

1) (2,5 Pontos) Quais os comandos necessários para criação de um botão no seu código?

São necessários alguns comandos como o Rectangle com os Points como parâmetro, para definir o tamanho do retângulo solicitado, foi usado também alguns comandos secundários para estilização, como setFill para colorir a área, Text para inserir um texto no botão, além do setSize e setStyle para definir o tamanho e estilo da fonte. Quando o botão for criado, é necessário aplicar um draw() para “desenhá-lo” em si na página. Agora, para torná-lo clicável, utiliza-se a função checkMouse() para ver se houve algum clique, assim, recebe as coordenadas X e Y do clique e determina-se uma área clicável em cima do botão, de acordo com seu tamanho

Exemplo:

```
botao_verificar = Rectangle(Point(7.75, 8), Point(11.75, 9.2))
botao_verificar.setFill('grey')
botao_verificar.draw(win)
texto_verificar = Text(botao_verificar.getCenter(), 'Verificar Estoque')
texto_verificar.setSize(18)
texto_verificar.setStyle('bold')
texto_verificar.draw(win)

click = win.checkMouse()
if click:
    x = click.getX()
    y = click.getY()
    if 7.75 <= x <= 11.75 and 8 <= y <= 9.2:
        janela_verificar_estoque()
```

2) (2,5 Pontos) Como, em seu projeto, tu acessas o CSV como sua base de dados?

Utiliza-se as funções de ler um arquivo usando o open com o caminho do csv e o mode 'r' ou 'w', dependendo da função necessária. No caso para ler os dados, utiliza-se o readlines() para ler linha por linha do csv, já que cada linha tem o ID, Nome, Valor e Quantidade de determinado produto, separado por ','. Desse modo, realiza-se um for para percorrer as linhas e um strip e split(',') para remover quebras de página e separá-los por virgula num Array.

```
def ler_estoque():
    with open('./media/estoque.csv', mode='r') as arq:
        linhas = arq.readlines()
        linhas_sem_cabecalho = linhas[1:]
        estoque = []
        for linha in linhas_sem_cabecalho:
            dados = linha.strip().split(',')
            estoque.append(dados)
        return estoque
```

a) Qual comando foi utilizado para Ler um arquivo CSV?

Foi utilizado o comando with open() as arq, com mode='r' (read), além do readlines(), for e funções de manipulação de strings como strip() e split(','), como explicado acima.

b) Qual comando foi utilizado para escrever um arquivo CSV?

Foi utilizado o comando with open() as arq, com mode='w' (write), além do newline="" para desabilitar a quebra de linha. Também foi criada uma variável writer que tem a função de “escrever” no arquivo em si, juntamente com o writerow(), para escrever fileira por fileira.

Exemplo:

```
def atualizar_estoque(estoque):  
    with open('./media/estoque.csv', mode='w', newline='') as arq:  
        writer = csv.writer(arq)  
        writer.writerow(['ID', 'Nome', 'Valor', 'Quantidade'])  
        writer.writerows(estoque)
```

3) (2,5 Pontos) Qual comando é utilizado para identificar o clique do mouse no botão?

Seria o comando de checkMouse() para verificar se houve um clique, utilizando-se juntamente o comando de getX() e getY() para retornar as coordenadas clicadas.

Exemplo:

```
while True:  
    click = win.checkMouse()  
    if click and 4 <= click.getX() <= 6 and 2 <= click.getY() <= 3:  
        win.close()  
        return
```

4) (2,5 Pontos) Indique alguma linha do código que gera mensagem de erro ao usuário quando algo lançado está incorreto.

No meu código, utiliza-se mensagens de erro para validação nas telas de “Cadastro de Peça” para verificar se cada input está preenchido da forma devida e também na tela de “Realizar Compra” quando o usuário informa a quantidade desejada de determinado produto, para alertar se a quantidade inserida é maior que a no estoque ou é um valor negativo. Ambas funcionam do mesmo método, inicia-se um Text vazio e recebe um erro caso o usuário cometa um erro.

Exemplo “Cadastro de Peça”:

```
texto_erro = Text(Point(5, 3.5), "")  
texto_erro.setFill('red')
```

O texto começa vazio, e recebe um texto dependendo do erro realizado.

```
if nome == "":  
    texto_erro.setText('O nome não pode ser vazio!')
```

Nesse exemplo, quando o usuário não digita o Nome do produto, o texto_erro recebe o a string de texto e o mostra na tela, assim que o botão de enviar é clicado.