

TestePrático1

- Dado que um cliente deseja realizar uma recarga de celular pela a sua instituição financeira. Implemente uma função que permita que o cliente realize uma recarga em sua linha onde o débito ocorrerá de sua conta corrente. Considerando testes negativos e positivos, implemente um teste unitário para as funções de recarga de saldo e para o saldo atual da linha. Utilize as classes abaixo para o desenvolvimento do programa.

Observação: Utilize a linguagem que preferir para o desenvolvimento do teste prático

Classes:

```
-----  
  
public class Cliente {  
  
    private String nome_cliente;  
    private Telefone telefone;  
    private Conta conta;  
    public Cliente(String nome_cliente, Telefone telefone, Conta conta){  
        this.nome_cliente=nome_cliente;  
        this.telefone=telefone;  
        this.conta=conta;  
    }  
    public void setNome(String nome_cliente) {  
        this.nome_cliente = nome_cliente;  
    }  
    public String getNome() {  
        return this.nome_cliente;  
    }  
    public void setLinha(Telefone telefone) {  
        this.telefone = telefone;  
    }  
    public Telefone getTelefone() {  
        return this.telefone;  
    }  
    public void setConta(Conta conta) {  
        this.conta = conta;  
    }  
    public Conta getConta() {  
        return this.conta;  
    }  
}
```

```
public class Conta {

    private int saldo;
    private int vl_recarga;
    public Conta(int saldo){
        this.saldo = saldo;
    }
    public void setSaldo(int saldo) {
        this.saldo = saldo;
    }
    public int getSaldo() {
        return this.saldo;
    }
    public void setRecarga(int vl_recarga) {
        this.vl_recarga = vl_recarga;
    }
    public int getRecarga() {
        return this.vl_recarga;
    }
}
```

```
public class Telefone {

    private String numero_linha;
    private int saldo;
    public Telefone(String numero_linha, int saldo){
        this.numero_linha = numero_linha;
        this.saldo=saldo;
    }
    public void setNumeroLinha(String numero_linha) {
        this.numero_linha = numero_linha;
    }
    public String getNumeroLinha() {
        return this.numero_linha;
    }
    public void setSaldo(int saldo) {
        this.saldo = saldo;
    }
    public int getSaldo() {
        return this.saldo;
    }
}
```

Teste Prático 2

- Dado que um cliente deseja realizar o cadastro do endereço no Meu Vivo. Para isso será necessário que a aplicação do Meu Vivo acesse a API dos correios informando o CEP e retorne os campos de identificação do endereço (Rua, Bairro, etc.). Realize a validação dos campos de retorno da api dos correios de acordo com o cep informado, onde esses campos não podem ser vazios. Considere receber o cep de uma variável de ambiente e utilize a aba "Pre-request script" para construir o request da api.
- Observação: Para a realização desse teste assuma utilizar Postman.
- Download Postman: <https://www.postman.com/downloads/>
- Endpoint :
<https://apps.correios.com.br/SigepMasterJPA/AtendeClienteService/AtendeCliente>
- WSDL (Capturar o body do serviço):
<https://apps.correios.com.br/SigepMasterJPA/AtendeClienteService/AtendeCliente?wsdl> - (Utilize a operação de "consultaCEP")

Teste Prático 3

Observação: para a realização desse teste considere criar o banco de dados MySQL abaixo em sua máquina.

```
CREATE DATABASE `banco_teste_automacao`;  
CREATE TABLE `massas` (  
  `IDMASSAS` int(11) NOT NULL AUTO_INCREMENT,  
  `NAME_PRODUCT` varchar(45) DEFAULT NULL,  
  `CUSTOMIZATION` varchar(45) DEFAULT NULL,  
  `DISPLAY` varchar(600) DEFAULT NULL,  
  `DISPLAY_RESOLUTION` varchar(45) CHARACTER SET utf8 COLLATE utf8_general_ci  
  DEFAULT NULL,  
  `DISPLAY_SIZE` varchar(45) DEFAULT NULL,  
  `MEMORY` varchar(45) DEFAULT NULL,  
  `OPERATING_SYSTEM` varchar(45) DEFAULT NULL,  
  `PROCESSOR` varchar(255) DEFAULT NULL,  
  `TOUCHSCREEN` varchar(45) DEFAULT NULL,  
  `WEIGHT` varchar(45) DEFAULT NULL,  
  `COLOR` varchar(45) DEFAULT NULL,  
  PRIMARY KEY (`IDMASSAS`)  
) ENGINE=InnoDB AUTO_INCREMENT=2 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

```
insert into  
massas(NAME_PRODUCT,CUSTOMIZATION,DISPLAY,DISPLAY_RESOLUTION,DISPLAY_S  
IZE,MEMORY,OPERATING_SYSTEM,PROCESSOR,TOUCHSCREEN,WEIGHT,COLOR)  
values("HP PAVILION 15Z TOUCH LAPTOP","Simplicity","15.6-inch diagonal Full HD  
WLED-backlit Display (1920x1080) Touchscreen","1920x1080","15.6","16GB DDR3 - 2  
DIMM","Windows 10","AMD Quad-Core A10-8700P Processor + AMD Radeon(TM) R6  
Graphics","Yes","5.51 lb","GRAY");
```

- **Transcreva os cenários abaixo em gherkin e após isso realize a automação destes com Selenium:**
- **Cenário 1 – Validar especificações do produto**
 - Acessar o site <https://advantageonlineshopping.com>
 - No menu, clicar na opção "Special Offer"
 - Clicar no botão "See offer"

- Validar que as especificações do produto de acordo com as informações retornadas do banco de automação

- **Cenário 2 – Validar alteração de cor do produto no carrinho**
 - Acessar o site <https://advantageonlineshopping.com>
 - No menu, clicar na opção “Special Offer”
 - Clicar no botão “See offer”
 - Alterar a cor do produto de acordo com a cor informada no banco de automação
 - Clicar no botão “Add to cart”
 - Validar que produto foi adicionado ao carrinho com a cor selecionada

- **Cenário 3 – Validar página de checkout**
 - Acessar o site <https://advantageonlineshopping.com>
 - Pesquisar o produto clicando no ícone de lupa (Seguir o nome do produto do banco de automação)
 - Selecionar produto pesquisado
 - Alterar a cor do produto para uma diferente da existente no banco de automação
 - Alterar a quantidade de produtos que deseja comprar
 - Clicar no botão “Add to cart”
 - Acessar a página de checkout
 - Validar que a soma dos preços corresponde ao total apresentado na página de checkout
 - Realizar um update no banco de automação alterar a cor existente no banco para a cor selecionada no teste.

- **Cenário 4 – Remover produto do carrinho de compras**
 - Acessar o site <https://advantageonlineshopping.com>
 - No menu, clicar na opção “Special Offer”
 - Clicar no botão “See offer”
 - Clicar no botão “Add to cart”
 - Clicar no carrinho de compras
 - Remover produto do carrinho de compras
 - Validar que o carrinho de compras está vazio