

Erros e Soluções



FOURSHOPP

Easy Way

Nome: Luiz Afonso Silva Costa

Turma: Fourcamp 2022

Sumário

1 - Erro: Sistema aceita CPFs inválidos e no formato incorreto.	2
1.1 - Descrição do erro	2
1.2 - Solução do erro	3
2 - Erro: Sistema aceita senhas menores que 8 caracteres	5
2.1 - Descrição do erro	5
2.2 - Solução do erro:	6
3 - Erro: O sistema se encontra sem tratamentos de exceções, ou seja, em campo que aceita apenas números , caso o usuário digite uma letra, o sistema encerra com erro.	7
3.1 - Descrição do erro	7
3.2 - Solução do erro:	10
4 - Erro: O sistema está aceitando que o usuário adquira uma quantidade de produtos maior em relação ao estoque.	12
4.1 - Descrição do erro	12
4.2 - Solução do erro	13
5 - Erro: O cliente não está recebendo um desconto de 10% a cada 500 reais em produtos do setor de mercearia.	14
5.1 - Descrição do erro	14
5.2 - Solução do erro	15
6 - Erro: ao cadastrar o Cliente (opção 3 do menu principal):	16
6.1 - Descrição do erro	16
6.2 - Solução do erro	23
7 - Erro: ao acessar a opção de cliente com cpf e senha não cadastrados (opção 1 do menu principal):	26
7.1 - Descrição do erro	26
7.2 - Solução do erro	30
8 - Erro: é possível entrar como um funcionário, chefe (opção 4 do menu principal e 1 do submenu), com um cpf e senha não cadastrados	31
8.1 - Descrição do erro	31
8.2 - Solução do erro	32
9 - Erro: Após realizar algumas funções do sistema, o usuário não é redirecionado ao menu	33
9.1 - Descrição do erro	33
9.2 - Solução do erro	34
10 - Erro: O sistema não finaliza ao digitar a opção 5	35
10.1 - Descrição do erro	35
10.2 - Solução do erro	36

Fourshop



1 - **Erro:** Sistema aceita CPFs inválidos e no formato incorreto.

1.1 - Descrição do erro

Em todos os lugares do sistema onde é requisitado o CPF ao usuário não existe um tratamento para verificar se o mesmo é válido e se está no formato correto. Exemplo de uma entrada de CPF pedida ao cliente e informada incorretamente, porém salva no banco de dados:

Insira seu cpf:

123

Insira sua rua:

Rua Foursys

	cd_pessoa	nr_celular	nr_cpf	ds_email	nm_nome	ds_senha	cd_endereco
▶	1	15 91234-1234	123 ←	email@gmail.com	Luiz	1122	2

Fourshop



1.2 - Solução do erro

Foi criado um método na classe UtilMenu (dentro do package util) para realizar o tratamento das entradas que exigiam o CPF ao usuário, este método verifica se a entrada veio no formato correto (xxx.xxx.xxx-xx) e se o CPF é válido. Método:

```
public static boolean cpfValidate(String cpf) {  
  
    String pattern = "([0-9]{3}[\\.]?[0-9]{3}[\\.]?[0-9]{3}[\\-]?[0-9]{2})";  
    Pattern regex = Pattern.compile(pattern);  
  
    Matcher matcher = regex.matcher(cpf);  
  
    if(!matcher.matches()) {  
        return false;  
    }  
  
    String S1, S2, S3, S4, S5, S6, S7, S8, S9, check = "";  
    int N1, N2, N3, N4, N5, N6, N7, N8, N9, verify1, verify2;  
  
    S1 = cpf.substring(0, 1);  
    N1 = Integer.valueOf(S1);  
    S2 = cpf.substring(1, 2);  
    N2 = Integer.parseInt(S2);  
    S3 = cpf.substring(2, 3);  
    N3 = Integer.valueOf(S3);  
    S4 = cpf.substring(4, 5);  
    N4 = Integer.parseInt(S4);  
    S5 = cpf.substring(5, 6);  
    N5 = Integer.valueOf(S5);  
    S6 = cpf.substring(6, 7);  
    N6 = Integer.parseInt(S6);  
    S7 = cpf.substring(8, 9);  
    N7 = Integer.parseInt(S7);  
    S8 = cpf.substring(9, 10);  
    N8 = Integer.valueOf(S8);  
    S9 = cpf.substring(10, 11);  
    N9 = Integer.parseInt(S9);  
  
    verify1 = (N1 * 10 + N2 * 9 + N3 * 8 + N4 * 7 + N5 * 6 + N6 * 5 + N7 * 4 + N8 * 3 + N9 * 2);  
  
    if ((verify1 % 11) < 2)  
        verify1 = 0;  
    else  
        verify1 = 11 - (verify1 % 11);  
  
    verify2 = (N1 * 11 + N2 * 10 + N3 * 9 + N4 * 8 + N5 * 7 + N6 * 6 + N7 * 5 + N8 * 4 + N9 * 3 + verify1 * 2);  
    if ((verify2 % 11) < 2)  
        verify2 = 0;  
    else  
        verify2 = 11 - (verify2 % 11);  
    check = (S1 + S2 + S3 + "." + S4 + S5 + S6 + "." + S7 + S8 + S9 + "-" + verify1 + "" + verify2);  
  
    if (check.equals(cpf))  
        return true;  
    else  
        return false;  
}
```

Após a criação do método, o mesmo foi implementado em todas as partes do sistema que pediram o CPF para o usuário, da seguinte forma:

```
System.out.println("Insira seu cpf (xxx.xxx.xxx-xx): ");
String cpf = scanner.next();

while(true) {
    if(!cpfValidate(cpf)) {
        System.out.println("\n===== \n" +
            "Insira um cpf válido e no formato correto (xxx.xxx.xxx-xx)! \n" +
            "===== \n");
        System.out.println("Insira seu cpf: ");
        cpf = scanner.next();
        continue;
    }
    break;
}
```

Fourshop



2 - **Erro:** Sistema aceita senhas menores que 8 caracteres

2.1 - Descrição do erro

Em todos os lugares do sistema onde é requisitado o cadastro de uma senha ao usuário, não existe um tratamento para verificar se a mesma é maior ou igual a 8 caracteres. Exemplo de uma entrada de senha cadastrada pelo usuário e salva no banco de dados:

Insira sua password:

1122

	cd_pessoa	nr_celular	nr_cpf	ds_email	nm_nome	ds_senha	cd_endereco
▶	1	15 91234-1234	123	email@gmail.com	Luiz	1122 ←	2

Fourshop



2.2 - Solução do erro:

Foi criado um método na classe UtilMenu (dentro do package util) para realizar o tratamento das entradas que exigiam o cadastro de uma senha ao usuário, este método verifica se a senha é maior ou igual a 8 caracteres e se a mesma estiver incorreta ele pede para o usuário digitar a senha novamente seguindo esta regra. Método:

```
public static String passwordValidate(String password, Scanner scanner) {  
    while (true) {  
        if(password.length() < 8) {  
            System.out.println("\n===== \n" +  
                "Digite uma senha maior ou igual a 8 caracteres! \n" +  
                "===== \n");  
            System.out.println("Insira sua senha: ");  
            password = scanner.next();  
            continue;  
        }  
        return password;  
    }  
}
```

Após a criação do método, o mesmo foi implementado em todas as partes do sistema que pediam uma senha para o usuário, da seguinte forma:

```
System.out.println("Insira sua password: ");  
String password = scanner.nextLine();  
  
password = passwordValidate(password, scanner);
```

Fourshop



3 - **Erro:** O sistema se encontra sem tratamentos de exceções, ou seja, em campo que aceita apenas números , caso o usuário digite uma letra, o sistema encerra com erro.

3.1 - Descrição do erro

Em todos os lugares do sistema onde é requisitado um inteiro ou um double ao usuário, não existe o respectivo tratamento dos erros que os mesmos geram quando o usuário insere um ou vários caracteres inválidos.

```
***** BEM-VINDO AO FOURSHOPP *****
1- Sou cliente
2- Área do ADM
3- Seja um Cliente
4- Login funcionário
5- Encerrar
***
2022-06-28 10:43:35.252 INFO 6568 --- [ restartedMain] ConditionEvaluationReportLoggingListener :

Error starting ApplicationContext. To display the conditions report re-run your application with 'debug' enabled.
2022-06-28 10:43:35.277 ERROR 6568 --- [ restartedMain] o.s.boot.SpringApplication : Application run failed

java.lang.IllegalStateException: Create breakpoint : Failed to execute CommandLineRunner
    at org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:770) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.callRunners(SpringApplication.java:751) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:309) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:1301) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:1290) ~[spring-boot-2.6.1.jar:2.6.1]
    at br.com.fourshopp.FourShoppApplication.main(FourShoppApplication.java:51) ~[classes/:na] <4 internal lines>
    at org.springframework.boot.devtools.restart.RestartLauncher.run(RestartLauncher.java:49) ~[spring-boot-devtools-2.6.1.jar:2.6.1]
Caused by: java.util.InputMismatchException: Create breakpoint : null
    at java.base/java.util.Scanner.throwFor(Scanner.java:939) ~[na:na]
    at java.base/java.util.Scanner.next(Scanner.java:1594) ~[na:na]
    at java.base/java.util.Scanner.nextInt(Scanner.java:2258) ~[na:na]
    at java.base/java.util.Scanner.nextInt(Scanner.java:2212) ~[na:na]
    at br.com.fourshopp.FourShoppApplication.run(FourShoppApplication.java:63) ~[classes/:na]
    at org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:767) ~[spring-boot-2.6.1.jar:2.6.1]
    ... 10 common frames omitted

2022-06-28 10:43:35.309 INFO 6568 --- [ restartedMain] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2022-06-28 10:43:35.312 INFO 6568 --- [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2022-06-28 10:43:35.323 INFO 6568 --- [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.

Process finished with exit code 0
```

Erro:

```
2022-06-28 10:43:35.252 INFO 6568 --- [ restartedMain]
ConditionEvaluationReportLoggingListener :
```



```

Error starting ApplicationContext. To display the conditions report re-run
your application with 'debug' enabled.
2022-06-28 10:43:35.277 ERROR 6568 --- [ restartedMain]
o.s.boot.SpringApplication                : Application run failed

java.lang.IllegalStateException: Failed to execute CommandLineRunner
    at
org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:770) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.callRunners(SpringApplication.java:751) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.java:309)
~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.java:1301)
~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.java:1290)
~[spring-boot-2.6.1.jar:2.6.1]
    at
br.com.fourshopp.FourShoppApplication.main(FourShoppApplication.java:51)
~[classes/:na]
    at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native
Method) ~[na:na]
    at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethod
AccessorImpl.java:77) ~[na:na]
    at
java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(Delegati
ngMethodAccessorImpl.java:43) ~[na:na]
    at java.base/java.lang.reflect.Method.invoke(Method.java:568)
~[na:na]
    at
org.springframework.boot.devtools.restart.RestartLauncher.run(RestartLaunch
er.java:49) ~[spring-boot-devtools-2.6.1.jar:2.6.1]
Caused by: java.util.InputMismatchException: null
    at java.base/java.util.Scanner.throwFor(Scanner.java:939) ~[na:na]
    at java.base/java.util.Scanner.next(Scanner.java:1594) ~[na:na]
    at java.base/java.util.Scanner.nextInt(Scanner.java:2258) ~[na:na]
    at java.base/java.util.Scanner.nextInt(Scanner.java:2212) ~[na:na]
    at
br.com.fourshopp.FourShoppApplication.run(FourShoppApplication.java:63)
~[classes/:na]
    at
org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:767) ~[spring-boot-2.6.1.jar:2.6.1]
    ... 10 common frames omitted

```

```
2022-06-28 10:43:35.309 INFO 6568 --- [ restartedMain]
j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory
for persistence unit 'default'
2022-06-28 10:43:35.312 INFO 6568 --- [ restartedMain]
com.zaxxer.hikari.HikariDataSource      : HikariPool-1 - Shutdown
initiated...
2022-06-28 10:43:35.323 INFO 6568 --- [ restartedMain]
com.zaxxer.hikari.HikariDataSource      : HikariPool-1 - Shutdown
completed.
```

Process finished with exit code 0

Fourshop

3.2 - Solução do erro:

Foram criados dois métodos na classe UtilValidate (dentro do package util) para realizar o tratamento das entradas que exigiam um inteiro, double ou long ao usuário, os métodos verificam se o usuário digitou um inteiro, um double ou um long. Métodos:

```
public static int integerValidate(Scanner scanner) {  
    int opcao = 0;  
  
    try {  
        opcao = scanner.nextInt();  
    } catch (InputMismatchException e) {  
        System.out.println("Digite um valor inteiro!");  
        scanner.next();  
        integerValidate(scanner);  
    }  
  
    return opcao;  
}
```

```

public static double doubleValidate(Scanner scanner) {
    double opcao = 0;

    try {
        opcao = scanner.nextDouble();
    } catch (InputMismatchException e) {
        System.out.println("Digite um valor double!");
        scanner.next();
        doubleValidate(scanner);
    }

    return opcao;
}

```

```

public static Long longValidate(Scanner scanner) {
    Long opcao = 0L;

    try {
        opcao = scanner.nextLong();
    } catch (InputMismatchException e) {
        System.out.println("Digite um valor Long!");
        scanner.next();
        longValidate(scanner);
    }

    return opcao;
}

```

Após a criação dos métodos, os mesmos foram implementados em todas as partes do sistema que pediam um inteiro, um double ou um Long para o usuário, das seguintes formas:

```
int opcao = inputValidate(scanner);
```

```
double salario = scanner.nextDouble();
```

```
Long opcaoProduto = longValidate(scanner);
```

Fourshop



4 - **Erro:** O sistema está aceitando que o usuário adquira uma quantidade de produtos maior em relação ao estoque.

4.1 - Descrição do erro

Quando é requerido a quantidade de produtos desejada, o sistema aceita e deixa a quantidade do estoque negativa:

```
Hibernate: select produto0_.id_produto as id_produ1_10_, produto0_.dt_vencimento as dt_venci2_
1- Bala Preço: 500.0 Estoque = 37 ←
Informe o número do produto desejado:
1
Escolha a quantidade
50 ←
Hibernate: select produto0_.id_produto as id_produ1_10_0_, produto0_.dt_vencimento as dt_venci
Hibernate: select produto0_.id_produto as id_produ1_10_0_, produto0_.dt_vencimento as dt_venci
Hibernate: update tb_produto set dt_vencimento=?, nm_nome=?, vl_preco=?, qtd_quantidade=?, ds_
Produto comprado com sucesso!
Produto(id=1, nome=Bala, quantidade=-13, preco=500.0, setor=1, dataVencimento=2030-03-10)
Deseja outro produto S/N ?
n
Produto(id=1, nome=Bala, quantidade=50, preco=25000.0, setor=1, dataVencimento=2030-03-10)
Gerando nota fiscal...
Fechando a aplicação...
```



Bala (50)
Preço unidade : R\$500
.....R\$ 22.500

TOTAL: R\$22.500

Fourshop



4.2 - Solução do erro

Foi alterado o método `diminuirEstoque` da classe `productService`, o mesmo agora realiza uma verificação se a quantidade informada pelo usuário é maior do que a quantidade do estoque e retorna um booleano:

```
public boolean diminuirEstoque(int quantidade, Produto produto) {  
  
    if(produto.getQuantidade() < quantidade) {  
        return false;  
    }  
  
    produto.setQuantidade(produto.getQuantidade() - quantidade);  
    produtoRepository.save(produto);  
    return true;  
}
```

Assim, o mesmo pode ser verificado durante a venda de um produto:

```
while(true) {  
    if(productService.diminuirEstoque(quantidade, findById)) {  
        System.out.println("Produto comprado com sucesso!");  
        break;  
    }  
    System.out.println("Escolha uma quantidade igual ou menor que o estoque!");  
    quantidade = integerValue(scanner);  
}
```



5 - **Erro:** O cliente não está recebendo um desconto de 10% a cada 500 reais em produtos do setor de mercearia.

5.1 - Descrição do erro

Quando é gerado o cupom fiscal ao cliente, o mesmo não gera um valor com desconto de 10% a cada 500 reais para produtos do setor de mercearia.



Bala (2)

Preço unidade : R\$500

.....R\$ 1.000

TOTAL: R\$1.000



5.2 - Solução do erro

Foi alterado o método “gerarCupomFiscal” da classe UtilMenu, para que ao gerar o valor das compras do setor de mercearia, os mesmo recebam um desconto de 10% a cada 500 reais. Alteração feita entre as linhas 143 e 153:

```
if(produto.getSetor() == 1) {  
    double desconto = produto.getPreco() / 500;  
  
    Math.floor(desconto);  
  
    double descontoTotal = 0.10 * (500 * desconto);  
  
    produto.setPreco(produto.getPreco() - descontoTotal);  
}
```

Após a alteração do método, foi possível obter o desconto:



Bala (2)
Preço unidade : R\$500

.....R\$ 900

TOTAL: R\$900

Fourshopp

6 - **Erro:** ao cadastrar o Cliente (opção 3 do menu principal):

6.1 - Descrição do erro

```
===== BEM-VINDO AO FOURSHOPP =====
1- Sou cliente
2- Área do ADM
3- Seja um Cliente
4- Login funcionário
5- Encerrar
3
Insira seu nome:
nome
Insira seu email:
email
Insira seu celular:
15981500100
Insira sua password:
123456
Insira seu cpf:
123.123.123-12
Insira sua rua:
Rua dos imigrantes
Insira seu cidade:
Insira seu bairro:
Insira seu numero:
100
Insira sua data de nascimento (dd/MM/yyyy):
22/03/2001
```

Os campos onde se pede a cidade, bairro e número requisitados ao usuário são ignorados e não permitem que os mesmos sejam informados. Após preencher os campos possíveis, os mesmos não são adicionados no banco de dados e é retornado o seguinte erro:

```
Error starting ApplicationContext. To display the conditions report re-run your application with 'debug' enabled.
2022-06-25 22:17:05.074 ERROR 47388 --- [ restartedMain] o.s.boot.SpringApplication : Application run failed

java.lang.IllegalStateException: Failed to execute CommandLineRunner
    at org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:770) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.callRunners(SpringApplication.java:751) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:309) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:1301) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:1290) ~[spring-boot-2.6.1.jar:2.6.1]
    at br.com.fourshopp.FourShoppApplication.main(FourShoppApplication.java:47) ~[classes/:na] <4 internal lines>
    at org.springframework.boot.devtools.restart.RestartLauncher.run(RestartLauncher.java:49) ~[spring-boot-devtools-2.6.1.jar:2.6.1]
Caused by: org.springframework.dao.InvalidDataAccessApiUsageException: org.hibernate.TransientPropertyValueException: object references an unsaved transient instance - save the transient instance first
    at org.springframework.orm.jpa.EntityManagerFactoryUtils.convertJpaAccessExceptionIfPossible(EntityManagerFactoryUtils.java:371) ~[spring-orm-5.3.13.jar:5.3.13]
    at org.springframework.orm.jpa.vendor.HibernateJpaDialect.translateExceptionIfPossible(HibernateJpaDialect.java:235) ~[spring-orm-5.3.13.jar:5.3.13]
    at org.springframework.orm.jpa.JpaTransactionManager.doCommit(JpaTransactionManager.java:566) ~[spring-orm-5.3.13.jar:5.3.13]
    at org.springframework.transaction.support.AbstractPlatformTransactionManager.processCommit(AbstractPlatformTransactionManager.java:743) ~[spring-tx-5.3.13.jar:5.3.13]
    at org.springframework.transaction.support.AbstractPlatformTransactionManager.commit(AbstractPlatformTransactionManager.java:711) ~[spring-tx-5.3.13.jar:5.3.13]
    at org.springframework.transaction.interceptor.TransactionAspectSupport.commitTransactionAfterReturning(TransactionAspectSupport.java:654) ~[spring-tx-5.3.13.jar:5.3.13]
    at org.springframework.transaction.interceptor.TransactionAspectSupport.invokeWithinTransaction(TransactionAspectSupport.java:487) ~[spring-tx-5.3.13.jar:5.3.13]
    at org.springframework.transaction.interceptor.TransactionInterceptor.invoke(TransactionInterceptor.java:119) ~[spring-tx-5.3.13.jar:5.3.13]
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:186) ~[spring-aop-5.3.13.jar:5.3.13]
    at org.springframework.dao.support.PersistenceExceptionTranslationInterceptor.invoke(PersistenceExceptionTranslationInterceptor.java:127) ~[spring-tx-5.3.13.jar:5.3.13]
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:186) ~[spring-aop-5.3.13.jar:5.3.13]
    at org.springframework.data.jpa.repository.support.CrudMethodMetadataPostProcessor$CrudMethodMetadataPopulatingMethodInterceptor.invoke(CrudMethodMetadataPostProcessor.java:174) ~[spring-data-jpa-5.3.13.jar:5.3.13]
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:186) ~[spring-aop-5.3.13.jar:5.3.13]
    at org.springframework.aop.framework.ReflectiveMethodInvocation.proceed(ReflectiveMethodInvocation.java:186) ~[spring-aop-5.3.13.jar:5.3.13]
    at org.springframework.aop.framework.JdkDynamicAopProxy.invoke(JdkDynamicAopProxy.java:215) ~[spring-aop-5.3.13.jar:5.3.13]
    at jdk.proxy4/jdk.proxy4.$Proxy4.save(Unknown Source) ~[na:na]
```

Erro:

Error starting ApplicationContext. To display the conditions report re-run your application with 'debug' enabled.

2022-06-25 22:17:05.074 ERROR 47388 --- [restartedMain]

o.s.boot.SpringApplication : Application run failed

```
java.lang.IllegalStateException: Failed to execute CommandLineRunner
    at
org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:770) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.callRunners(SpringApplication.java:751) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.java:309) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.java:1301) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.java:1290) ~[spring-boot-2.6.1.jar:2.6.1]
    at
br.com.fourshopp.FourShoppApplication.main(FourShoppApplication.java:47) ~[classes/:na]
```

```

    at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native
Method) ~[na:na]
    at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(Native
MethodAccessorImpl.java:77) ~[na:na]
    at
java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(D
elegatingMethodAccessorImpl.java:43) ~[na:na]
    at java.base/java.lang.reflect.Method.invoke(Method.java:568)
~[na:na]
    at
org.springframework.boot.devtools.restart.RestartLauncher.run(Restart
Launcher.java:49) ~[spring-boot-devtools-2.6.1.jar:2.6.1]
Caused by:
org.springframework.dao.InvalidDataAccessApiUsageException:
org.hibernate.TransientPropertyValueException: object references an
unsaved transient instance - save the transient instance before
flushing : br.com.fourshopp.entities.Cliente.endereco ->
br.com.fourshopp.entities.Endereco; nested exception is
java.lang.IllegalStateException:
org.hibernate.TransientPropertyValueException: object references an
unsaved transient instance - save the transient instance before
flushing : br.com.fourshopp.entities.Cliente.endereco ->
br.com.fourshopp.entities.Endereco
    at
org.springframework.orm.jpa.EntityManagerFactoryUtils.convertJpaAcce
ssExceptionIfPossible(EntityManagerFactoryUtils.java:371)
~[spring-orm-5.3.13.jar:5.3.13]
    at
org.springframework.orm.jpa.vendor.HibernateJpaDialect.translateExce
ptionIfPossible(HibernateJpaDialect.java:235)
~[spring-orm-5.3.13.jar:5.3.13]
    at
org.springframework.orm.jpa.JpaTransactionManager.doCommit(JpaTransa
ctionManager.java:566) ~[spring-orm-5.3.13.jar:5.3.13]
    at
org.springframework.transaction.support.AbstractPlatformTransactionM
anager.processCommit(AbstractPlatformTransactionManager.java:743)
~[spring-tx-5.3.13.jar:5.3.13]
    at
org.springframework.transaction.support.AbstractPlatformTransactionM
anager.commit(AbstractPlatformTransactionManager.java:711)
~[spring-tx-5.3.13.jar:5.3.13]
    at
org.springframework.transaction.interceptor.TransactionAspectSupport

```

```

.commitTransactionAfterReturning(TransactionAspectSupport.java:654)
~[spring-tx-5.3.13.jar:5.3.13]
    at
org.springframework.transaction.interceptor.TransactionAspectSupport
.invokeWithinTransaction(TransactionAspectSupport.java:407)
~[spring-tx-5.3.13.jar:5.3.13]
    at
org.springframework.transaction.interceptor.TransactionInterceptor.i
nvoke(TransactionInterceptor.java:119)
~[spring-tx-5.3.13.jar:5.3.13]
    at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:186)
~[spring-aop-5.3.13.jar:5.3.13]
    at
org.springframework.dao.support.PersistenceExceptionTranslationInter
ceptor.invoke(PersistenceExceptionTranslationInterceptor.java:137)
~[spring-tx-5.3.13.jar:5.3.13]
    at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:186)
~[spring-aop-5.3.13.jar:5.3.13]
    at
org.springframework.data.jpa.repository.support.CrudMethodMetadataPo
stProcessor$CrudMethodMetadataPopulatingMethodInterceptor.invoke(Cru
dMethodMetadataPostProcessor.java:174)
~[spring-data-jpa-2.6.0.jar:2.6.0]
    at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:186)
~[spring-aop-5.3.13.jar:5.3.13]
    at
org.springframework.aop.interceptor.ExposeInvocationInterceptor.invo
ke(ExposeInvocationInterceptor.java:97)
~[spring-aop-5.3.13.jar:5.3.13]
    at
org.springframework.aop.framework.ReflectiveMethodInvocation.proceed
(ReflectiveMethodInvocation.java:186)
~[spring-aop-5.3.13.jar:5.3.13]
    at
org.springframework.aop.framework.JdkDynamicAopProxy.invoke(JdkDynam
icAopProxy.java:215) ~[spring-aop-5.3.13.jar:5.3.13]
        at jdk.proxy4/jdk.proxy4.$Proxy121.save(Unknown Source)
~[na:na]

```

```

    at
br.com.fourshopp.service.ClienteService.create(ClienteService.java:2
0) ~[classes/:na]
    at
br.com.fourshopp.FourShoppApplication.menuInicial(FourShoppApplicati
on.java:144) ~[classes/:na]
    at
br.com.fourshopp.FourShoppApplication.run(FourShoppApplication.java:
56) ~[classes/:na]
    at
org.springframework.boot.SpringApplication.callRunner(SpringApp
lication.java:767)
~[spring-boot-2.6.1.jar:2.6.1]
    ... 10 common frames omitted
Caused by: java.lang.IllegalStateException:
org.hibernate.TransientPropertyValueException: object references an
unsaved transient instance - save the transient instance before
flushing : br.com.fourshopp.entities.Cliente.endereco ->
br.com.fourshopp.entities.Endereco
    at
org.hibernate.internal.ExceptionConverterImpl.convert(ExceptionConve
rterImpl.java:151) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.internal.ExceptionConverterImpl.convert(ExceptionConve
rterImpl.java:181) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.internal.ExceptionConverterImpl.convert(ExceptionConve
rterImpl.java:188) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.internal.SessionImpl.doFlush(SessionImpl.java:1420)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.internal.SessionImpl.managedFlush(SessionImpl.java:507
) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.internal.SessionImpl.flushBeforeTransactionCompletion(
SessionImpl.java:3299) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.internal.SessionImpl.beforeTransactionCompletion(Sessi
onImpl.java:2434) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.engine.jdbc.internal.JdbcCoordinatorImpl.beforeTransac
tionCompletion(JdbcCoordinatorImpl.java:449)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.resource.transaction.backend.jdbc.internal.JdbcResourc
eLocalTransactionCoordinatorImpl.beforeCompletionCallback(JdbcResour

```

```

ceLocalTransactionCoordinatorImpl.java:183)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.resource.transaction.backend.jdbc.internal.JdbcResourc
eLocalTransactionCoordinatorImpl.access$300(JdbcResourceLocalTransac
tionCoordinatorImpl.java:40)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.resource.transaction.backend.jdbc.internal.JdbcResourc
eLocalTransactionCoordinatorImpl$TransactionDriverControlImpl.commit
(JdbcResourceLocalTransactionCoordinatorImpl.java:281)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.engine.transaction.internal.TransactionImpl.commit(Tra
nsactionImpl.java:101) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.springframework.orm.jpa.JpaTransactionManager.doCommit(JpaTransa
ctionManager.java:562) ~[spring-orm-5.3.13.jar:5.3.13]
    ... 28 common frames omitted
Caused by: org.hibernate.TransientPropertyValueException: object
references an unsaved transient instance - save the transient
instance before flushing :
br.com.fourshopp.entities.Cliente.endereco ->
br.com.fourshopp.entities.Endereco
    at
org.hibernate.engine.spi.CascadingActions$8.noCascade(CascadingActio
ns.java:379) ~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.engine.internal.Cascade.cascade(Cascade.java:169)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.event.internal.AbstractFlushingEventListener.cascadeOn
Flush(AbstractFlushingEventListener.java:159)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.event.internal.AbstractFlushingEventListener.prepareEn
tityFlushes(AbstractFlushingEventListener.java:149)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.event.internal.AbstractFlushingEventListener.flushEver
ythingToExecutions(AbstractFlushingEventListener.java:82)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
    at
org.hibernate.event.internal.DefaultFlushEventListener.onFlush(Defau
ltFlushEventListener.java:39)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]

```

```
        at
org.hibernate.event.service.internal.EventListenerGroupImpl.fireEven
tOnEachListener(EventListenerGroupImpl.java:107)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
        at
org.hibernate.internal.SessionImpl.doFlush(SessionImpl.java:1416)
~[hibernate-core-5.6.1.Final.jar:5.6.1.Final]
... 37 common frames omitted
```

```
2022-06-25 22:17:05.102 INFO 47388 --- [ restartedMain]
j.LocalContainerEntityManagerFactoryBean : Closing JPA
EntityManagerFactory for persistence unit 'default'
2022-06-25 22:17:05.107 INFO 47388 --- [ restartedMain]
com.zaxxer.hikari.HikariDataSource      : HikariPool-1 - Shutdown
initiated...
2022-06-25 22:17:05.118 INFO 47388 --- [ restartedMain]
com.zaxxer.hikari.HikariDataSource      : HikariPool-1 - Shutdown
completed.
```

Fourshopp

6.2 - Solução do erro

Primeiramente visualizei que não havia conversão da String dataNascimento para Date dentro do método “menuCadastroCliente” da classe “UtilMenu”, então criei o um método para converter a mesma para Date (parseStringToDate). Método criado:

```
private static Date parseStringToDate(String data) {  
    SimpleDateFormat format = new SimpleDateFormat( pattern: "dd/MM/yyyy");  
  
    System.out.println("entrou no parseStringToDate!");  
  
    try {  
        Date formatDate = format.parse(data);  
        return formatDate;  
    } catch (ParseException e) {  
        throw new RuntimeException(e);  
    }  
}
```

Após a criação do método, utilizei o mesmo dentro dos parâmetros do Objeto Cliente criado dentro do método “menuCadastroCliente”, além deste método, foi trocado a entrada do scanner de “next” para “nextLine” e colocado com uma duplicidade nas entradas de “nome” e “dataNascimento” para que não houvesse o problema de pular o campo ou não preenchê-lo. A imagem a seguir representa as alterações:


```

public static Cliente menuCadastroCliente(Scanner scanner){

    System.out.println("Insira seu nome: ");
    String nome = scanner.nextLine();
    nome = scanner.nextLine();

    System.out.println("Insira seu email: ");
    String email = scanner.nextLine();

    System.out.println("Insira seu celular: ");
    String celular = scanner.nextLine();

    System.out.println("Insira sua password: ");
    String password = scanner.nextLine();

    System.out.println("Insira seu cpf: ");
    String cpf = scanner.nextLine();

    System.out.println("Insira sua rua: ");
    String rua = scanner.nextLine();

    System.out.println("Insira seu cidade: ");
    String cidade = scanner.nextLine();

    System.out.println("Insira seu bairro: ");
    String bairro = scanner.nextLine();

    System.out.println("Insira seu numero: ");
    int numero = scanner.nextInt();

    System.out.println("Insira sua data de nascimento (dd/MM/yyyy): ");
    String dataNascimento = scanner.nextLine();
    dataNascimento = scanner.nextLine();

    Endereco endereco = new Endereco(rua, cidade, bairro, numero);
    Cliente cliente = new Cliente(nome, email, celular, password, cpf, endereco, parseStringToDate(dataNascimento));
}

```

A última alteração foi feita dentro da classe `ClienteService`, onde foi instanciado e feito a injeção de dependências do repository “`EnderecoRepository`”, esta alteração foi necessária para que fosse possível a utilização do mesmo dentro do método `create` localizado na classe `ClienteService` também. Dentro deste método `create` foi colocado o método para salvar o endereço do cliente que estava ocasionando o erro por não estar sendo salvo antes de salvar o objeto cliente. A seguir a imagem representa o método `create` da classe `ClienteService` alterado:

6 usages

`@Autowired`

```
private ClienteRepository clienteRepository;
```

1 usage

`@Autowired`

```
private EnderecoRepository enderecoRepository;
```

1 usage

```
public Cliente create(Cliente cliente) {  
    enderecoRepository.save(cliente.getEndereco());  
    return clienteRepository.save(cliente);  
}
```

Fourshop



7 - **Erro:** ao acessar a opção de cliente com cpf e senha não cadastrados (opção 1 do menu principal):

7.1 - Descrição do erro

```
===== BEM-VINDO AO FOURSHOPP =====
1- Sou cliente
2- Área do ADM
3- Seja um Cliente
4- Login funcionário
5- Encerrar
1
Insira seu cpf (xxx.xxx.xxx-xx):
243.977.280-11
Insira sua password:
12341234
```

Após informar um cpf e senha não cadastrados o sistema encerra com o seguinte **erro**:

```
Hibernate: select pessoa0_cd_pessoa as cd_pessoa0_9_, pessoa0_nr_celular as nr_celul2_9_, pessoa0_nr_cpf as nr_cpf3_9_, pessoa0_ds_email as ds_email4_9_, pessoa0...
2022-06-29 22:16:55.876 INFO 1328 --- [ restartedMain] ConditionEvaluationReportLoggingListener :

Error starting ApplicationContext. To display the conditions report re-run your application with 'debug' enabled.
2022-06-29 22:16:55.905 ERROR 1328 --- [ restartedMain] o.s.boot.SpringApplication : Application run failed

java.lang.IllegalStateException: Create breakpoint : Failed to execute CommandLineRunner
    at org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:770) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.callRunners(SpringApplication.java:751) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:309) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:1301) ~[spring-boot-2.6.1.jar:2.6.1]
    at org.springframework.boot.SpringApplication.run(SpringApplication.java:1290) ~[spring-boot-2.6.1.jar:2.6.1]
    at br.com.fourshopp.FourShoppApplication.main(FourShoppApplication.java:53) ~[classes/:na] <4 internal lines>
    at org.springframework.boot.devtools.restart.RestartLauncher.run(RestartLauncher.java:49) ~[spring-boot-devtools-2.6.1.jar:2.6.1]
Caused by: org.hibernate.ObjectNotFoundException: Create breakpoint : No row with the given identifier exists: [Cliente#1]
    at br.com.fourshopp.FourShoppApplication.lambda$menuInicial$0(FourShoppApplication.java:104) ~[classes/:na]
    at java.base/java.util.Optional.orElseThrow(Optional.java:403) ~[na:na]
    at br.com.fourshopp.FourShoppApplication.menuInicial(FourShoppApplication.java:104) ~[classes/:na]
    at br.com.fourshopp.FourShoppApplication.run(FourShoppApplication.java:78) ~[classes/:na]
    at org.springframework.boot.SpringApplication.callRunner(SpringApplication.java:767) ~[spring-boot-2.6.1.jar:2.6.1]
    ... 10 common frames omitted

2022-06-29 22:16:55.938 INFO 1328 --- [ restartedMain] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'default'
2022-06-29 22:16:55.942 INFO 1328 --- [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown initiated...
2022-06-29 22:16:55.952 INFO 1328 --- [ restartedMain] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Shutdown completed.

Process finished with exit code 0
```

Erro:

```
Hibernate: select pessoa0_.cd_pessoa as cd_pesso1_9_,
pessoa0_.nr_celular as nr_celul2_9_, pessoa0_.nr_cpf as nr_cpf3_9_,
pessoa0_.ds_email as ds_email4_9_, pessoa0_.cd_endereco as
cd_ender7_9_, pessoa0_.nm_nome as nm_nome5_9_, pessoa0_.ds_senha as
ds_senha6_9_, pessoa0_1_.dt_nascimento as dt_nasci1_0_,
pessoa0_2_.ds_cargo as ds_cargo1_5_, pessoa0_2_.conta_corrente_id as
conta_co6_5_, pessoa0_2_.dt_hire_date as dt_hire_2_5_,
pessoa0_2_.vl_salario as vl_salar3_5_, pessoa0_2_.ds_setor as
ds_setor4_5_, pessoa0_3_.ds_cargo as ds_cargo1_8_,
pessoa0_3_.dt_hire_date as dt_hire_2_8_, pessoa0_3_.cd_funcionario
as cd_func16_8_, pessoa0_3_.horario as horario3_8_,
pessoa0_3_.vl_salario as vl_salar4_8_, case when
pessoa0_1_.cd_cliente is not null then 1 when
pessoa0_2_.cd_funcionario is not null then 2 when
pessoa0_3_.cd_operador is not null then 3 when pessoa0_.cd_pessoa is
not null then 0 end as clazz_ from tb_pessoa pessoa0_ left outer
join tb_cliente pessoa0_1_ on
pessoa0_.cd_pessoa=pessoa0_1_.cd_cliente left outer join
tb_funcionario pessoa0_2_ on
pessoa0_.cd_pessoa=pessoa0_2_.cd_funcionario left outer join
tb_operador pessoa0_3_ on pessoa0_.cd_pessoa=pessoa0_3_.cd_operador
where pessoa0_.nr_cpf=? and pessoa0_.ds_senha=?
2022-06-29 22:16:55.876 INFO 1328 --- [ restartedMain]
ConditionEvaluationReportLoggingListener :
```

Error starting ApplicationContext. To display the conditions report
re-run your application with 'debug' enabled.

```
2022-06-29 22:16:55.905 ERROR 1328 --- [ restartedMain]
```

```
o.s.boot.SpringApplication : Application run failed
```

```
java.lang.IllegalStateException: Failed to execute CommandLineRunner
    at
org.springframework.boot.SpringApplication.callRunner(SpringApplicat
ion.java:770) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.callRunners(SpringApplica
tion.java:751) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.jav
a:309) ~[spring-boot-2.6.1.jar:2.6.1]
    at
org.springframework.boot.SpringApplication.run(SpringApplication.jav
a:1301) ~[spring-boot-2.6.1.jar:2.6.1]
```

```

        at
org.springframework.boot.SpringApplication.run(SpringApplication.jav
a:1290) ~[spring-boot-2.6.1.jar:2.6.1]
        at
br.com.fourshopp.FourShoppApplication.main(FourShoppApplication.java
:53) ~[classes/:na]
        at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Nati
ve Method) ~[na:na]
        at
java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(Nativ
eMethodAccessorImpl.java:77) ~[na:na]
        at
java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(D
elegatingMethodAccessorImpl.java:43) ~[na:na]
        at java.base/java.lang.reflect.Method.invoke(Method.java:568)
~[na:na]
        at
org.springframework.boot.devtools.restart.RestartLauncher.run(Restart
Launcher.java:49) ~[spring-boot-devtools-2.6.1.jar:2.6.1]
Caused by: org.hibernate.ObjectNotFoundException: No row with the
given identifier exists: [Cliente#1]
        at
br.com.fourshopp.FourShoppApplication.lambda$menuInicial$0(FourShopp
Application.java:104) ~[classes/:na]
        at java.base/java.util.Optional.orElseThrow(Optional.java:403)
~[na:na]
        at
br.com.fourshopp.FourShoppApplication.menuInicial(FourShoppApplicati
on.java:104) ~[classes/:na]
        at
br.com.fourshopp.FourShoppApplication.run(FourShoppApplication.java:
78) ~[classes/:na]
        at
org.springframework.boot.SpringApplication.callRunner(SpringApplicat
ion.java:767) ~[spring-boot-2.6.1.jar:2.6.1]
        ... 10 common frames omitted

```

```

2022-06-29 22:16:55.938 INFO 1328 --- [ restartedMain]
j.LocalContainerEntityManagerFactoryBean : Closing JPA
EntityManagerFactory for persistence unit 'default'
2022-06-29 22:16:55.942 INFO 1328 --- [ restartedMain]
com.zaxxer.hikari.HikariDataSource      : HikariPool-1 - Shutdown
initiated...

```

```
2022-06-29 22:16:55.952 INFO 1328 --- [ restartedMain]
com.zaxxer.hikari.HikariDataSource      : HikariPool-1 - Shutdown
completed.
```

```
Process finished with exit code 0
```

Fourshopp

7.2 - Solução do erro

Foi adicionado um try catch para verificar a existência do funcionário:

```
try {  
    this.cliente = clienteService.loadByEmailAndPassword(cpf, password).orElseThrow(() -> new ObjectNotFoundException(1L, "Cliente"));  
} catch (ObjectNotFoundException e) {  
    System.err.println("Cliente não encontrado!\n");  
    menuInicial( opcao: 4);  
}
```

Após a implementação do mesmo, quando o cliente digitar um cpf e uma senha não cadastrados, o sistema irá mandar a seguinte mensagem:

```
Insira seu cpf (xxx.xxx.xxx-xx):  
358.979.180-23  
Insira sua password:  
12341234  
Hibernate: select pessoa0_.cd_pes  
Cliente não encontrado!
```



8 - **Erro:** é possível entrar como um funcionário, chefe (opção 4 do menu principal e 1 do submenu), com um cpf e senha não cadastrados

8.1 - Descrição do erro

Após informar um cpf e senha não cadastrados, o sistema não faz uma validação para verificar se existe o CPF e senha cadastrado e libera as opções do funcionário ao usuário:

```
Insira seu cpf:
159.304.280-93
Insira sua password:
12341234
Hibernate: select pessoa0_
1 - Cadastrar produto
2 - Alterar produto
3 - Excluir produto
4 - Cadastrar operadores
```


Fourshopp

8.2 - Solução do erro

Foi adicionado um try catch para fazer o tratamento do erro e a verificação da existência de um funcionário com um cargo de chefe de seção atribuído a ele:

```
try {
    Optional<Funcionario> chefe = this.funcionarioService.loadByEmailAndPassword(cpf, password);
    if(chefe.get().getCargo().getCd() != Cargo.CHEFE_SECAO.getCd()) {
        System.out.println("Você não possui cargo de chefe de seção!");
        menuInicial( opcao: 4);
    }
} catch (NoSuchElementException e) {
    System.err.println("Chefe não encontrado!");
    menuInicial( opcao: 4);
}
```

Fourshop



9 - **Erro:** Após realizar algumas funções do sistema, o usuário não é redirecionado ao menu

9.1 - Descrição do erro

Em algumas partes do sistema, não existe um retorno para o menu principal, após executar uma determinada função do sistema:

```
1 - Cadastrar produto
2 - Alterar produto
3 - Excluir produto
4 - Cadastrar operadores
Insira uma opção:
1
Digite o nome do produto:
Feijão 400g
Digite a quantidade do produto:
100
Digite o preço do produto:
100
Setores (1 - MERCEARIA / 2 - BAZAR / 3 - ELETRONICOS / 4 - COMERCIAL)
Insira uma opção:
1
Digite a data de vencimento do produto:
10/10/2030
Hibernate: select next_val as id_val from sq_produto for update
Hibernate: update sq_produto set next_val= ? where next_val=?
Hibernate: insert into tb_produto (dt_vencimento, nm_nome, vl_preco, qtd_quantidade, ds_setor, id_produto) values (?, ?, ?, ?, ?, ?)
```

Fourshopp

9.2 - Solução do erro

Foi criado um método na classe UtilMenu chamado home, com este método é possível imprimir o menu principal para o usuário e receber uma nova opção:

```
public static int home(Scanner scanner) {  
    System.out.println("===== BEM-VINDO AO FOURSHOPP =====");  
    System.out.println("1- Sou cliente " +  
        "\n2- Área do ADM" +  
        "\n3- Seja um Cliente " +  
        "\n4- Login funcionário" +  
        "\n5- Encerrar ");  
    int opcao = opcaoMenu(scanner, inicio: 1, fim: 6);  
  
    return opcao;  
}
```

Este foi implementado nos pontos do sistema onde não existiam mais opções para o usuário.

Fourshop



10 - **Erro:** O sistema não finaliza ao digitar a opção 5

10.1 - Descrição do erro

Após digitar a opção 5 o sistema não se encerra:

```
===== BEM-VINDO AO FOURSHOPP =====  
1- Sou cliente  
2- Área do ADM  
3- Seja um Cliente  
4- Login funcionário  
5- Encerrar  
5  
5  
5
```

Fourshopp

10.2 - Solução do erro

Foi adicionado um `System.exit(0)` para finalizar o sistema na opção 5:

```
if(opcao == 5) {  
    System.out.println("Sistema finalizado!");  
    System.exit(status: 0);  
}
```