

Desenvolva uma classe **Carro**, com os seguintes atributos:

**Marca** (não pode ficar vazio);  
**Modelo** (não pode ficar vazio);  
**Consumo médio**, em quilômetros por litro (não pode ser negativo);  
**Quantidade de combustível**, em litros (não pode ser negativo);  
**Quilometragem** (inicia com zero e só aumenta);

Encapsule todos os dados (`private`), e não crie métodos `set`.

Crie os seguintes métodos:

- **dois construtores**: um que recebe todos os dados, e outro que recebe todos exceto a quilometragem e o nível de combustível (indicando um carro zero km e sem combustível).
- **andar**: Recebe a quilometragem a ser percorrida, que deve ser maior que zero. Caso haja combustível suficiente para rodar a quilometragem pedida (de acordo com o consumo do carro), faz as seguintes ações:
  - incremente a quilometragem do carro;
  - diminua a quantidade de combustível (de acordo com o consumo do seu carro);
- **abastecer**: Recebe uma quantidade de combustível (valor positivo) e adiciona ao tanque do carro. Não pode receber um valor negativo;
- **exibir (toString)**: retorna uma String com todos os dados do carro;



Crie uma classe **CelularPrePago**, que tenha os atributos:

**Nome do cliente** (pode ser vazio);  
**Número do telefone** (não pode ser vazio);  
**Saldo** (não pode ser negativo);

Todas as variáveis devem ser encapsuladas, e o saldo não deve possuir um método de `set`.

O saldo será alterado pelos seguintes métodos:

- **fazerLigacao**: caso o usuário tenha saldo suficiente, desconta o valor da ligação do saldo; caso contrário, não faz a ligação. Considere que cada ligação custa R\$ 0,75.
- **recarregar**: recebe um valor a ser adicionado no saldo. Não aceita valores negativos.

Escreva também os seguintes métodos:

- **construtor** que receba apenas o nome e número, e outro que receba todos os tributos.
- **exibir**, que retorna uma String com todos os dados.

