


ACADEMIA Python

ATIVIDADE PRÁTICA



MÓDULO 003

Python- OOP

Python - Atividade Prática

Atividade prática

Academia Python

Utilizando os conhecimentos adquiridos na aula, onde foi abordado os princípios da orientação a Objetos criação de classe e herança e CRUD, iremos realizar exercícios neste paradigma.

Nos slides seguintes estarão as instruções para realização desta atividade prática.

Bons estudos!

Academia Python

Iremos aplicar as outras duas funções do nosso CRUD **Update** e **delete**

Vocês irão aplicar inicialmente para realizar os testes com essas outras duas funções no exercício do dia 16/11 onde foi criado uma classe conta, um documento controller, e um documento main, estes que estavam funcionando apenas com **create** e **read**

Academia Python

-Crie uma função update recebendo um atributo referenciando a nossa classe Conta .

```
def update(conta update:Conta):
```

Crie uma variável **lista_contas** recendo uma lista vazia

Crie uma variável **contas** abrindo o arquivo txt `contas = open('contas.txt', 'r')`

Crie um for com variável de **conta** percorrendo variável **contas**

Crie uma variável **conta_limpa** recebendo **conta** e a nossa função interna do python **.strip()**

Crie uma variavel **conta_objeto** recebendo **conta_limpa** e a nossa função interna do python **.split(';')**

Crie um **if** defina como variavel **conta_update** o mesmo e atributo da função **conta_update**, acesse **.numero**, este é o atributo interno da nossa classe, se for igualdade para índice **[1]** de **conta_objeto**

```
if conta update.numero == int(conta objeto[1]):
```

Academia Python

-Dentro do if escreva variável **lista_contas** recebendo a função interna **.append()** dentro dela converta nossa variavel **conta_objeto** para string concatenada com uma quebra de linha

Faça um else e dentro chame a variável **lista_contas** e atribua a variavel do for **conta** dentro da funcao **.append()**

Chame a função interna do **python .close()** para fechar o arquivo

Crie uma variável **contas** escrevendo no arquivo txt

- Chame essa variável, chamando a função interna do python que escreve **.writelines()**

- Coloque a variável **lista_contas** dentro de parênteses da função **.writelines()**

Chame a função interna do **python .close()** para fechar o arquivo

Academia Python

-Crie uma função update recebendo um atributo referenciando a nossa classe Conta .

```
def delete(numero conta):
```

Crie uma variável **lista_contas** recendo uma lista vazia

Crie uma variável **contas** abrindo o arquivo txt `contas = open('contas.txt', 'r')`

Crie um for com variável de **conta** percorrendo variável **contas**

Crie uma variável **conta_limpa** recebendo **conta** e a nossa função interna do python **.strip()**

Crie uma variavel **conta_objeto** recebendo **conta_limpa** e a nossa função interna do python **.split(';')**

Crie um **if** defina como variavel **conta_update** o mesmo e atributo da função **conta_update**, acesse **.numero**, este é o atributo interno da nossa classe, se for diferente para i índice **[1]** de **conta_objeto**

```
if numero conta != int(conta objeto[1]):
```

Academia Python

-Dentro do if escreva variável **lista_contas** recebendo a função interna **.append()** dentro del atribua variavel **conta**

Chame a função interna do **python .close()** para fechar o arquivo

Crie uma variável **contas** escrevendo no arquivo txt

- Chame essa variável, chamando a função interna do python que escreve **.writelines()**

- Coloque a variável **lista_contas** dentro de parênteses da função **.writelines()**

Chame a função interna do **python .close()** para fechar o arquivo

Academia Python

No documento main realize os testes com as nossas funções update e delete

#goZuplae2022