

PROJETO CC1612 – BANCO QUEM POUPA TEM

PROFESSOR: DANILO PERICO

ALUNO: LUCAS KRAUS MONTEIRO ALVES

R.A: 22.120.061-1

INFORMAÇÕES SOBRE A PROGRAMAÇÃO DO SISTEMA

def menu_principal():

IDE

Programa

Olá, seja bem vindo ao Banco Quem Poupa Tem! Abaixo está nosso menu de opções:

```
1 - Abrir uma conta
2 - Cancelar Minha Conta
3 - Debitar Valor da Conta
4 - Realizar Depósito
5 - Ver Saldo
6 - Ver Extrato
0 - Sair do Sistema
```

Por favor, digite o número da operação que deseja realizar:

- Função principal que é chamada ao iniciar o programa. A variável "opcao" irá receber um número de 0 a 6, que chamará a função responsável pela respectiva opção solicitada.

```
#Função do Menu Principal.
def menu_principal():
    print("Olá, seja bem vindo ao Banco Quem Poupa Tem! Abaixo está nosso menu de opções: \n")
    print("1 - Abrir uma conta \n"
          "2 - Cancelar Minha Conta \n"
          "3 - Debitar Valor da Conta \n"
          "4 - Realizar Depósito \n"
          "5 - Ver Saldo \n"
          "6 - Ver Extrato \n"
          "0 - Sair do Sistema \n")
    opcao = int(input("Por favor, digite o número da operação que deseja realizar: "))
    print("\n")

    #Condições de execução das opções do menu.
    if (opcao < 0) or (opcao > 6):
        print("\n")
        print("Número de opção inválida \n")
        menu_principal()
    elif opcao == 0:
        opcao0()
    elif opcao == 1:
        opcao1()
    elif opcao == 2:
        opcao2()
    elif opcao == 3:
        opcao3()
    elif opcao == 4:
        opcao4()
    elif opcao == 5:
        opcao5()
    elif opcao == 6:
        opcao6()
```

def opçao1():

Parte 1

Programa

```
Por favor, digite o número da operação que deseja realizar: 1

Por favor, digite seu CPF com 11 dígitos para que possamos dar continuidade com seu cadastro: 51971983896
Tipos de Contas:

1 - CONTA SALÁRIO - Desconto de 5% por débito e sem possibilidade de saldo negativo.
2 - CONTA COMUM - Desconto de 3% por débito e máximo de -R$500,00 de saldo negativo.
3 - CONTA PLUS - Desconto de 1% por débito e máximo de -R$5000,00 de saldo negativo.

Digite o número respectivo do tipo de conta que deseja abrir (1, 2 ou 3): 3
Digite seu primeiro nome: Lucas
Digite seu sobrenome: Kraus
Escolha uma senha de 6 dígitos: 270302
Insira o valor inicial para depósito em sua conta: R$5000
```

- Função chamada ao digitar 1 no menu principal. A variável "opçao1" irá receber todas as informações para cadastro do cliente e irá armazenar em um arquivo.txt, cujo nome será o CPF do mesmo, e as linhas serão as informações (Tipo de Conta, Nome, Sobrenome, Senha e Valor Inicial da Conta).

def opcao1():

Parte 2

IDE

```
#Função para abertura de uma nova conta (Opção 1).
def opcao1():
    cpf = int(input("Por favor, digite seu CPF com 11 dígitos para que possamos dar continuidade com seu cadastro: "))

    arquivo_cliente = open("{}_txt".format(cpf), "w")

    #Exibe os tipos de conta para que o cliente possa escolher.
    print("Tipos de Contas: \n")
    print("1 - CONTA SALÁRIO - Desconto de 5% por débito e sem possibilidade de saldo negativo. \n"
          "2 - CONTA COMUM - Desconto de 3% por débito e máximo de -R$500,00 de saldo negativo. \n"
          "3 - CONTA PLUS - Desconto de 1% por débito e máximo de -R$5000,00 de saldo negativo. \n")

    #Coleta de dados do cliente para cadastro.
    tipo = int(input("Digite o número respectivo do tipo de conta que deseja abrir (1, 2 ou 3): "))
    nome = input("Digite seu primeiro nome: ")
    sobrenome = input("Digite seu sobrenome: ")
    senha = int(input("Escolha uma senha de 6 dígitos: "))
    valor_inicial = float(input("Insira o valor inicial para depósito em sua conta: R$"))

    #Guarda os dados em um arquivo.
    arquivo_cliente.write("%d\n%s\n%s\n%d\n%f\n" % (tipo, nome, sobrenome, senha, valor_inicial))
    arquivo_cliente.close()

    #Retorna ao Menu Principal.
    print("\n")
    print("Pronto! Seu cadastro foi realizado com sucesso.")
    print("Você será redirecionado para o menu principal. \n")
    menu_principal()
```

def opçao2():

Parte 1

Programa

```
Por favor, digite o número da operação que deseja realizar: 2
```

```
Poxa, uma pena que não tenha gostado do nosso serviço.  
Digite seu CPF para que possamos cancelar sua conta:
```

- Função chamada ao digitar 2 no menu principal. A variável “cpf” irá receber o número de CPF do usuário e então, a função irá excluir o arquivo com o nome do respectivo CPF digitado.

def opcao2():

Parte 2

IDE

```
#Função que exclui a conta do cliente do sistema do banco (Opção 2).
def opcao2():
    print("\n")
    print("Poxa, uma pena que não tenha gostado do nosso serviço.")
    cpf = int(input("Digite seu CPF para que possamos cancelar sua conta: "))

    #Verifica se existe o arquivo com o CPF do cliente para poder excluir.
    if os.path.exists("{} .txt".format(cpf)):
        os.remove("{} .txt".format(cpf))
        print()
        print("Sua conta foi cancelada com sucesso!\n")
        menu_principal()

    #Caso não encontre, o programa irá retornar o usuário para o menu principal.
    else:
        print("Não localizamos nenhuma conta vinculada a este CPF em nosso sistema.")
        menu_principal()
```

def opcao3():

Parte 1

Programa

```
Por favor, digite o número da operação que deseja realizar: 3
```

```
Por favor, digite seu CPF com 11 dígitos: 51971983896
```

```
Por favor, digite a sua senha de 6 dígitos: 270302
```

```
O seu saldo atual em conta é R$5000.00
```

```
Digite o valor que gostaria de debitar da sua conta: R$3000
```

```
Taxa de Débito = 1%
```

```
Operação realizada com sucesso!
```

```
Seu novo saldo é R$1950.00
```

```
Você será redirecionado ao menu principal.
```

- Função chamada ao digitar 3 no menu principal. A variável "cpf" irá abrir o arquivo, a variável "senha" será verificada na linha 3 do arquivo, e então de acordo com os critérios do tipo de conta (podendo ter ou não saldo negativo), o débito será realizado.

def opçao3():

Parte 2

IDE

```
def opçao3():
    global x
    global year
    global month
    global day
    global hour
    global minute
    #Solicite o CPF do cliente.
    cpf = int(input("Por favor, digite seu CPF com 11 dígitos: "))

    #Procura o arquivo do CPF digitado.
    if os.path.exists("{} .txt".format(cpf)):
        cadastro = open("{} .txt".format(cpf), "r")

        leitura_cadastro = cadastro.readlines()

        cadastro.close()

    #Retorna o cliente para o menu principal caso não encontre arquivo com o CPF digitado.
    else:
        print("\n")
        print("Não localizamos conta vinculada a este CPF. Você será redirecionado para o Menu Principal.\n")
        menu_principal()

    senha = int(input("Por favor, digite a sua senha de 6 dígitos: "))

    #Confere se a senha está correta e mostra o saldo atual.
    if senha == (int(leitura_cadastro[3])):
        global valor_debito
        #Exibe o saldo atual da conta do cliente.
        print("O seu saldo atual em conta é R${:.2f} \n" % float(leitura_cadastro[4]))
        #Função que faz a subtração do saldo pelo valor de débito
        valor_debito = float(input("Digite o valor que gostaria de debitar da sua conta: R$"))

    #Caso a senha não esteja correta, a função será iniciada novamente.
    else:
        print("A senha digitada está incorreta! Por favor, repita toda a operação. \n")
        opçao3()
```


def opcao3():

```
#Operação a ser realizada no tipo de conta SALÁRIO.
if (int(leitura_cadastro[0])) == 1:
    print("\n")
    print("Taxa de Débito = 5%")
    #Este tipo de conta não pode ser saldo negativo.
    if valor_debito > (float(leitura_cadastro[4])):
        print("Operação não permitida para seu tipo de conta. Saldo negativo não possível.")
        print("Por favor, repita o processo e atenda ao critério do seu tipo de conta.\n")
        opcao3()
    #Realiza a subtração caso o critério do tipo de conta não seja violado.
    else:
        #Escreve as informações junto ao novo saldo.
        leitura_cadastro[4] = (float(leitura_cadastro[4])) - valor_debito - ((float(leitura_cadastro[4]))*0.05)
        cadastro = open("{}txt".format(cpf),"w")

        cadastro.write("%s%s%s%s%f" % (leitura_cadastro[0],leitura_cadastro[1],leitura_cadastro[2],leitura_cadastro[3],leitura_cadastro[4]))

        cadastro.close()

        #Escreve a data, hora e valor da operação.
        cadastro = open("{}extrato.txt".format(cpf),"a")

        cadastro.write("Data: {}-{}-{} Hora: {}:{} Retirou: -R${:.2f} Saldo Final: R${:.2f}\n".format(day, month, year, hour, minute%(valor_debito,leitura_cadastro[4]))

        cadastro.close()

    #Exibe o saldo e retorna para o menu principal.
    print("\n")
    print("Operação realizada com sucesso!")
    print("Seu novo saldo é R${:.2f}% (leitura_cadastro[4]))
    print("Você será redirecionado ao menu principal.\n")
    menu_principal()
```

def opcao3():

```
#Operação a ser realizada no tipo de conta COMUM.
if (int(leitura_cadastro[0])) == 2:
    print("\n")
    print("Taxa de Débito = 3%")
    #O saldo negativo deste tipo não pode ser menor que -R$500,00.
    if (float(leitura_cadastro[4])) - valor_debito - ((float(leitura_cadastro[4]))*0.03) < -500:
        print("Seu saldo negativo não pode ser superior a -R$500,00")
        print("Por favor, repita o processo e atenda ao critério do seu tipo de conta.\n")
        opcao3()
    #Realiza a subtração caso o critério do tipo de conta não seja violado
    else:
        #Escreve as informações junto ao novo saldo.
        leitura_cadastro[4] = (float(leitura_cadastro[4])) - valor_debito - ((float(leitura_cadastro[4]))*0.03)
        cadastro = open("{}_txt".format(cpf),"w")

        cadastro.write("%s%s%s%s%f" % (leitura_cadastro[0],leitura_cadastro[1],leitura_cadastro[2],leitura_cadastro[3],leitura_cadastro[4]))

        cadastro.close()

        #Escreve a data, hora e valor da operação.
        cadastro = open("{}extrato.txt".format(cpf),"a")

        cadastro.write("Data: {}-{}-{} Hora: {}:{} Retirou: -R${:.2f} Saldo Final: R${:.2f}\n".format(day, month, year, hour, minute,valor_debito,leitura_cadastro[4]))

        cadastro.close()

        #Exibe o saldo e retorna para o menu principal.
        print("\n")
        print("Operação realizada com sucesso!")
        print("Seu novo saldo é R${:.2f}% (leitura_cadastro[4]))
        print("Você será redirecionado ao menu principal.\n")
        menu_principal()
```

def opçao4():

Parte 1

Programa

```
Por favor, digite o número da operação que deseja realizar: 4
```

```
Por favor, digite seu CPF com 11 dígitos: 51971983896
```

```
Digite o valor que gostaria de depositar em sua conta: R$2000
```

```
Operação realizada com sucesso!
```

```
Seu novo saldo é R$3950.00
```

```
Você será redirecionado para o menu principal.
```

- Função chamada ao digitar 4 no menu principal. A variável "cpf" irá abrir o arquivo, e irá somar o valor digitado na variável "valor_somar" a linha 4 do arquivo, que é onde se localiza o saldo da conta.

def opcao4():

```
def opcao4():
    #Variáveis de data e hora para registro no extrato.
    global x
    global year
    global month
    global day
    global hour
    global minute
    #Solicita o CPF do cliente.
    cpf = int(input("Por favor, digite seu CPF com 11 dígitos: "))

    #Procura o arquivo do CPF digitado.
    if os.path.exists("{} .txt".format(cpf)):
        cadastro = open("{} .txt".format(cpf), "r")

        leitura_cadastro = cadastro.readlines()

        cadastro.close()

        #Solicita o valor a ser depositado em conta.
        global valor_somar
        valor_somar = float(input("Digite o valor que gostaria de depositar em sua conta: R$"))

        #Realiza a soma do saldo atual ao valor depositado.
        leitura_cadastro[4] = (float(leitura_cadastro[4])) + valor_somar

        #Escreve as informações junto ao novo saldo.
        cadastro = open("{} .txt".format(cpf), "w")

        cadastro.write("%s%s%s%s%f\n" % (leitura_cadastro[0], leitura_cadastro[1], leitura_cadastro[2], leitura_cadastro[3], leitura_cadastro[4]))

        cadastro.close()

        #Escreve a data, hora e valor da operação.
        cadastro = open("{}extrato.txt".format(cpf), "a")

        cadastro.write("Data: {}-{}-{} Hora: {}:{} Depositou: +R${:.2f} Saldo Final: R${:.2f}\n".format(day, month, year, hour, minute, valor_somar, leitura_cadastro[4]))

        cadastro.close()

        #Conclui a operação e retorna para o menu principal.
        print("\n")
        print("Operação realizada com sucesso!")
        print("Seu novo saldo é R${:.2f}" % (leitura_cadastro[4]))
        print("Você será redirecionado para o menu principal.\n")
        menu_principal()
```

def opcao5():

Parte 1

Programa

```
Por favor, digite o número da operação que deseja realizar: 5
```

```
Por favor, digite seu CPF com 11 dígitos: 51971983896
```

```
Digite a sua senha de 6 dígitos: 270302
```

```
O seu saldo atual em conta é de R$3950.00
```

```
Você será redirecionado para o menu principal.
```

- Função chamada ao digitar 5 no menu principal. A variável "cpf" irá abrir o arquivo, a variável "senha" será verificada e caso esteja igual a linha 3 do arquivo, será possível verificar o saldo da conta.

def opcao5():

IDE

```
def opcao5():
    cpf = int(input("Por favor, digite seu CPF com 11 dígitos: "))

    #Procura o arquivo do CPF digitado.
    if os.path.exists("{} .txt".format(cpf)):
        cadastro = open("{} .txt".format(cpf), "r")

        leitura_cadastro = cadastro.readlines()

        cadastro.close()

    #Caso o CPF digitado não seja encontrado, o cliente será redirecionado para o menu principal.
    else:
        print("Não localizamos conta vinculada a este CPF. Por favor, digite-o novamente. \n")
        menu_principal()

    senha = int(input("Digite a sua senha de 6 dígitos: "))
    #Verifica se a senha digitada está correta.
    if senha == (int(leitura_cadastro[3])):
        print("\n")
        print("O seu saldo atual em conta é de R${:.7s}% (leitura_cadastro[4])".format(leitura_cadastro[4]))
        print("Você será redirecionado para o menu principal.\n")
        menu_principal()
    #Caso a senha digitada esteja incorreta, a função será iniciada novamente.
    else:
        print("A senha digitada está incorreta. Por favor, repita toda a operação.")
        opcao5()
```

def opcao6():

Parte 1

Programa

```
Por favor, digite o número da operação que deseja realizar: 6

Por favor, digite seu CPF com 11 dígitos: 51971983896
Digite a sua senha de 6 dígitos: 270302

TIPO DE CONTA: Plus.
NOME: Lucas
SOBRENOME: Kraus
CPF: 51971983896

=====
Data: 25-4-2020 Hora: 0:30 Retirou: -R$3000.00 Saldo Final: R$1
950.00

Data: 25-4-2020 Hora: 0:30 Depositou: +R$2000.00 Saldo Final: R
$3950.00

Saldo Atual: R$3950.00
=====
Você será redirecionado para o menu principal.
```

*O extrato se encontra cortado pois o visualizador que estava utilizando tinha um limite de espaço, porém, no programa funciona perfeitamente.

- Função chamada ao digitar 6 no menu principal. A variável "cpf" irá abrir o arquivo, a variável "senha" será verificada e caso esteja igual a linha 3 do arquivo, será possível verificar o extrato da conta. O arquivo extrato será aberto em modo leitura para que seja possível printar os operações.

def opcao6():

Parte 2

IDE

```
def opcao6():
    cpf = int(input("Por favor, digite seu CPF com 11 dígitos: "))

    #Procura o arquivo do CPF digitado.
    if os.path.exists("{} .txt".format(cpf)):
        cadastro = open("{} .txt".format(cpf), "r")

        leitura_cadastro = cadastro.readlines()

        cadastro.close()

    #Retorna o usuário para o menu principal caso não encontre arquivo com o CPF digitado.
    else:
        print("Não localizamos conta vinculada a este CPF. Por favor, digite-o novamente. \n")
        menu_principal()

    senha = int(input("Digite a sua senha de 6 dígitos: "))
    #Verifica se a senha digitada está correta.
    if senha == (int(leitura_cadastro[3])):
        cadastro = open("{} .txt".format(cpf), "r")

        leitura_cadastro = cadastro.readlines()

        cadastro.close()
        if (int(leitura_cadastro[0])) == 1:
            print("\n")
            print("TIPO DE CONTA: Comum.")
        elif (int(leitura_cadastro[0])) == 2:
            print("\n")
            print("TIPO DE CONTA: Salário.")
        elif (int(leitura_cadastro[0])) == 3:
            print("\n")
            print("TIPO DE CONTA: Plus.")
```

```
        leitura_cadastro[1] = leitura_cadastro[1].replace("\n", "")
        leitura_cadastro[2] = leitura_cadastro[2].replace("\n", "")

        #Exibe nome, sobrenome e CPF do cliente no extrato.
        print("NOME: %s" % leitura_cadastro[1])
        print("SOBRENOME: %s" % leitura_cadastro[2])
        print("CPF: %d \n" % cpf)
        #Começo do bloco de extrato.
        print("=====")

        #Abre o arquivo de extrato e em seguida imprime o mesmo.
        extrato = open("{}extrato.txt".format(cpf), "r")

        extrato_leitura = extrato.readlines()

        extrato.close()

        for linha in extrato_leitura:
            print("%s" % linha)
        #Fim do bloco de extrato.
        print("\n")
        print("Saldo Atual: R$%.7s" % leitura_cadastro[4])
        print("=====")
        print("Você será redirecionado para o menu principal.")
        print("\n")

        #Redireciona para o menu principal após exibir o extrato
        menu_principal()
```



OBRIGADO POR VISUALIZAR!