

Combinando ConnectionFactory e NegociacaoDao

Qual das opções abaixo combina corretamente as classes `ConnectionFactory` e `NegociacaoDao`. Lembre-se que `NegociacaoDao` depende de uma `connection` e `ConnectionFactory` é a classe que possui a responsabilidade de devolver conexões.

Selecione uma alternativa

A

```
ConnectionFactory()
  .getConnection()
  .then(conexao => new NegociacaoDao(conexao))
  .then(dao => dao.adiciona(new Negociacao(new Date(), 1, 200.13)))
  .then(() => console.log('adicionado com sucesso'))
  .catch(() => console.log('não foi possível adicionar'));
```

B

```
ConnectionFactory()
  .then(conexao => new NegociacaoDao(conexao))
  .then(dao.adiciona(new Negociacao(new Date(), 1, 200.13)))
  .then(() => console.log('adicionado com sucesso'))
  .catch(() => console.log('não foi possível adicionar'));
```

C

```
ConnectionFactory
  .getConnection()
  .then(conexao => new NegociacaoDao(conexao))
  .then(dao => dao.adiciona(new Negociacao(new Date(), 1, 200.13)))
  .then(() => console.log('adicionado com sucesso'))
  .catch(() => console.log('não foi possível adicionar'));
```