

Você deverá construir uma aplicação em javascript que simula o funcionamento de um banco, com as funcionalidades de depósito, transferência e empréstimos em uma conta bancária. Para isso, crie as seguintes classes:

- **Deposit:**
 - Deverá possuir um atributo para o valor e um atributo para a data de criação.
- **Transfer:**
 - Deverá possuir atributos para: o usuário que enviou a transferência, o que recebeu a transferência, o valor e a data de criação.
- **Loan:**
 - Deverá possuir um atributo estático privado para a taxa de juros e que possuir um getter estático para ler o atributo privado e um setter estático para definir uma nova taxa de juros a partir de uma porcentagem.
 - Deverá possuir os atributos para o valor do empréstimo e para a data de criação.
 - Também deverá possuir um atributo para as parcelas do empréstimo, sendo que o construtor deve ter como parâmetro o número de parcelas e então deve calcular as parcelas (instâncias de Installments) e armazená-las nesse atributo.
- **Installment:**
 - Deve possuir atributos para: o valor da parcela, o número dela e sua situação (paga ou pendente)
- **Account:**
 - Deve possuir atributos para: o saldo, todos os depósitos realizados, todos os empréstimos, todas as transferências e para o dono da conta.
 - O atributo do saldo deve ser privado e somente de leitura, ou seja, seu valor não pode ser reatribuído, somente podendo ser modificado através dos depósitos, empréstimos e transferências.
 - Deve possuir um método para fazer um novo depósito, onde o valor do depósito será acrescentado ao saldo e a instância de Deposit ao atributo array de depósitos.
 - Deve possuir um método para fazer um novo empréstimo, que deverá acrescentar o valor do empréstimo ao saldo e salvar a instância de Loan no atributo array de empréstimos.
 - Deve possuir um método de fazer uma transferência, onde é verificado se a transferência foi feita para o dono da conta ou pelo dono da conta para outro usuário. Se feito

para o dono da conta, o valor dela deve ser acrescentado ao saldo. Se feito pelo dono da conta para outro usuário, o valor dela deve ser descontado do saldo. Além disso, a transferência deve ser salva no atributo array de transferências.

- User:

- Deve possuir atributos para nome completo, email e conta.

- App:

- Deve possuir um atributo estático privado para a lista dos usuários do app.
- Deve possuir um método estático para criar um novo usuário que só permite criar um usuário por email, ou seja, se o email já estiver sendo usado por outro usuário não deverá ser utilizado novamente.
- Deve possuir um método estático para encontrar um usuário a partir do seu email.
- Deve possuir métodos estáticos para realizar as operações de depósito, transferência e empréstimo. Elas devem ter como parâmetro as informações necessárias, como o email do usuário que está realizando a operação e o valor.
- Deve possuir um método para alterar a taxa dos empréstimos.