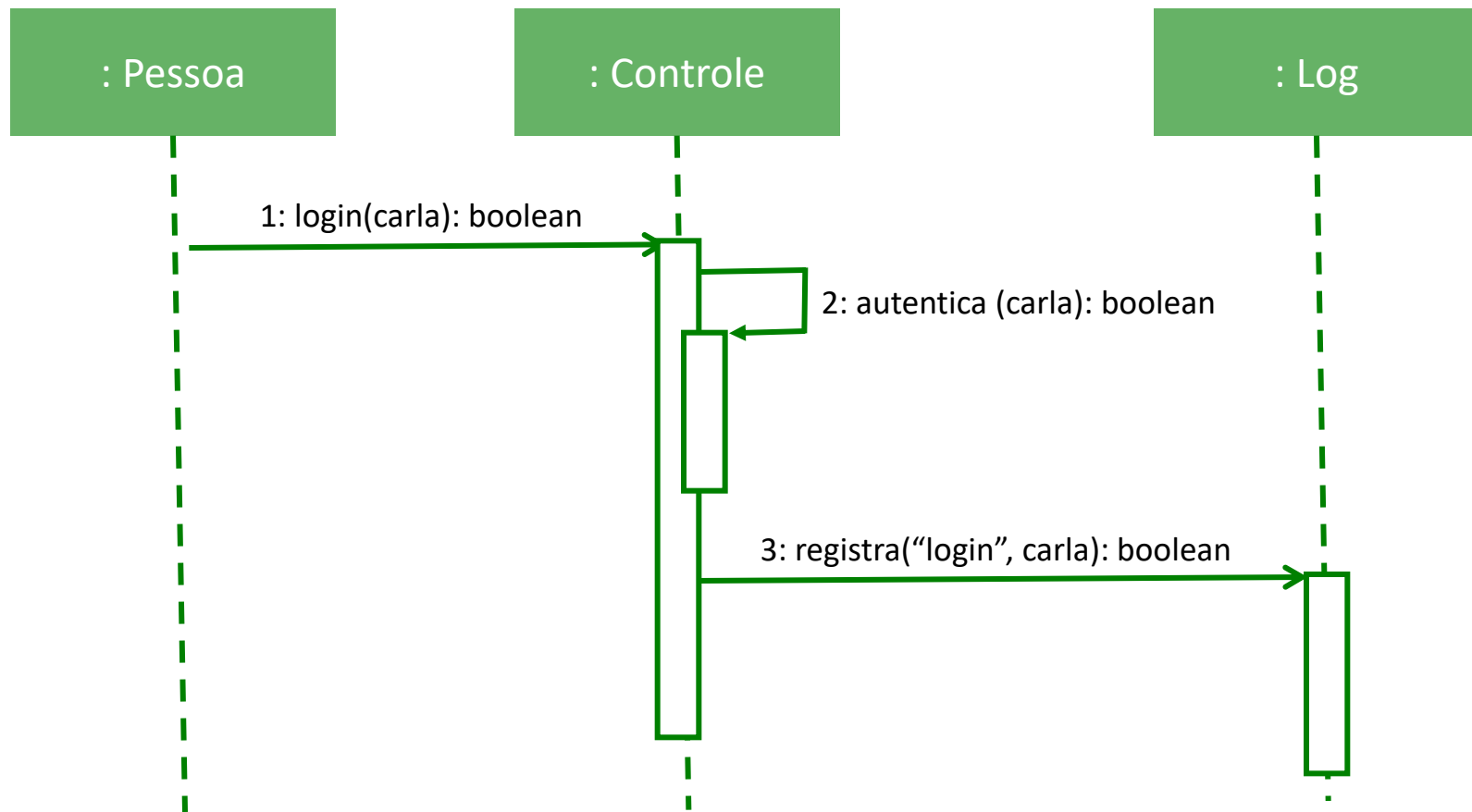
Exemplo:



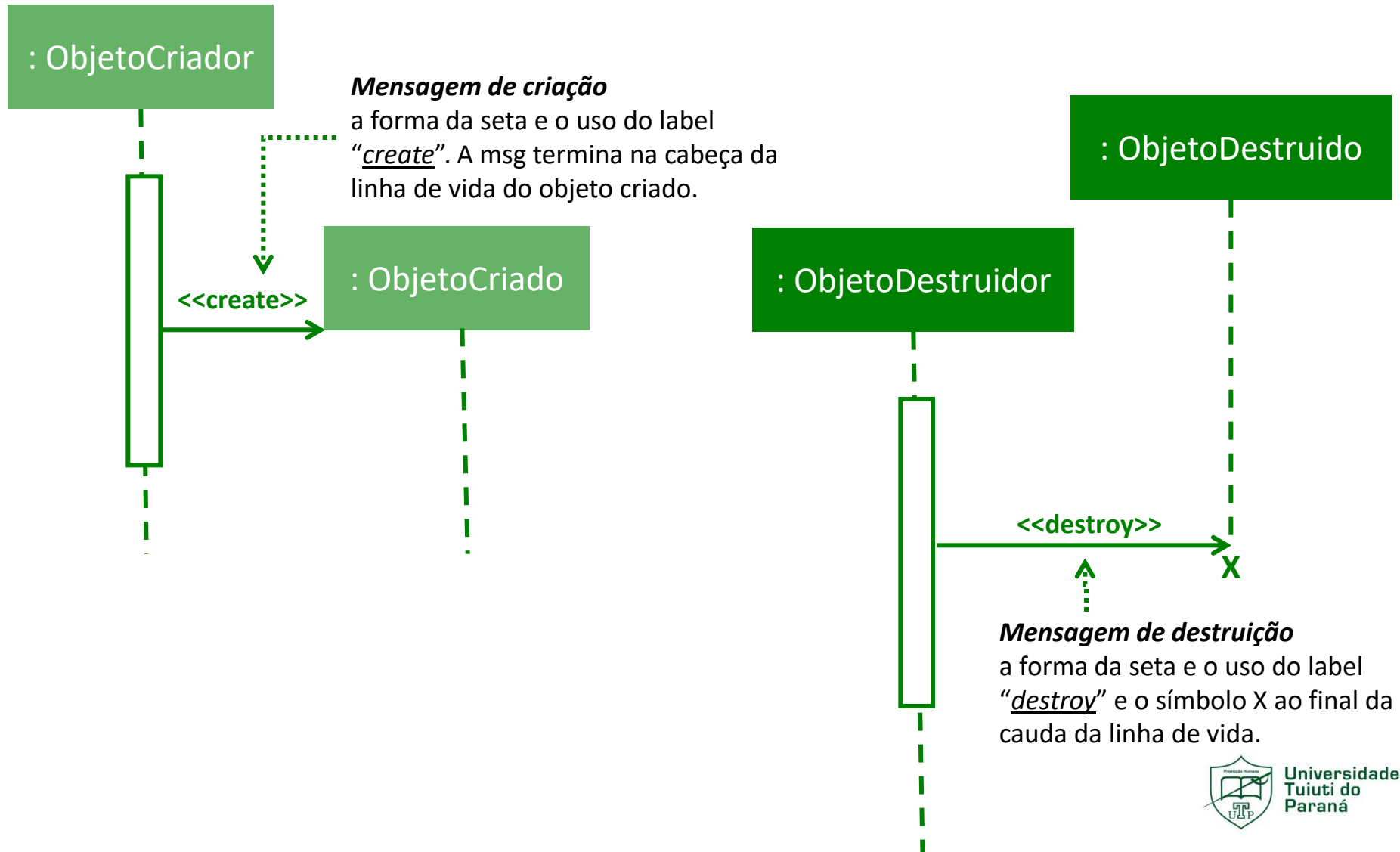
A mensagem por ser **reflexiva** (ou auto-mensagem), onde ela é o emissor e o receptor.



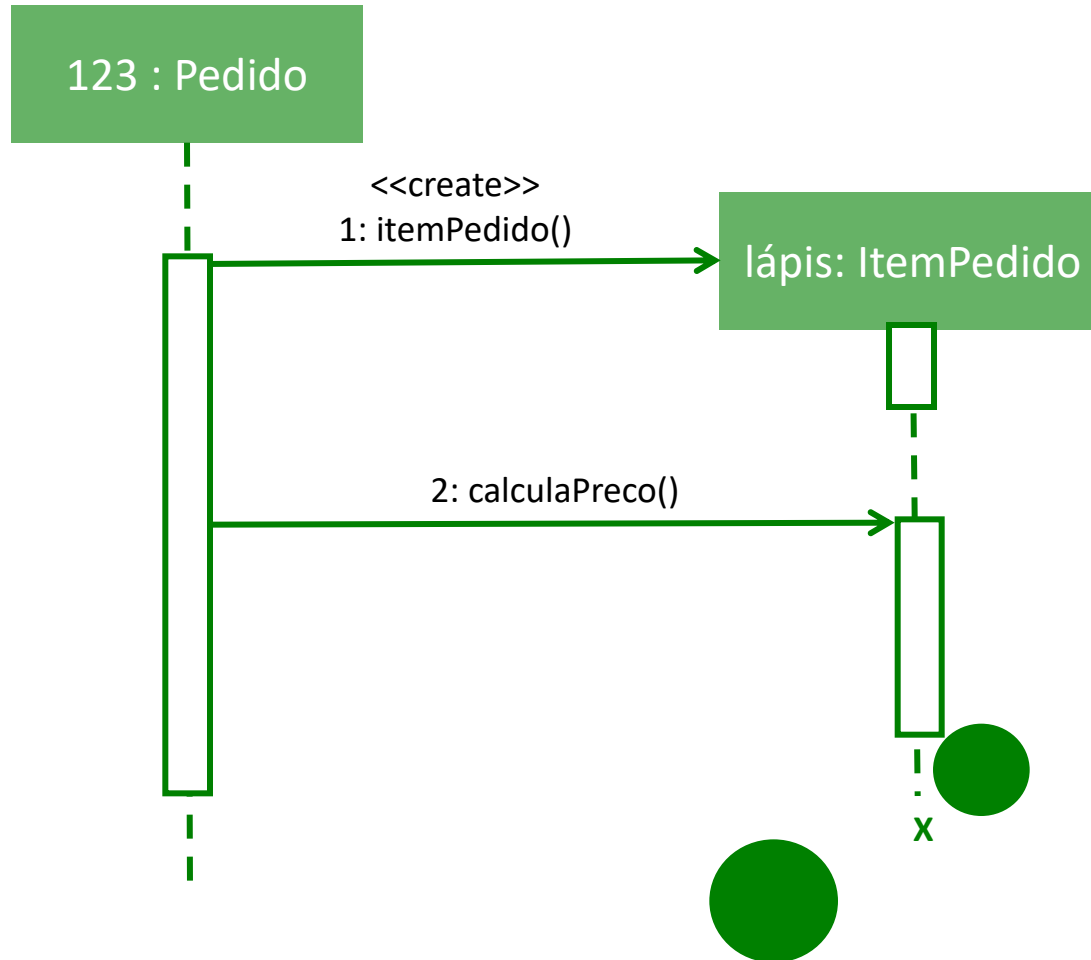
Exemplo:



Mensagem de **criação** (e **destruição**) de um objeto



Exemplo:

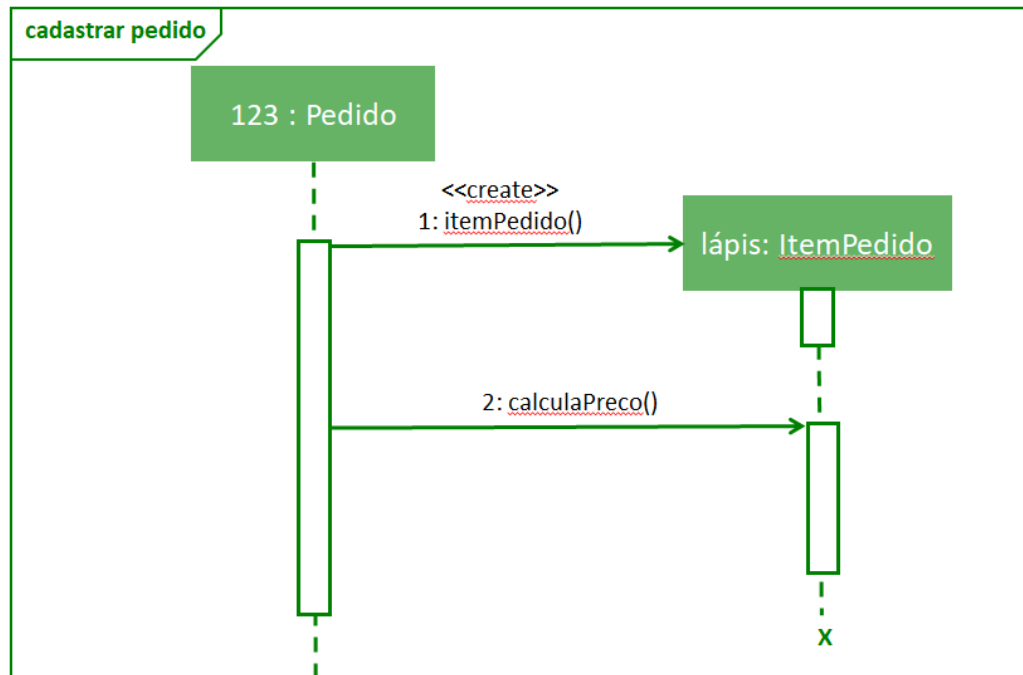


Quadro de interação

elemento gráfico, que serve para modularizar a construção de diagramas de sequência (ou de comunicação)

nome

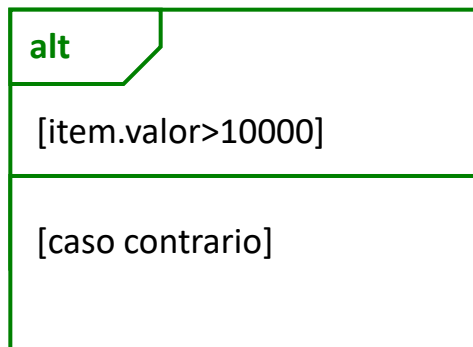
Exemplo:



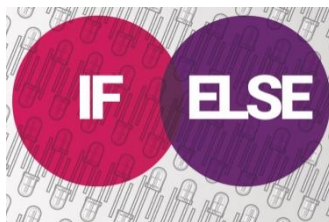
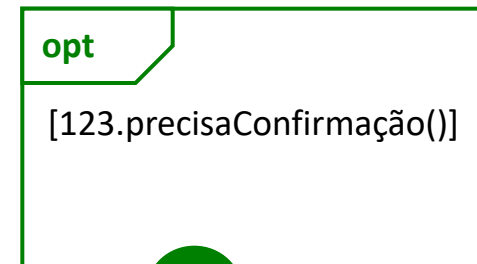
Quadros de interação servem
para registrar **decisões**.

Para isso são utilizados quadros
do tipo **alt** e **opt** com as condições.

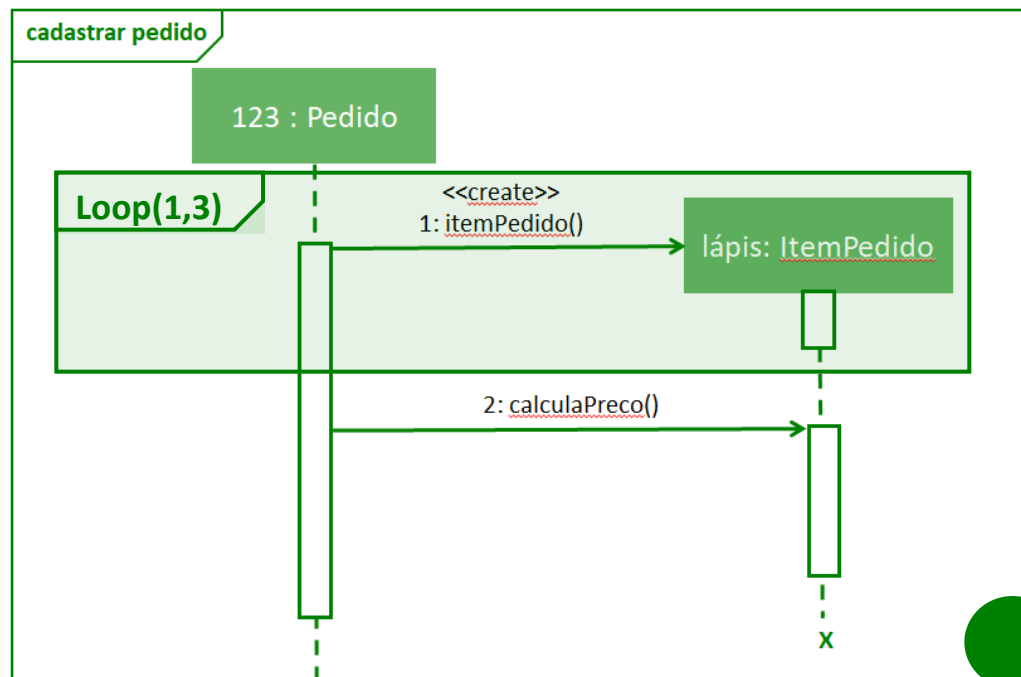
If + else



if

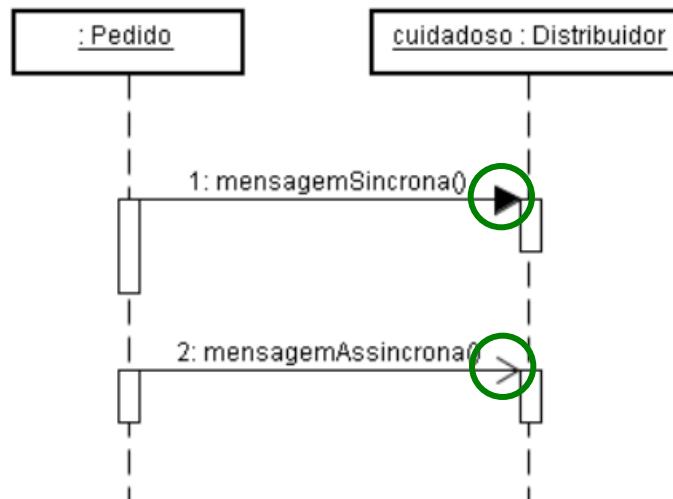


Quadros de interação também servem para registrar **repetições** durante o fluxo. Para isso os quadros são do tipo **loop**.



Sequenciamento procedural x plano

- Sequenciamento procedural ou síncrono (seta cheia): a execução fica bloqueada até o retorno do método
- Sequenciamento plano ou assíncrono (seta vazia): a execução continua em paralelo ao método que foi chamado

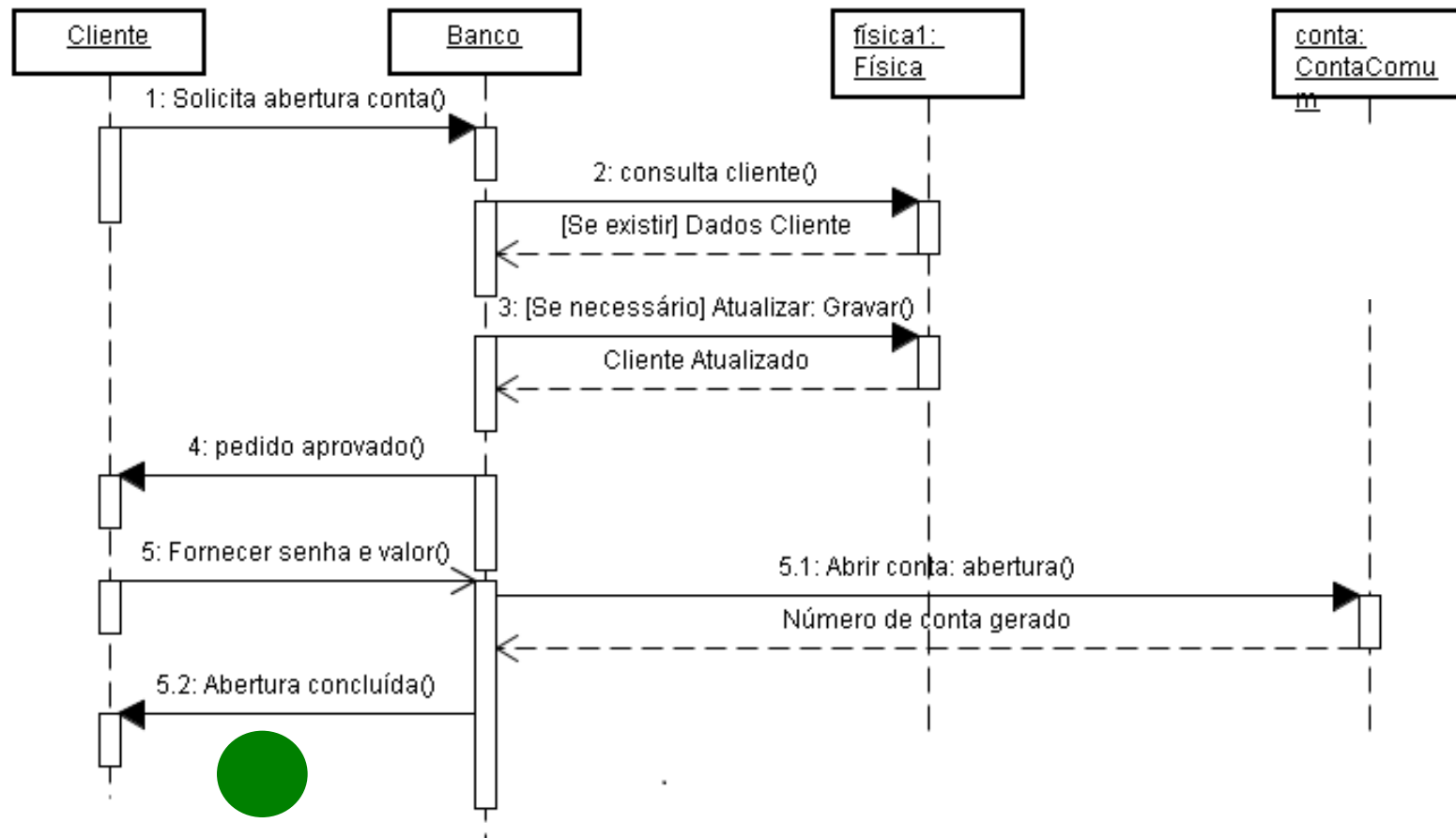


Melhores práticas da modelagem de diagramas de Sequência:

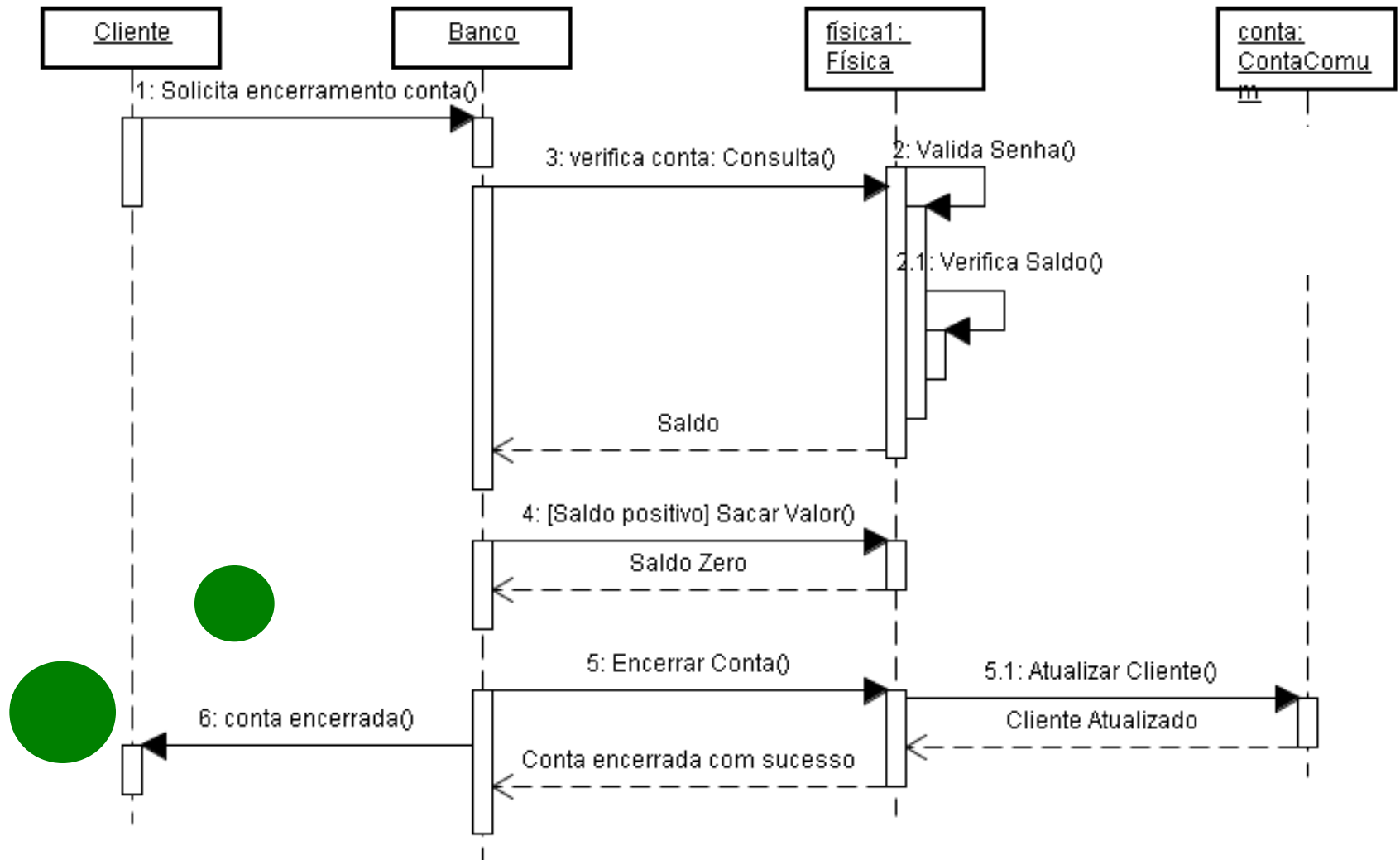
- Na primeira coluna deve ser colocado o ator que iniciou o caso de uso
- Na segunda coluna deve ser colocado um objeto de fronteira (usado pelo ator para iniciar o caso de uso)
- Na terceira coluna deve ser colocado o objeto de controle responsável pelo caso de uso
- Objetos de entidade normalmente são compartilhados por outros diagramas de sequência

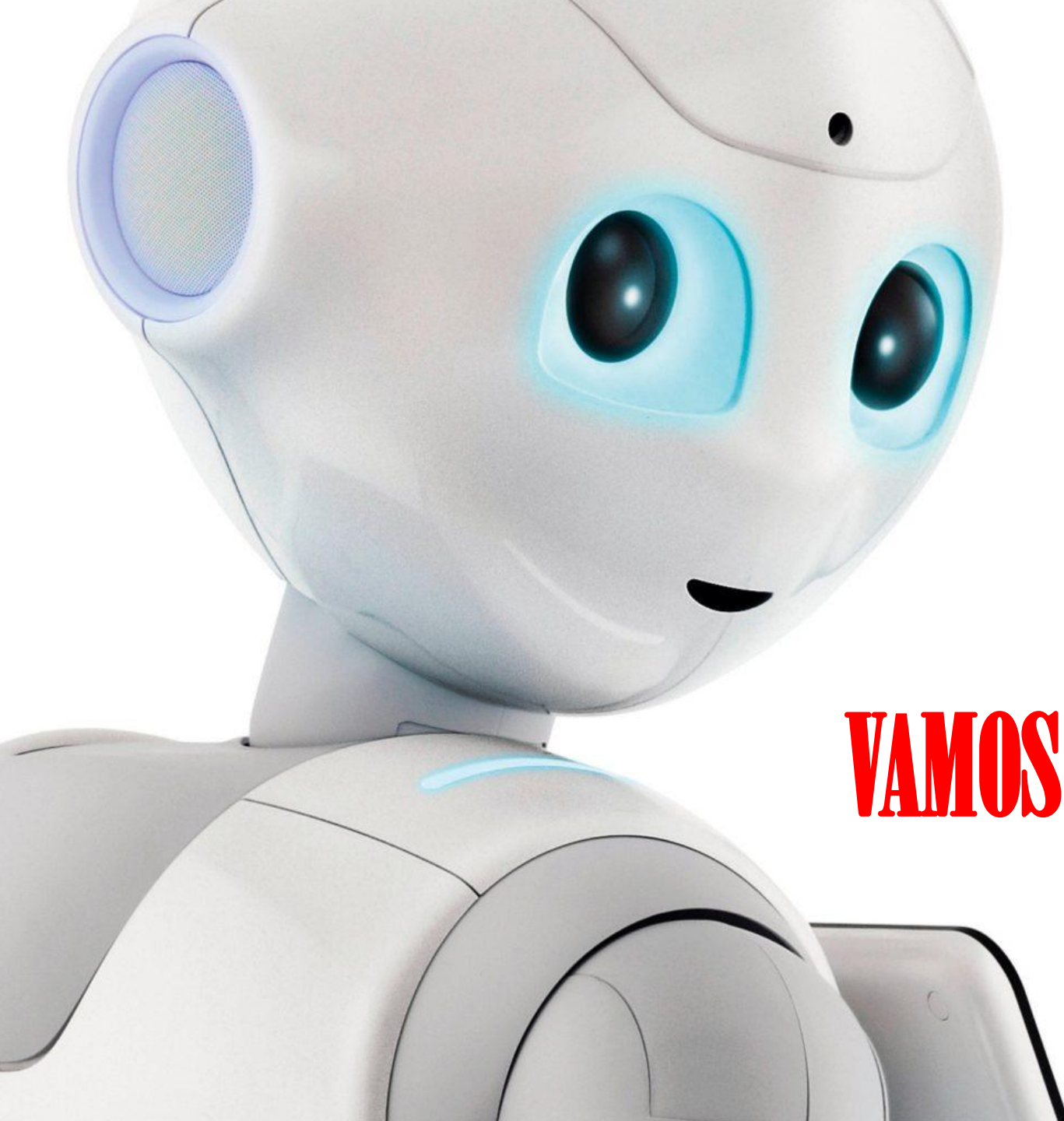
EXAMPLES

Exemplo: Abertura de Conta

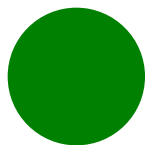


Exemplo: Encerramento da Conta

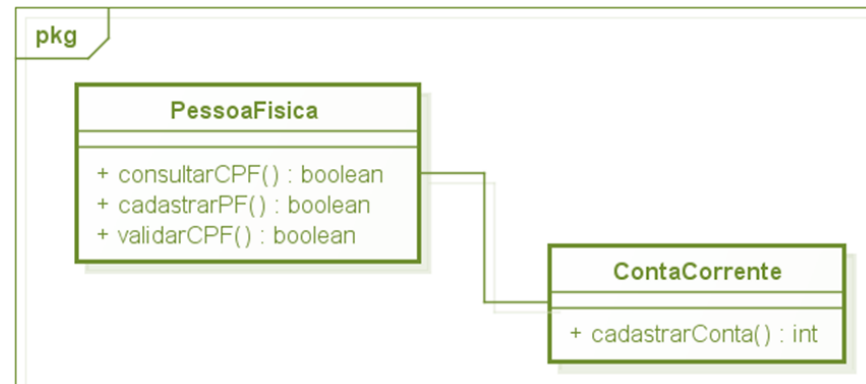




VAMOS PRATICAR?



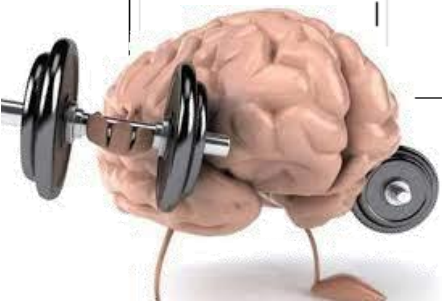
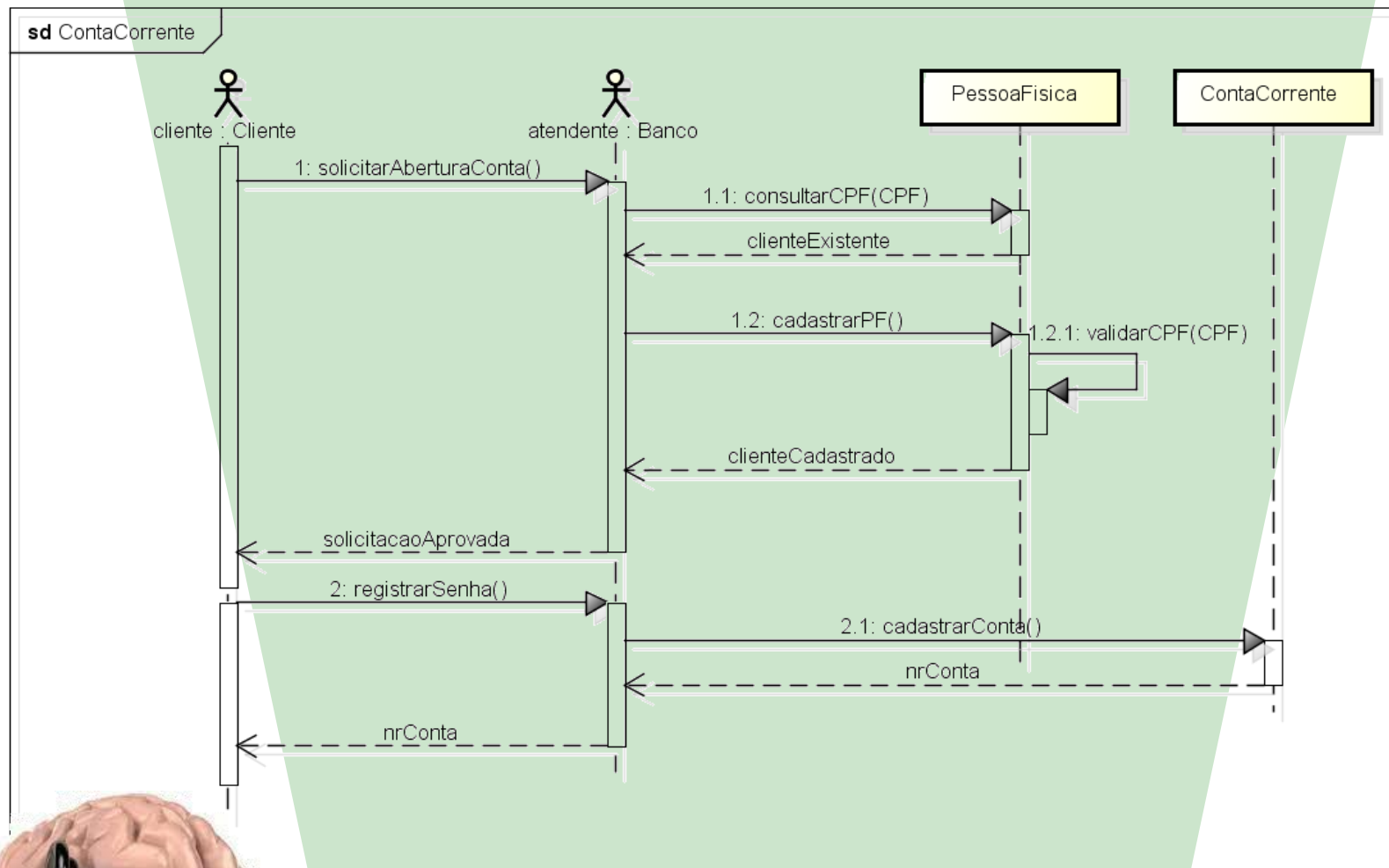
EXERCÍCIO 1: DIAGRAMA ABERTURA CONTA CORRENTE



Construa o diagrama de sequência conforme descrição abaixo:

- O cliente solicita a abertura de conta entregando toda a documentação exigida ao banco
- O atendente do banco verifica se já existe cadastro para o cliente, por intermédio da consulta de CPF, se não existir ele cria o cadastro
- Caso seja necessário cadastrar o cliente, o atendente deve validar se o CPF informado é um CPF válido
- Após a confirmação do cadastro do cliente, o cliente é informado que o cadastro ocorreu com sucesso e informa a senha para que assim o atendente realize abertura da conta

RESPOSTA EXERCÍCIO 1





Obrigada!

