



Situação Finalizada

Iniciado quarta-feira, 5 jun. 2024, 19:47

Concluído quarta-feira, 5 jun. 2024, 19:53

Duração 5 minutos 44 segundos

Notas 15,00/15,00

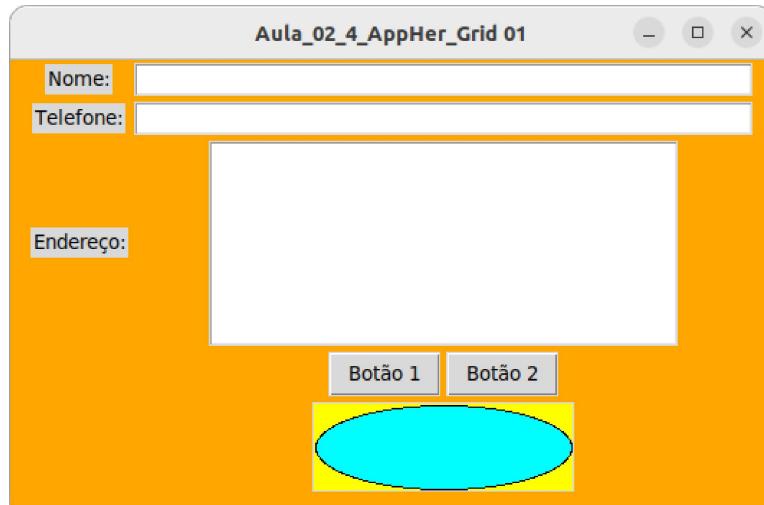
Nota 10,00 de um máximo de 10,00(100%)



Questão 1

Correto

Atingiu 1,00 de 1,00

Dado o código abaixo, responda:

Janela.py

```
01... from tkinter import *
02...
03... #####
04...
05... class Janela(Tk):
06...     def __init__(self, Str="Janela", px=0, py=0, dx=640, dy=480, cor="ligthgray"):
07...         super().__init__()
08...         super().title(Str)
09...         super().geometry("%dx%d+%d+%d" % (dx, dy, px, py))
10...         super().configure(bg=cor)
11...         self.inicialize()
12...
13...     def inicialize(self):
14...         Lb1=Label(self, text="Nome:")
15...         Lb2=Label(self, text="Telefone:")
16...         Lb3=Label(self, text="Endereço:")
17...
18...         Et1=Entry(self, width=52)
19...         Et2=Entry(self, width=52)
20...
21...         Txt1=Text(self, height=8, width=40)
22...
23...         Bt1=Button(self, text='Botão 1')
24...         Bt2=Button(self, text='Botão 2')
25...
26...         Cnv=Canvas(self, bd=0, bg="yellow", width=182, height=62)
27...         Cnv.create_oval(2, 2, 180, 60, fill = "cyan")
28...
29...         Lb1.grid(row=0, column=0, padx=2, pady=2)
```



```

30...
31...
32...
33...
34...
35...
36...
37...
38...     Cnv.grid(row=4, column=1, columnspan=2, padx=2, pady=2)
39...
40... #####

```

Aula_02_4_AppHer_Grid.py

```

41... from Janela import Janela
42...
43... #####
44...
45... Jan1 = Janela("Minha janela", 400, 200, 530, 310, "orange")
46... Jan1.mainloop()
47...
48... #####

```

**Quais as alternativas corretas?**

Escolha uma ou mais:

- a. Na linha 09 do programa há um erro nos parâmetros, o correto seria:
super(Janela, self).geometry(dx, dy, px, py)

- b. Na linha 10 do programa há um erro no Horizontal Gap, o correto seria:
super(Janela, self).configure(cor=laranja)
- c. Na linha 09 do programa há um comando alternativo no Python 2, pode-se utilizar:
super(Janela, self).geometry("%dx%d+%d+%d" % (dx, dy, px, py))
- d. Na linha 10 do programa há um erro no comando, o correto seria:
self.setConfiguracao(bg=cor)
- e. Na linha 14 do programa o comando serve para alocar um Label e atribuir a ele o rótulo:
"Nome:"



Sua resposta está correta.

As respostas corretas são:

Na linha 09 do programa há um comando alternativo no Python 2, pode-se utilizar:
super(Janela, self).geometry("%dx%d+%d+%d" % (dx, dy, px, py))

,
Na linha 14 do programa o comando serve para alocar um Label e atribuir a ele o rótulo:
"Nome:"

Questão 2

Correto

Atingiu 1,00 de 1,00

Na questão anterior um comentário na linha 24 causa o seguinte efeito:

Escolha uma ou mais:

- a. O botão 2 fica com rótulo vazio
- b. Ocorre um erro de execução na linha 35
- c. Ocorre um erro de execução na linha 36 ✓
- d. O programa continua funcionando normalmente exceto pelo fato do botão 2 não aparecer na tela
- e. Ocorre o seguinte erro: NameError: name 'Bt2' is not defined ✓



Sua resposta está correta.

As respostas corretas são: Ocorre um erro de execução na linha 36, Ocorre o seguinte erro: NameError: name 'Bt2' is not defined

Questão 3

Correto

Atingiu 1,00 de 1,00

Em relação ao código apresentado anteriormente, quais as alternativas estão corretas?

Escolha uma ou mais:

- a. Na linha 19 há um erro, o correto seria: Et2=Textfield(width=52)
- b. O comando da linha 19 cria uma caixa de digitação de Strings ✓
- c. Na linha 21 o correto seria: Txt1=TextArea(altura=8, largura=40)
- d. Na linha 21 não há necessidade de transformar o objeto **Txt1** em variável global, pois ele não será utilizado fora do contexto em que foi criado. ✓
- e. O comando da linha 23 cria um botão e atribui a ele um evento, por isso ele obrigatoriamente deve ser uma variável global.



Sua resposta está correta.

As respostas corretas são: O comando da linha 19 cria uma caixa de digitação de Strings, Na linha 21 não há necessidade de transformar o objeto **Txt1** em variável global, pois ele não será utilizado fora do contexto em que foi criado.

Questão 4

Correto

Atingiu 1,00 de 1,00

No código apresentado anteriormente pode-se afirmar o seguinte a respeito do componente *Canvas* :

Escolha uma ou mais:

- a. É um componente gráfico que pode ser utilizado em Janelas na biblioteca Tkinter do Python ✓
- b. O comando da linha 27 cria um círculo com raio de 180 pixels no Canvas.
- c. O comando da linha 26 determina que o fundo do Canvas seja amarelo ✓
- d. O comando da linha 27 cria uma oval com borda de cor **Cyan** e preenchimento verde
- e. O comando da linha 26 determina que a largura do Canvas seja 182 pixels e sua altura 62 pixels. ✓



Sua resposta está correta.

As respostas corretas são: É um componente gráfico que pode ser utilizado em Janelas na biblioteca Tkinter do Python, O comando da linha 26 determina que o fundo do Canvas seja amarelo, O comando da linha 26 determina que a largura do Canvas seja 182 pixels e sua altura 62 pixels.



Questão 5

Correto

Atingiu 1,00 de 1,00

Sobre o GridLayout, apresentado no código anterior, é verdadeiro afirmar que:

Escolha uma ou mais:

- a. Os parâmetros apresentados no comando da linha 29 são:
Linha, Coluna, posição x, posição y
- b. Os parâmetros apresentados no comando da linha 29 são:
Linha, Coluna, separação entre componentes no eixo x, separação entre componentes no eixo y ✓
- c. Na linha 32 o parâmetro colspan=2 significa que duas colunas não serão ocupadas por nenhum componente
- d. Na linha 32 o parâmetro colspan=2 significa que o componente ocupará duas colunas. ✓
- e. Na linha 35 o parâmetro sticky=E significa que o botão ocupará o lado leste da segunda coluna. ✓



Sua resposta está correta.

As respostas corretas são: Os parâmetros apresentados no comando da linha 29 são:

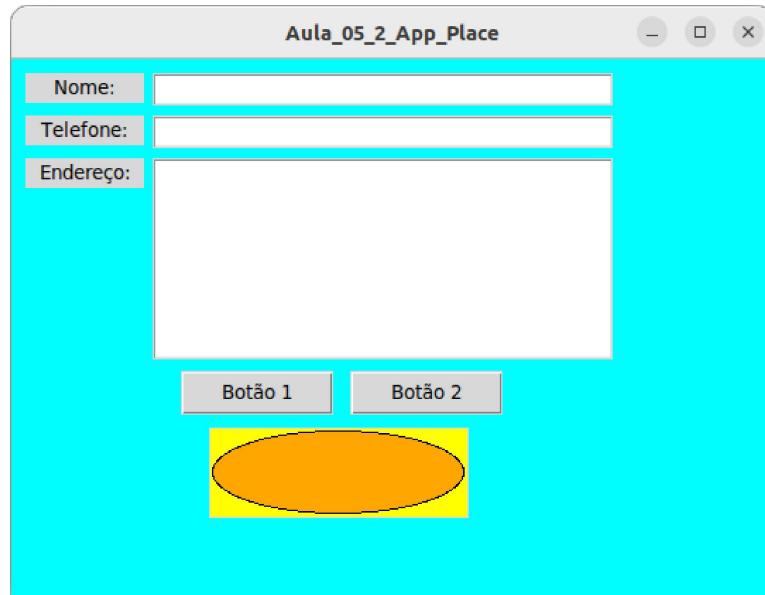
Linha, Coluna, separação entre componentes no eixo x, separação entre componentes no eixo y, Na linha 32 o parâmetro colspan=2 significa que o componente ocupará duas colunas., Na linha 35 o parâmetro sticky=E significa que o botão ocupará o lado leste da segunda coluna.



Questão 6

Correto

Atingiu 1,00 de 1,00

Dado o código abaixo:

Janela.py

```
01... from tkinter import *
02...
03...
04...
05... class Janela(Tk):
06...     def __init__(self, Str="Janela", px=0, py=0, dx=640, dy=480, cor="ligthgray"):
07...         super(Janela, self).__init__()
08...         super(Janela, self).title(Str)
09...         super(Janela, self).geometry("%dx%d+%d+%d" % (dx, dy, px, py))
10...         super(Janela, self).configure(bg=cor)
11...         self.inicialize()
12...
13...     def Posiciona_Componentes(self):
14...         self.Lb1.place(x=10, y=10)
15...         self.Lb2.place(x=10, y=40)
16...         self.Lb3.place(x=10, y=70)
17...         self.Et1.place(x=100, y=10)
18...         self.Et2.place(x=100, y=40)
19...         self.Txt1.place(x=100, y=70)
20...         self.Bt1.place(x=120, y=220)
21...         self.Bt2.place(x=240, y=220)
22...         self.Cnv.place(x=140, y=260)
23...
24...     def inicialize(self):
25...         self.Lb1 = Label(self, text="Nome:", height=1, width=10)
26...         self.Lb2 = Label(self, text="Telefone:", height=1, width=10)
27...         self.Lb3 = Label(self, text="Endereço:", height=1, width=10)
28...
29...         self.Et1 = Entry(self, width=40)
```



```

30...
31...
32...     self.Et2 = Entry(self, width=40)
33...
34...     self.Txt1 = Text(self, height=8, width=40)
35...     self.Bt1 = Button(self, text='Botão 1', height=1, width=10)
36...     self.Bt2 = Button(self, text='Botão 2', height=1, width=10)
37...     self.Cnv = Canvas(self, bd=0, bg="yellow", width=182, height=62)
38...     self.Cnv.create_oval(2, 2, 180, 60, fill="orange")
39...
40...     self.Posiciona_Componentes()
41...
42... #####

```

Aula_05_2_App_Place.py



```

43... from Janela import Janela
44...
45... #####
46...
47... Jan1=Janela("Minha janela", 400, 200, 540, 380, "cyan")
48... Jan1.mainloop()
49...
50... #####

```

A omissão da linha 40 através de um comentário causa o seguinte efeito:

Escolha uma ou mais:

- a. Ocorre um erro de execução na linha 47, NullPointerException, pois não houve posicionamento dos componentes na tela

- b. Quando omitimos o Layout de apresentação da tela, o Python assume o GridLayout como Layout padrão
- c. O Python roda o programa e omite a apresentação dos componentes ✓
- d. Ocorre um erro de execução na linha 48, pois o sistema tentará posicionar um componente na tela
- e. O Python roda o programa e apresenta uma tela vazia ✓

Sua resposta está correta.

As respostas corretas são: O Python roda o programa e omite a apresentação dos componentes, O Python roda o programa e apresenta uma tela vazia



Questão 7

Correto

Atingiu 2,00 de 2,00

Considereando o código apresentado anteriormente, responda qual o significado dos comandos:

- | | | |
|----|---|--|
| 01 | <code>__init__</code> | Construtor da classe |
| 02 | <code>place</code> | Comando para posicionar componentes |
| 03 | <code>super(Janela, self).__init__()</code> | Chama construtor da classe base |
| 04 | <code>geometry</code> | Comando para posicionar a janela na tela e atribuir uma largura e altura |
| 05 | <code>self</code> | Faz referência a variáveis globais ou métodos da própria classe |
| 06 | <code>Entry</code> | Componente visual para digitação de texto de uma única linha |
| 07 | <code>Text</code> | Componente visual para digitação de texto de várias linhas |



08 Button

Componente visual que representa um botão na interface



09 Canvas

componente visual para desenhar figuras



10 mainloop

Cria um loop infinito para que a janela fique na tela até que o usuário feche-a



Sua resposta está correta.

A resposta correta é:

01 __init__

→ Construtor da classe,

02 place

→ Comando para posicionar componentes,

03 super(Janela, self).__init__()

→ Chama construtor da classe base,

04 geometry

→ Comando para posicionar a janela na tela e atribuir uma largura e altura,

05 self

→ Faz referência a variáveis globais ou métodos da própria classe,

06 Entry



→ Componente visual para digitação de texto de uma única linha,

07 Text

→ Componente visual para digitação de texto de várias linhas,

08 Button

→ Componente visual que representa um botão na interface,

09 Canvas

→ componente visual para desenhar figuras,

10 mainloop

→ Cria um loop infinito para que a janela fique na tela até que o usuário feche-a.



Questão 8

Correto

Atingiu 1,00 de 1,00

Considerando o código apresentado anteriormente, responda quais as alternativas corretas:

Escolha uma ou mais:

- a. O comando **super(Janela, self).__init__()** pode ser substituído por **super(Tk).__init__()**
- b. O comando **super(Janela, self).__init__()** pode ser substituído por **super().__init__()** ✓
- c. O comando **super(Janela, self).__init__()** refere-se ao Python 2 ✓
- d. O comando **super().__init__()** refere-se ao Python 3 ✓
- e. O comando **super(Janela, self).title(Str)** atribui um título à janela ✓



Sua resposta está correta.

As respostas corretas são: O comando **super(Janela, self).__init__()** pode ser substituído por **super().__init__()**, O comando **super(Janela, self).__init__()** refere-se ao Python 2, O comando **super().__init__()** refere-se ao Python 3, O comando **super(Janela, self).title(Str)** atribui um título à janela

Questão 9

Correto

Atingiu 1,00 de 1,00

Analise o código abaixo:

Aula_06_2a_App_1Cam_Aluno

Nome=	João
Sexo=	Masc.
Nota1=	7.5
Nota2=	8.6
Nota3=	9.4
Média=	8.50
<input type="button" value="Calcular"/>	



Janela.py

```
01... from tkinter import *
02...
03... #####
04...
05... class Janela(Tk):
06...     __Et_Notal=None
07...     __Et_Nota2=None
08...     __Et_Nota3=None
09...     __Et_Media=None
10...
11... #####
12... ##
13... ##      Complete o código      ##
14... ##
15... ##
16... #####
17...
18... def action_Bt_Calc(self):
19...     #####
20...     ##
21...     ##      Complete o código do evento      ##
22...     ##
23...     ##
24...     #####
25...
26... def inicialize(self):
27...     #####
28...     ##
29...     ##      Complete o código de alocação      ##
```



```
30...          ##          ##
31...          ##          ##
32...          ######
33...
34...      Lb_Nome.configure(bg='yellow')
35...      Lb_Sexo.configure(bg='yellow')
36...      Lb_Nota1.configure(bg='yellow')
37...      Lb_Nota2.configure(bg='yellow')
38...      Lb_Nota3.configure(bg='yellow')
39...      Lb_Media.configure(bg='yellow')
40...
41...      Et_Nome=Entry(self, width=52)
42...      self.__Et_Nota1=Entry(self, width=52)
43...      self.__Et_Nota2=Entry(self, width=52)
44...      self.__Et_Nota3=Entry(self, width=52)
45...      self.__Et_Media=Entry(self, width=52)
46...
47...      var=StringVar(self)
48...      var.set('Escolha a opção')
49...      OM_Sexo=OptionMenu(self, var, 'Escolha a opção', 'Masc.', 'Fem.')
50...      OM_Sexo.config(width=32)
51...
52...      Bt_Calc=Button(self, text='Calcular', command=self.action_Bt_Calc)
53...
54...          #####
55...          ##          ##
56...          ## Complete o código para acrescentar ##
57...          ## os componentes faltantes na tela  ##
58...          ##          ##
```



```

59...
60...    ######
61...
62...    Bt_Calc.grid(row=6, column=0, sticky=NW, padx=4, pady=4)
63...
64...    Et_Nome.grid(row=0, column=1, sticky=NW, padx=4, pady=4)
65...    OM_Sexo.grid(row=1, column=1, sticky=NW, padx=4, pady=4)
66...    self.__Et_Nota1.grid(row=2, column=1, sticky=NW, padx=4, pady=4)
67...    self.__Et_Nota2.grid(row=3, column=1, sticky=NW, padx=4, pady=4)
68...    self.__Et_Nota3.grid(row=4, column=1, sticky=NW, padx=4, pady=4)
69...    self.__Et_Media.grid(row=5, column=1, sticky=NW, padx=4, pady=4)
70...    #####

```

Aula_06_2a_App_1Cam_Aluno.py

```

70... from Janela import Janela
71...
72... #####
73...
74... Jan1=Janela("Minha janela", 400, 200, 540, 380, "orange")
75... Jan1.mainloop()
76...
77... #####

```



**As linhas de código suprimidas nas linhas 11 à 16 devem conter o construtor
da classe Janela.**

Qual a alternativa correta para a sua implementação?

Escolha uma ou mais:

a.

```
11... def Construtor(self, Str="Janela", px=0, :
                     cor="ligthgray"):
12...     super().__init__()
13...     super().title(Str)
14...     super(Janela, self).geometry("%dx%d+%
15...     super(Janela, self).configure(bg=cor)
16...     self.inicialize()
```

b.

```
11... definir init__(self, Str="Janela", px=0, :
                     cor="ligthgray"):
12...     super().__init__()
13...     super().title(Str)
14...     super(Janela, self).geometry("%dx%d+%
15...     super(Janela, self).configure(bg=cor)
16...     self.inicialize()
```



c.

```
11... def __init__(self, Str="Janela", px=0, py=
                     cor="ligthgray"):
12...     super().__init__()
13...     super().title(Str)
14...     super(Janela, self).geometria("%dx%d+%
15...     super(Janela, self).configuração(bg=c
16...     self.inicialize()
```

d.

```
11...     def __init__(self, Str="Janela", px=0, py
                  cor="ligthgray"):
12...         super().__init__()
13...         super().title(Str)
14...         super(Janela, self).geometry("%dx%d+%
15...             super(Janela, self).configure(bg=cor)
16...             self.inicialize()
```

 e.

```
11...     def __Janela__(self, Str="Janela", px=0, :
                  cor="ligthgray"):
12...         super(this).__init__()
13...         super(this).title(Str)
14...         super(Janela, this).geometry("%dx%d+%
15...             super(Janela, self).configure(bg=cor)
16...             self.inicialize()
```



Sua resposta está correta.

A resposta correta é:

```
11...     def __init__(self, Str="Janela", px=0, py=0, dx=640, dy=480,
                  cor="ligthgray"):
12...         super().__init__()
13...         super().title(Str)
14...         super(Janela, self).geometry("%dx%d+%d+%d" % (dx, dy, px, py))
15...         super(Janela, self).configure(bg=cor)
16...         self.inicialize()
```

Questão 10

Correto

Atingiu 1,00 de 1,00

As linhas de código de 27 à 32 necessárias para alocar todos labels amarelos são:

Escolha uma ou mais:

**a.**

```
27...     Lb_Nome=Label(self, text="Nome=")
28...     Lb_Sexo=Setbackground(self, bg="yellow"
29...     Lb_Notal=Label(self, text="Nota1=")
30...     Lb_Nota2=Label(self, text="Nota2=")
31...     Lb_Nota3=Label(self, text="Nota3=")
32...     Lb_Media=Label(self, text="Media=")
```

**b.**

```
27...     Lb_Nome=Label(self, text="Nome=")
28...     Lb_Sexo=Label(self, text="Sexo=")
29...     Lb_Notal=Label(self, text="Nota1=")
30...     Lb_Nota2=Label(self, text="Nota2=")
31...     Lb_Nota3=Label(self, text="Nota3=")
32...     Lb_Media=Label(self, text="Media=")
```

c. 27... Lb_Nome=Label(self, text="Nome=")
28... Lb_Sexo=Label(self, text="Sexo=")
29... Lb_Notal=Label(self, text="Notal=")
30... Lb_Notal=Label(self, text="Nota2=")
31... Lb_Notal=Label(text="Nota3=")
32... Lb_Media=Label(self, text="Media=")

d. 27... Lb_Nome=Label(self, text="Nome=")
28... Lb_Nome=Label(self, text="Sexo=")
29... Lb_Nota2=Label(text="Notal=")
30... Lb_Nota2=Label(text="Nota2=")
31... Lb_Nota2=Label(text="Nota3=")
32... Lb_Media=Label(self, text="Media=")

e. 27... Lb_Nome=Entry(self, text="Nome=")
28... Lb_Sexo=Entry(self, text="Sexo=")
29... Lb_Notal=Entry(self, text="Notal=")
30... Lb_Nota2=Label(self, text="Nota2=")
31... Lb_Nota3=Label(self, text="Nota3=")
32... Lb_Media=Label(self, text="Media=")



Sua resposta está correta.

A resposta correta é:

```
27...     Lb_Nome=Label(self, text="Nome=")
28...     Lb_Sexo=Label(self, text="Sexo=")
29...     Lb_Nota1=Label(self, text="Nota1=")
30...     Lb_Nota2=Label(self, text="Nota2=")
31...     Lb_Nota3=Label(self, text="Nota3=")
32...     Lb_Media=Label(self, text="Media=")
```



Questão 11

Correto

Atingiu 1,00 de 1,00

Nas linhas 54 à 59 faltam os comandos necessários para acrescentar alguns componentes na tela.

A alternativa correta é:

Escolha uma opção:

a.

54...	Lb_Nome.grid(row=0, column=0, sticky=N)
55...	Lb_Sexo.grid(row=1, column=0, sticky=N)
56...	Lb_Notal.grid(row=1, column=0, sticky=E)
57...	Lb_Nota2.grid(row=1, column=0, sticky=E)
58...	Lb_Nota3.grid(row=1, column=0, sticky=E)
59...	Lb_Media.grid(row=1, column=0, sticky=E)

**b.**

54...	grid(Lb_Nome, row=0, column=0, sticky=E)
55...	grid(Lb_Sexo, row=1, column=0, sticky=E)
56...	grid(Lb_Notal, row=2, column=0, sticky=E)
57...	grid(Lb_Nota2, row=3, column=0, sticky=E)
58...	grid(Lb_Nota3, row=4, column=0, sticky=E)
59...	grid(Lb_Media, row=5, column=0, sticky=E)

c.

54....	Et_Nome.grid(row=0, column=0, sticky=N)
55....	Et_Sexo.grid(row=1, column=0, sticky=N)
56....	Et_Notal.grid(row=2, column=0, sticky=E)
57....	Et_Nota2.grid(row=3, column=0, sticky=E)
58....	Et_Nota3.grid(row=4, column=0, sticky=E)
59....	Et_Media.grid(row=5, column=0, sticky=E)

d.

54....	Lb_Nome.gridLayout(row=0, column=0, or
55....	Lb_Sexo.gridLayout(row=1, column=0, or
56....	Lb_Notal.gridLayout(row=2, column=0, or
57....	Lb_Nota2.gridLayout(row=3, column=0, or
58....	Lb_Nota3.gridLayout(row=4, column=0, or
59....	Lb_Media.gridLayout(row=5, column=0, or

e.

54....	Lb_Nome.grid(row=0, column=0, sticky=N)
55....	Lb_Sexo.grid(row=1, column=0, sticky=N)
56....	Lb_Notal.grid(row=2, column=0, sticky=E)
57....	Lb_Nota2.grid(row=3, column=0, sticky=E)
58....	Lb_Nota3.grid(row=4, column=0, sticky=E)
59....	Lb_Media.grid(row=5, column=0, sticky=E)



Sua resposta está correta.

A resposta correta é:

54....	Lb_Nome.grid(row=0, column=0, sticky=NW, padx=4, pady=4)
55....	Lb_Sexo.grid(row=1, column=0, sticky=NW, padx=4, pady=4)
56....	Lb_Nota1.grid(row=2, column=0, sticky=NW, padx=4, pady=4)
57....	Lb_Nota2.grid(row=3, column=0, sticky=NW, padx=4, pady=4)
58....	Lb_Nota3.grid(row=4, column=0, sticky=NW, padx=4, pady=4)
59....	Lb_Media.grid(row=5, column=0, sticky=NW, padx=4, pady=4)



Questão 12

Correto

Atingiu 1,00 de 1,00

No programa apresentado anteriormente, complete o código que processa o evento associado ao botão Bt_Calc (Linhas 19 a 24):

Escolha uma ou mais opções:

Escolha uma ou mais:

a.

```
19...     n1=self.__Et_Nota1.get()
20...     n2=self.__Et_Nota2.get()
21...     n3=self.__Et_Nota3.get()
22...     total=(n1+n2+n3)/3
23...     self.__Et_Media.delete(0, END)
24...     self.__Et_Media.insert(END, "%f" % tota
```



b.

```
19...     n1=float(self.__Et_Nota1.get())
20...     n2=float(self.__Et_Nota2.get())
21...     n3=float(self.__Et_Nota3.get())
22...     total=(n1+n2+n3)/3
23...     self.__Et_Media.delete(0, END)
24...     self.__Et_Media.insert(END, "%f" % tota
```

c. 19... n1=float(self.__Et_Notal.get())
20... n1=float(self.__Et_Notal.get())
21... n1=float(self.__Et_Notal.get())
22... total=(n1+n2+n3)/3
23... self.__Et_Media.delete(0, END)
24... self.__Et_Media.insert(END, "%f" % tota

d. 19... n1=float(Et_Notal.get())
20... n2=float(Et_Nota2.get())
21... n3=float(Et_Nota3.get())
22... total=(n1+n2+n3)/3
23... self.__Et_Media.delete(0, END)
24... self.__Et_Media.insert(END, "%f" % tota

e. 19... n1=float(self.__Et_Notal.getText())
20... n2=float(self.__Et_Nota2.getText())
21... n3=float(self.__Et_Nota3.getText())
22... total=(n1+n2+n3)/3
23... Et_Media.delete(0, END)
24... Et_Media.insert(END, "%f" % total)



Sua resposta está correta.

A resposta correta é:

```
19...     n1=float(self.__Et_Nota1.get())
20...     n2=float(self.__Et_Nota2.get())
21...     n3=float(self.__Et_Nota3.get())
22...     total=(n1+n2+n3)/3
23...     self.__Et_Media.delete(0, END)
24...     self.__Et_Media.insert(END, "%f" % total)
```



Questão 13

Correto

Atingiu 1,00 de 1,00

A sequência correta para ler e apresentar uma imagem em um Aplicativo Python é dado por:

Escolha uma ou mais:

- a.
- **Passo 1:** Importar a biblioteca PIL como comando:
from PIL import ImageTk, Image
 - **Passo 2:** Ler a imagem com o método getImagem("Nome do arquivo") com o nome do arquivo
 - **Passo 3:** Transformar a imagem em uma foto com o comando:
photo = Transforma(Imagen original)
 - **Passo 4:** Colar a foto sobre um Label com o comando:
Label(self, image=photo)



- b.
- **Passo 1:** Importar a biblioteca PIL como comando:
from PIL import ImageTk, Image
 - **Passo 2:** Ler a imagem com o método Imagem.open("Nome do arquivo") com o nome do arquivo
 - **Passo 3:** Transformar a imagem em uma foto com o comando:
photo = ImageTk.PhotoImage(Imagen original)
 - **Passo 4:** Colar a foto sobre um Label com o comando de alocação
Table(self, imagem=photo)

c.

- **Passo 1:** Importar a biblioteca PIL como comando:
from PIL import ImageTk, Image
- **Passo 2:** Ler a imagem com o método Text.open("Nome do arquivo") com o nome do arquivo
- **Passo 3:** Transformar a imagem em uma foto com o comando:
photo = TextTk.PhotoImage(Imagen original)
- **Passo 4:** Colar a foto sobre um Label com o comando atribuição:
Text(self, image=photo)

 d.

- **Passo 1:** Importar a biblioteca Photo como comando:
from PIL import ImagemTk, Image
- **Passo 2:** Ler a imagem com o método ImagemTKinter.open("Nome do arquivo") com o nome do arquivo
- **Passo 3:** Transformar a imagem em uma foto com o comando:
photo = ImageTkinter.PhotoImage(Imagen original)
- **Passo 4:** Colar a foto sobre um Label com o comando:
Button(self, image=photo)



e.

- **Passo 1:** Importar a biblioteca PIL como comando:
from PIL import ImageTk, Image
- **Passo 2:** Ler a imagem com o método Image.open("Nome do arquivo") com o nome do arquivo
- **Passo 3:** Transformar a imagem em uma foto com o comando:
photo = ImageTk.PhotoImage(Imagen original)
- **Passo 4:** Colar a foto sobre um Label com o comando de alocação:
Label(self, image=photo)

Sua resposta está correta.

A resposta correta é:

- **Passo 1:** Importar a biblioteca PIL como comando:
from PIL import ImageTk, Image
- **Passo 2:** Ler a imagem com o método Image.open("Nome do arquivo") com o nome do arquivo
- **Passo 3:** Transformar a imagem em uma foto com o comando:
photo = ImageTk.PhotoImage(Imagen original)
- **Passo 4:** Colar a foto sobre um Label com o comando de alocação:
Label(self, image=photo)



Questão 14

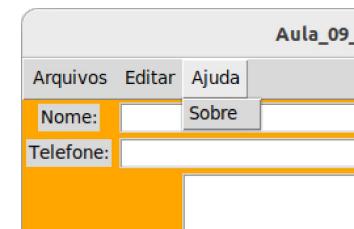
Correto

Atingiu 1,00 de 1,00

Dado o código de alocação:

```
avo_bar = Menu(self)
```

Qual a sequência correta para adicionar todos os itens de menu na janela a seguir?



Escolha uma ou mais:

a.

```
pai_arquivos = MenuLinear(tearoff=0)
pai_arquivos.add_command(Text="Abrir", command=self.abrir)
pai_arquivos.add_command(Text="Salvar", command=self.salvar)
pai_arquivos.add_separator()
pai_arquivos.add_command(Text="Sair", command=self.quit)
avo_bar.add_cascade(Text="Arquivos", menu=pai_arquivos)

pai_editar = MenuLinear(avo_bar, tearoff=0)
pai_editar.add_command(Text="Cortar")
pai_editar.add_command(Text="Copiar")
pai_editar.add_command(Text="Colar")
avo_bar.add_cascade(Text="Editar", menu=pai_editar)

pai_ajuda = MenuLinear(avo_bar, tearoff=0)
pai_ajuda.add_command(Text="Sobre", command=self.ajuda)
avo_bar.add_cascade(Text="Ajuda", menu=pai_ajuda)
```



b.

```
pai_arquivos = Menu(avo_bar, tearoff=0)
pai_arquivos.add_command(label="Abrir", command=self.
pai_arquivos.add_command(label="Salvar", command=self.
pai_arquivos.add_separator()
pai_arquivos.add_command(label="Sair", command=self.e
avo_bar.add_cascade(label="Arquivos", menu=pai_arquiv

pai_editar = Menu(avo_bar, tearoff=0)
pai_editar.add_command(label="Cortar")
pai_editar.add_command(label="Copiar")
pai_editar.add_command(label="Colar")
avo_bar.add_cascade(label="Editar", menu=pai_editar)

pai_ajuda = Menu(avo_bar, tearoff=0)
pai_ajuda.add_command(label="Sobre", command=self.act
avo_bar.add_cascade(label="Ajuda", menu=pai_ajuda)
```



c.

```
avo_arquivos = Menu(avo_bar, tearoff=0)
avo_arquivos.add_command(label="Abrir", command=self.
avo_arquivos.add_command(label="Salvar", command=self.
avo_arquivos.add_separator()
avo_arquivos.add_command(label="Sair", command=self.e
pai_bar.add_cascade(label="Arquivos", menu=pai_arquiv

avo_editar = Menu(avo_bar, tearoff=0)
avo_editar.add_command(label="Cortar")
avo_editar.add_command(label="Copiar")
avo_editar.add_command(label="Colar")
pai_bar.add_cascade(label="Editar", menu=pai_editar)

avo_ajuda = Menu(avo_bar, tearoff=0)
avo_ajuda.add_command(label="Sobre", command=self.act
pai_bar.add_cascade(label="Ajuda", menu=pai_ajuda)
```



d.

```
pai_arquivos = Menu(avo_bar, tearoff=0)
pai_arquivos.add_command(command=self.action_window,
pai_arquivos.add_command(command=self.action_salvar,
pai_arquivos.add_separator()
pai_arquivos.add_command(command=self.action_exit, l@)
avo_bar.add_cascade(menu=pai_arquivos, label="Arquivc

pai_editar = Menu(avo_bar, tearoff=0)
pai_editar.add_command(label="Cortar")
pai_editar.add_command(label="Copiar")
pai_editar.add_command(label="Colar")
avo_bar.add_cascade(label="Editar", menu=pai_editar)

pai_ajuda = Menu(avo_bar, tearoff=0)
pai_ajuda.add_command(label="Sobre", command=self.act
avo_bar.add_cascade(label="Ajuda", menu=pai_ajuda)
```



e.

```
pai_arquivos = Menu(avo_bar, tearoff=0)
pai_arquivos.adicionar_command(label="Abrir", command=... )
pai_arquivos.adicionar_command(label="Salvar", command=... )
pai_arquivos.adicionar_separator()
pai_arquivos.adicionar_command(label="Sair", command=... )
avo_bar.adicionar_cascade(label="Arquivos", menu=pai_arquivos)

pai_editar = Menu(avo_bar, tearoff=0)
pai_editar.adicionar_command(label="Cortar")
pai_editar.adicionar_command(label="Copiar")
pai_editar.adicionar_command(label="Colar")
avo_bar.adicionar_cascade(label="Editar", menu=pai_editar)

pai_ajuda = Menu(avo_bar, tearoff=0)
pai_ajuda.adicionar_command(label="Sobre", command=... )
avo_bar.adicionar_cascade(label="Ajuda", menu=pai_ajuda)
```



Sua resposta está correta.

As respostas corretas são:

```
pai_arquivos = Menu(avo_bar, tearoff=0)
pai_arquivos.add_command(label="Abrir", command=self.action_window)
pai_arquivos.add_command(label="Salvar", command=self.action_salvar)
pai_arquivos.add_separator()
pai_arquivos.add_command(label="Sair", command=self.action_exit)
avo_bar.add_cascade(label="Arquivos", menu=pai_arquivos)

pai_editar = Menu(avo_bar, tearoff=0)
pai_editar.add_command(label="Cortar")
pai_editar.add_command(label="Copiar")
pai_editar.add_command(label="Colar")
avo_bar.add_cascade(label="Editar", menu=pai_editar)

pai_ajuda = Menu(avo_bar, tearoff=0)
pai_ajuda.add_command(label="Sobre", command=self.action_sobre)
avo_bar.add_cascade(label="Ajuda", menu=pai_ajuda)
```

,



```
pai_arquivos = Menu(avo_bar, tearoff=0)
pai_arquivos.add_command(command=self.action_window, label="Abrir")
pai_arquivos.add_command(command=self.action_salvar, label="Salvar")
pai_arquivos.add_separator()
pai_arquivos.add_command(command=self.action_exit, label="Sair")
avo_bar.add_cascade(menu=pai_arquivos, label="Arquivos")

pai_editar = Menu(avo_bar, tearoff=0)
pai_editar.add_command(label="Cortar")
pai_editar.add_command(label="Copiar")
pai_editar.add_command(label="Colar")
avo_bar.add_cascade(label="Editar", menu=pai_editar)

pai_ajuda = Menu(avo_bar, tearoff=0)
pai_ajuda.add_command(label="Sobre", command=self.action_sobre)
avo_bar.add_cascade(label="Ajuda", menu=pai_ajuda)
```

