

Estatísticas e uma questão aberta do questionário final de BCC-QS-2020

Segue célula de código para ler a planilha contendo as respostas do questionário, além de criar listas para gráficos:

- BoxPlot e Violin:
 - **cols** - questões com valores de 1 até 5
 - **cols10** - questões com valores de 1 até 10
- Pizza:
 - **colsCateg** - questões com categorias

```

1  from google.colab import files
2  import matplotlib.pyplot as plt
3  import plotly.graph_objects as go
4  import pylab
5  import seaborn as sns
6  import textwrap
7  import pandas as pd
8  import numpy as np
9  %matplotlib inline
10
11  SERVER_PATH = 'http://vision.ufabc.edu.br/MCTest/public/WEI2021/'
12
13  # to save and download figures of paper
14  global SAVE_FIGS_PAPER
15  SAVE_FIGS_PAPER = False
16  if SAVE_FIGS_PAPER:
17      DPI_resolution = 200
18      width_resolution = 2400
19      height_resolution = 1600
20      !pip install plotly>=4.0.0
21      !wget https://github.com/plotly/orca/releases/download/v1.2.1/orca-1.2.1-x86
22      !chmod +x /usr/local/bin/orca
23      !apt-get install xvfb libgtk2.0-0 libgconf-2-4

1  # for reak on Google Drive
2  #!pip install --upgrade gspread
3  #from google.colab import auth
4  #auth.authenticate_user()
5  #from oauth2client.client import GoogleCredentials
6  #import gspread
7  #gc = gspread.authorize(GoogleCredentials.get_application_default())
8  #worksheet = gc.open('BCC-Exercícios Gerais-2020-QS (respostas)').sheet1
9  #df = pd.DataFrame.from_records(worksheet.get_all_values())
10

```

```

11 df = pd.read_csv(SERVER_PATH+'answers_omitted.csv')
12 #df.columns = df.iloc[0]
13 df.drop(df.index[:1], inplace=True)
14 df = df.reindex(sorted(df.columns), axis=1)
15
16 rowsSTR = df.columns
17 df.columns = [i for i in range(len(df.columns))]
18 cols, cols10, colsCateg = [], [], []
19 for c in df.columns:
20     try:
21         df[c] = df[c].astype(int)
22         if df[c].max()<=5:
23             cols.append(c)
24         else:
25             cols10.append(c)
26     except:
27         colsCateg.append(c)
28 df.shape

```

```
(463, 26)
```

```
1 print(cols)
```

```
[4, 9, 10, 12, 14, 15, 17, 18, 22, 23]
```

```
1 print(cols10)
```

```
[16, 19, 20, 21]
```

```
1 print(colsCateg)
```

```
[0, 1, 2, 3, 5, 6, 7, 8, 11, 13, 24, 25]
```

▼ BoxPlot

- box branco = média
- 1Q - base do retângulo = primeiro quartil = abaixo estão os 25% menores
- 2Q - segundo quartil = mediana
- 3Q - topo do retângulo = terceiro quartil = acima estão os 25% maiores
- limite inferior = $1Q - 1.5 \cdot (3Q - 1Q)$
- limite superior = $3Q + 1.5 \cdot (3Q - 1Q)$
- bolas - valores discrepantes (outliers) dos dados abaixo/acima dos limites

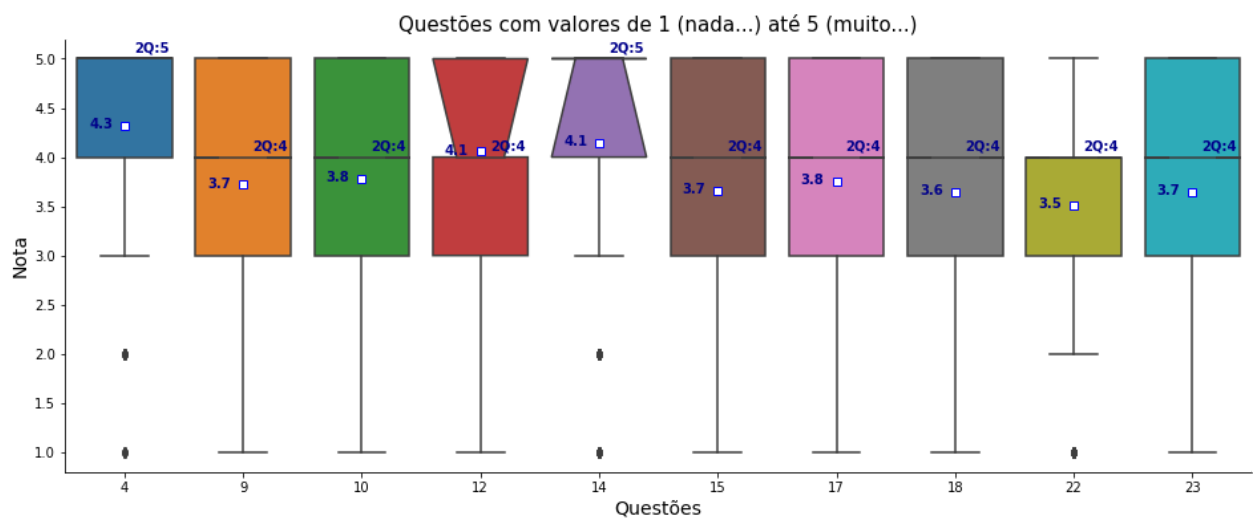
▼ Questões com notas de 0 (nada ...) até 5 (muito ...)

```
1 def draw_BoxPlot(df, title, cols):
```

```

2  pylab.rcParams['figure.figsize'] = (int(1.6*len(cols)),6)
3
4  ax = sns.boxplot(data=df[cols], showmeans=True, notch=True, bootstrap=10000,
5                  meanprops={"marker":"s","markerfacecolor":"white", "markere
6
7  means = df[cols].mean().values
8  medians = df[cols].median().values
9  ax.spines["top"].set_visible(False)
10 ax.spines["right"].set_visible(False)
11
12 plt.title(title, size='15')
13
14 pos = range(len(cols))
15 for tick,label in zip(pos,ax.get_xticklabels()):
16     ax.text(pos[tick] + 0.23, medians[tick] + 0.07, '2Q:'+str(int(medians[tick]
17     ax.text(pos[tick] - 0.2, means[tick] - 0.03, round(means[tick],1), horizon
18
19 ax.set_xlabel("Questões", fontsize=14)
20 ax.set_ylabel("Nota", fontsize=14)
21 #ax = df.boxplot(cols, return_type='axes', figsize=(10,6), showmeans=True)
22 ax.plot()
23
24 if SAVE_FIGS_PAPER:
25     file = 'fig_cols_' + '-'.join([str(i) for i in cols]) + '.tif'
26     plt.savefig(file, dpi=DPI_resolution)
27     files.download(file)
28
29 draw_BoxPlot(df,'Questões com valores de 1 (nada...) até 5 (muito...)',cols)

```



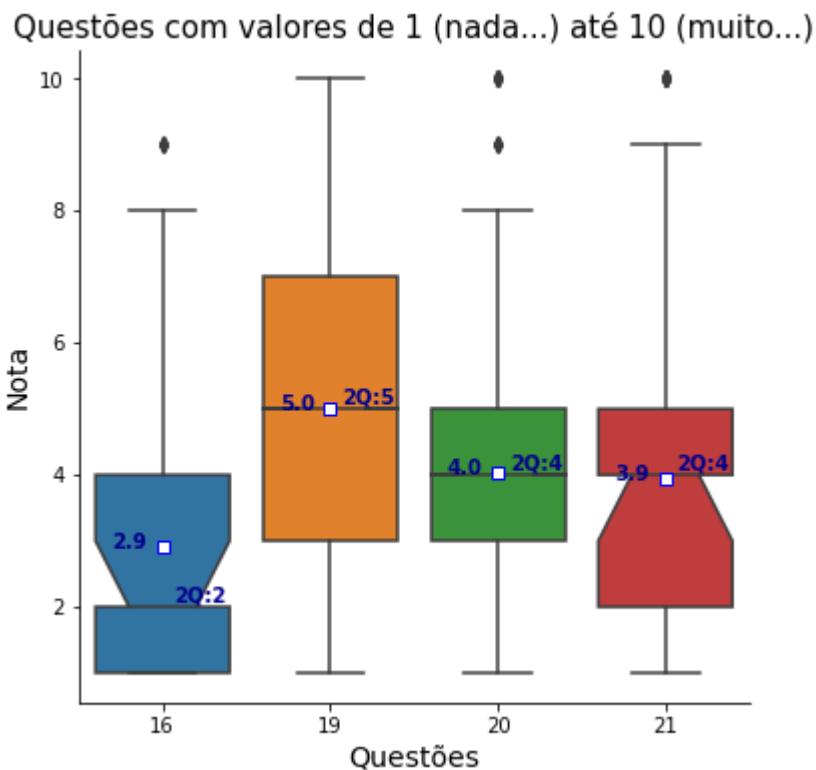
```
1 print('\n'.join([str(i) + ' ' + str(rowsSTR[i]) for i in cols]))
```

4 2.1. Eu gostaria de saber o quanto você é favorável ou não a usar linguagem
9 2.6. Na sua opinião, a sequência de tópicos foi adequada para cobrir a ement

10 2.7. Percebeu ajuda no ensino/aprendizagem com plantão de MONITORIA (por ex
 12 3.1. É favorável em usar EXERCÍCIOS unificados em TODAS as turmas, NIVELANI
 14 3.2. Na sua opinião, as CORREÇÕES AUTOMÁTICAS dos exercícios no Moodle, apr
 15 3.3. Na sua opinião, os FEEDBACKS (mensagens de erros em inglês que o pythc
 17 3.5. Na sua opinião, as LISTAS INDIVIDUAIS COM EXERCÍCIOS DISTINTOS, com pe
 18 3.6. Na sua opinião, se aumentar o número de questões a serem sorteadas din
 22 4.1. Qual é a sua avaliação GERAL para essa oferta do BCC em 2020-QS, quant
 23 4.2. Qual é a sua avaliação GERAL para essa oferta do BCC em 2020-QS, quant

▼ Questões com notas de 0 (nada ...) até 10 (muito ...)

```
1 draw_BoxPlot(df,'Questões com valores de 1 (nada...) até 10 (muito...)',cols10
```



```
1 print('\n'.join([str(i) + ' ' + str(rowsSTR[i]) for i in cols10]))
```

16 3.4. DO SEU CONHECIMENTO, qual foi a porcentagem de PLÁGIO existente nas er
 19 3.7. Na sua opinião, qual seria o melhor PESO atribuído às LISTAS no concei
 20 3.8. Na sua opinião, qual seria o melhor PESO atribuído à(s) PROVA(S) - se
 21 3.9. Na sua opinião, qual seria o melhor PESO atribuído ao PROJETO - se adc

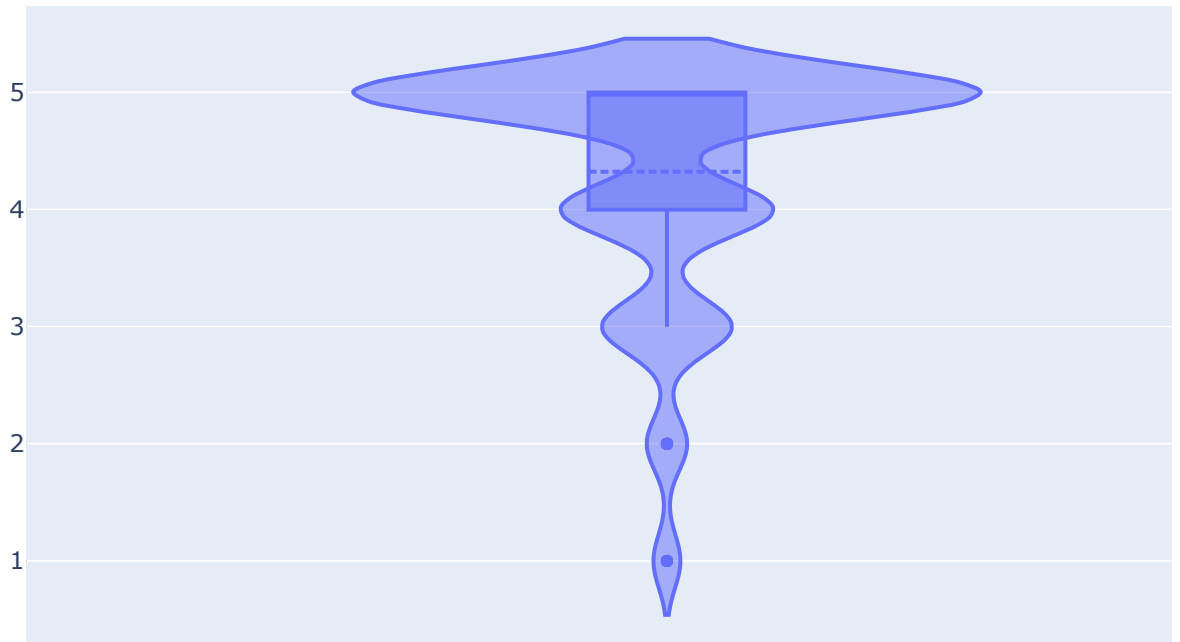
▼ Violin Plot

```
1 def drawViolinPlot(df,rows,col):
2     s = '<br>'.join(textwrap.wrap(rows[col], 110))
3     fig1 = go.Violin(meanline_visible=True, box_visible = True,
4         y=df[col].values, name='')
5     layout = go.Layout(title_font_size=12,
6         width=800,height=500,showlegend=False,
7         title = f'titulo = {s} | col = {col} | rows = {rows} | content = {content} | text = {text}')
```

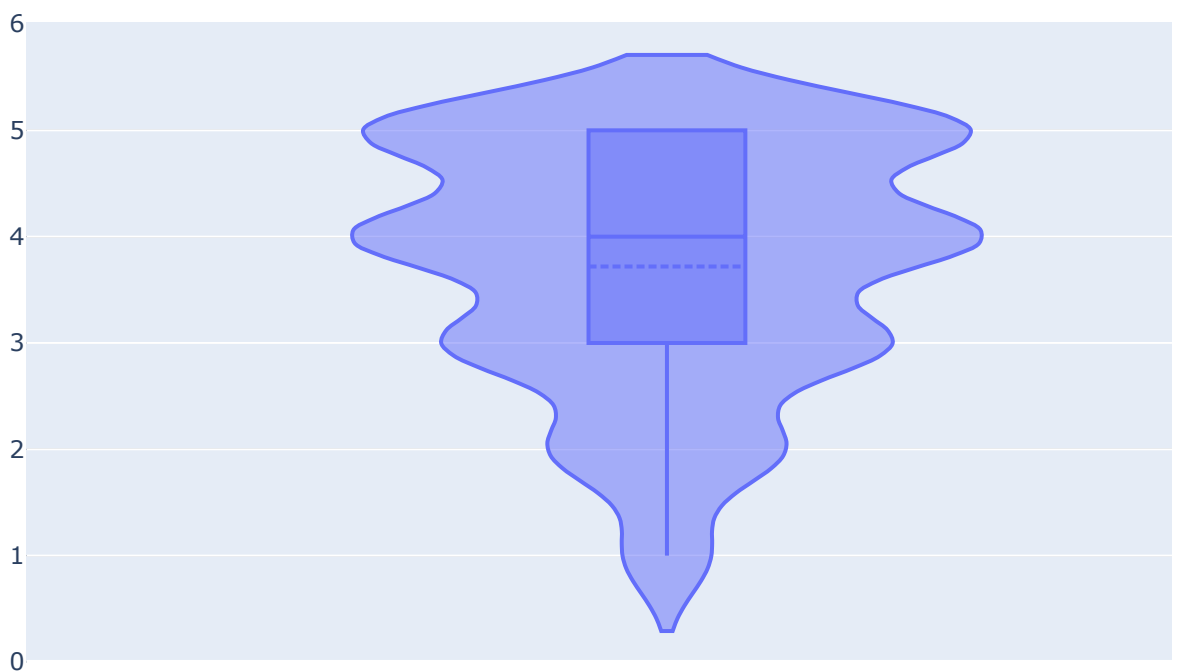
```
 /      title={text : s, y :0.9, x :0.5, xanchor : 'center', yanchor : 'top'},
 8      #yaxis=dict(title=cols[col])
 9      )
10  fig = go.Figure(layout=layout)
11  fig.add_trace(fig1)
12  fig.show()
13
14  if SAVE_FIGS_PAPER:
15      from PIL import Image
16      file = 'fig_col_'+ str(col) + '.png'
17      fig.write_image(file, width=width_resolution, height=height_resolution)
18      img = Image.open(file)
19      img.save(file)
20      files.download(file)
21
22  for i in cols+cols10:
23      drawViolinPlot(df,rowsSTR,i)
```



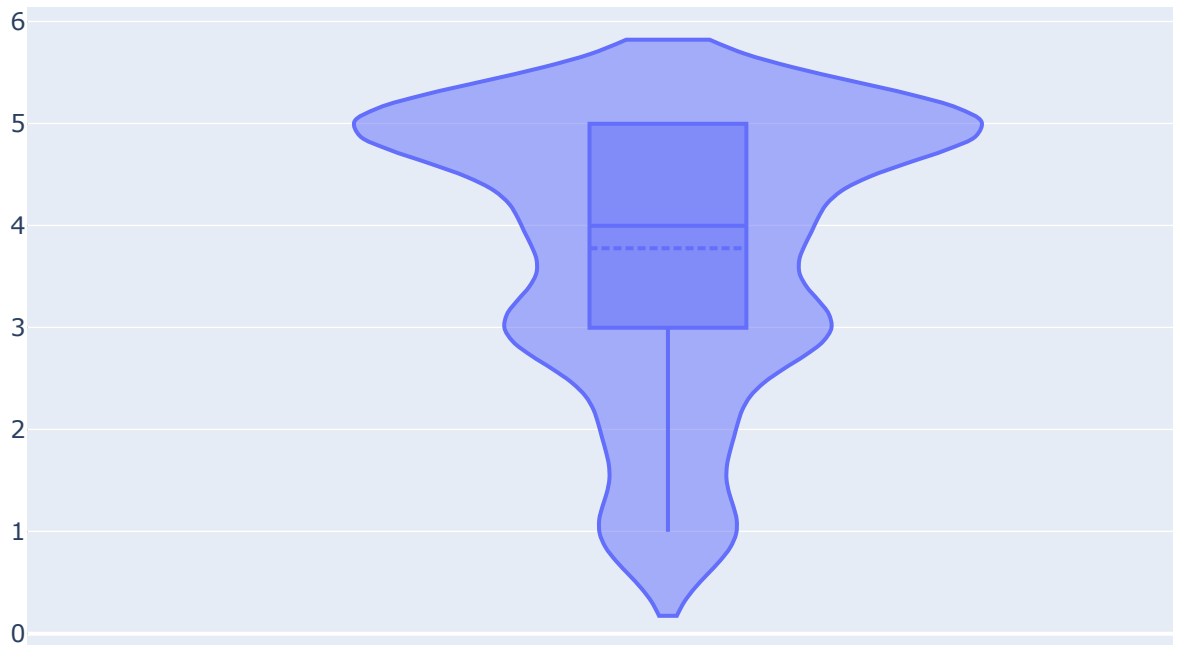
2.1. Eu gostaria de saber o quanto você é favorável ou não a usar linguagem unificada em TODAS exemplo, python)?



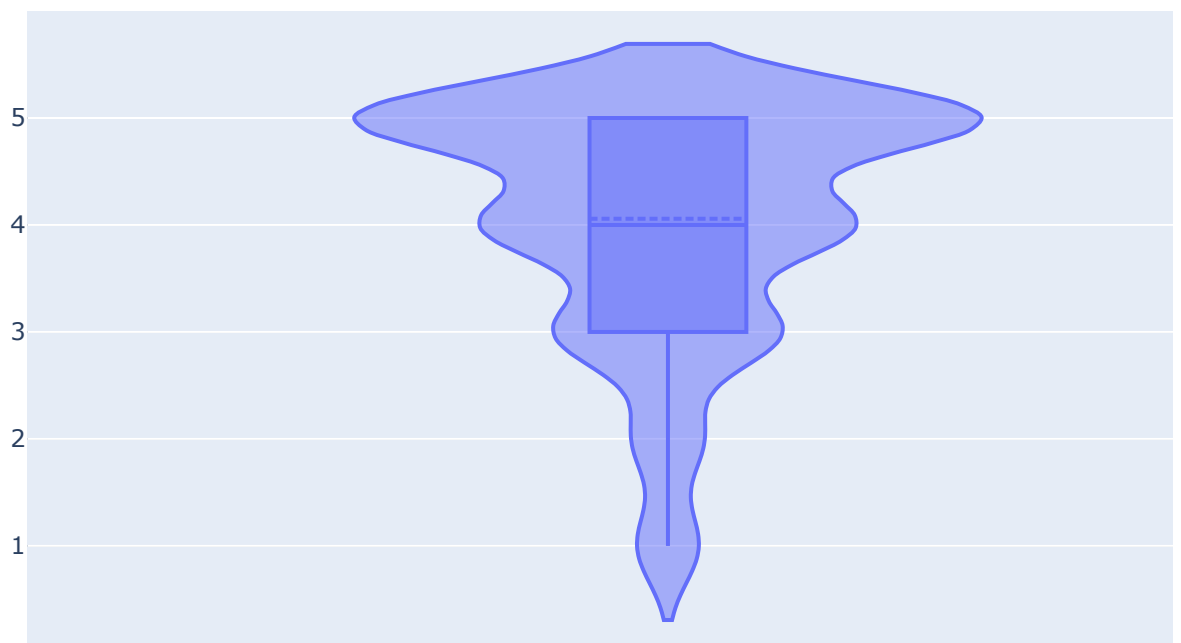
2.6. Na sua opinião, a sequência de tópicos foi adequada para cobrir a ementa de BCC? Ementa: ir de dados, gráficos, estatística descritiva, correlação-regressão, lógica de programação, e mo simulação.



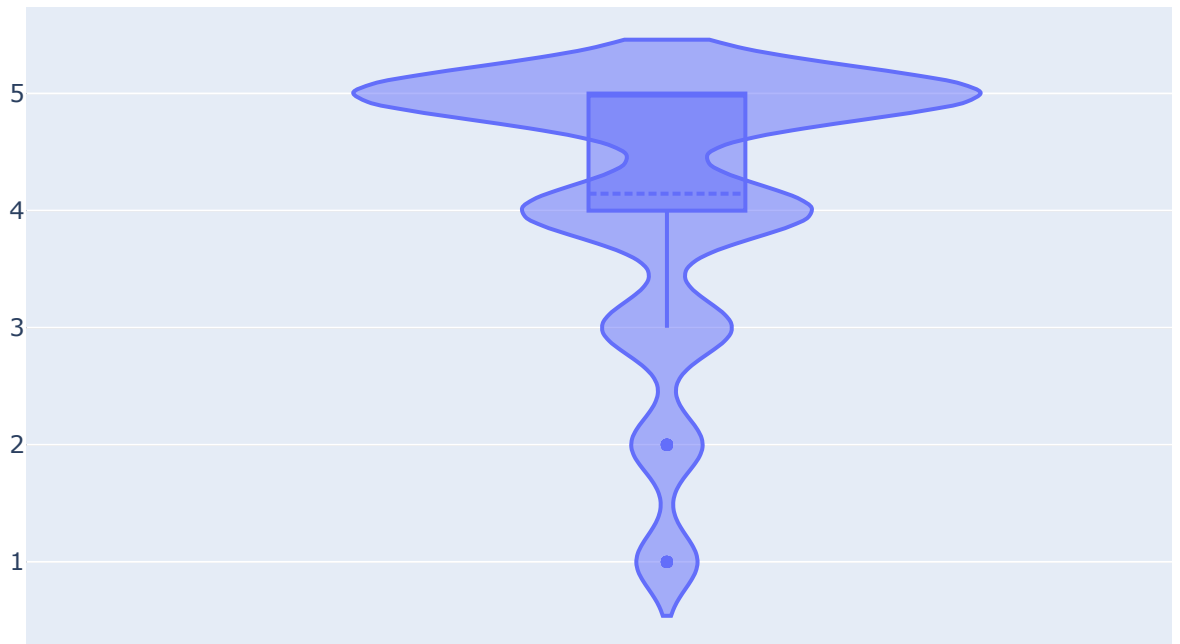
2.7. Percebeu ajuda no ensino/aprendizagem com plantão de MONITORIA (por exemplo, teve ajuda para resolver as listas)?



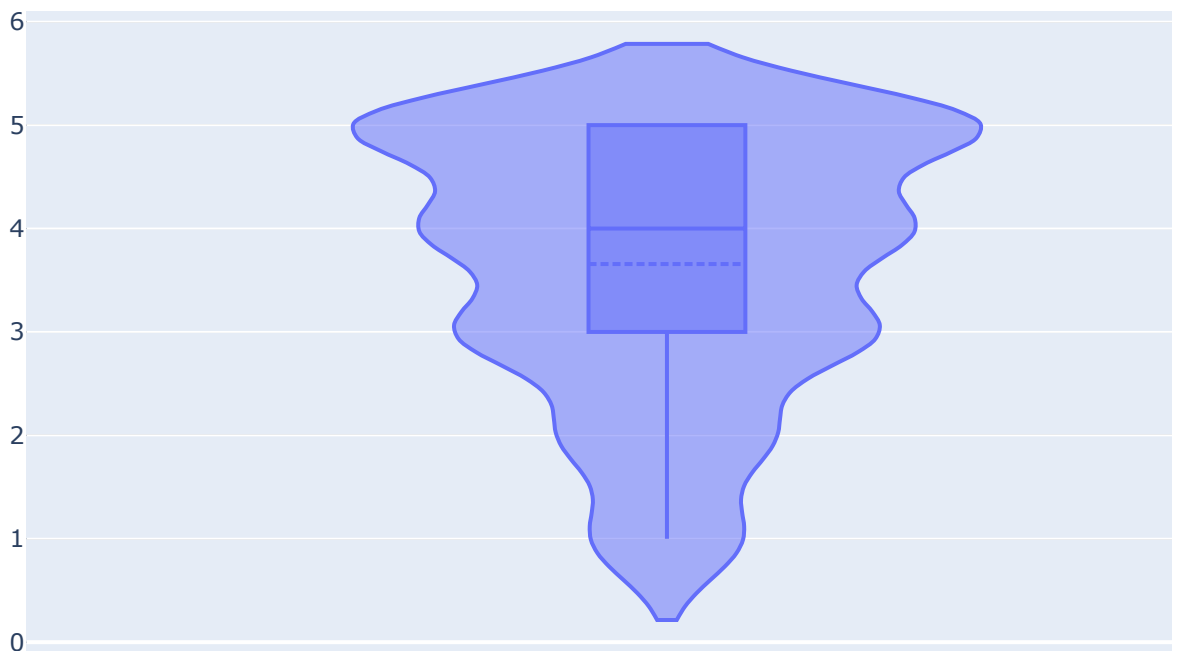
3.1. É favorável em usar EXERCÍCIOS unificados em TODAS as turmas, NIVELANDO as avaliações, e NIVELANDO o aprendizado do aluno (por exemplo, com as listas no Moodle), porém sem tirar a carga professor nas metodologias de ensino?



3.2. Na sua opinião, as CORREÇÕES AUTOMÁTICAS dos exercícios no Moodle, apresentando a NO automaticamente, ajudaram no aprendizado?

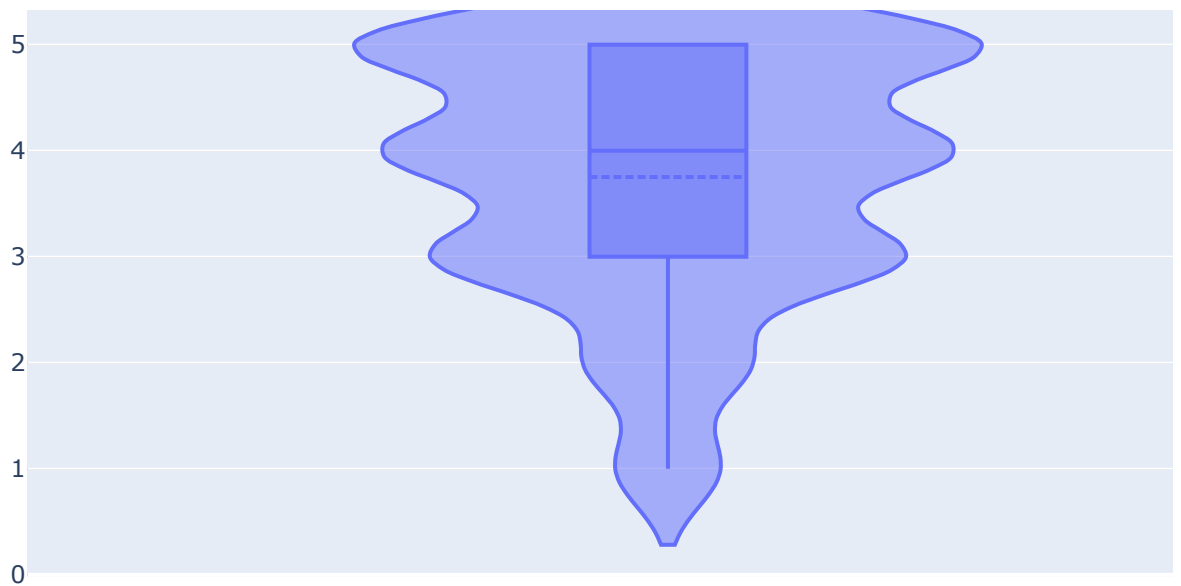


3.3. Na sua opinião, os FEEDBACKS (mensagens de erros em inglês que o python retorna quando o tentar avaliar uma atividade) nas correções automáticas dos exercícios no Moodle ajudaram no :

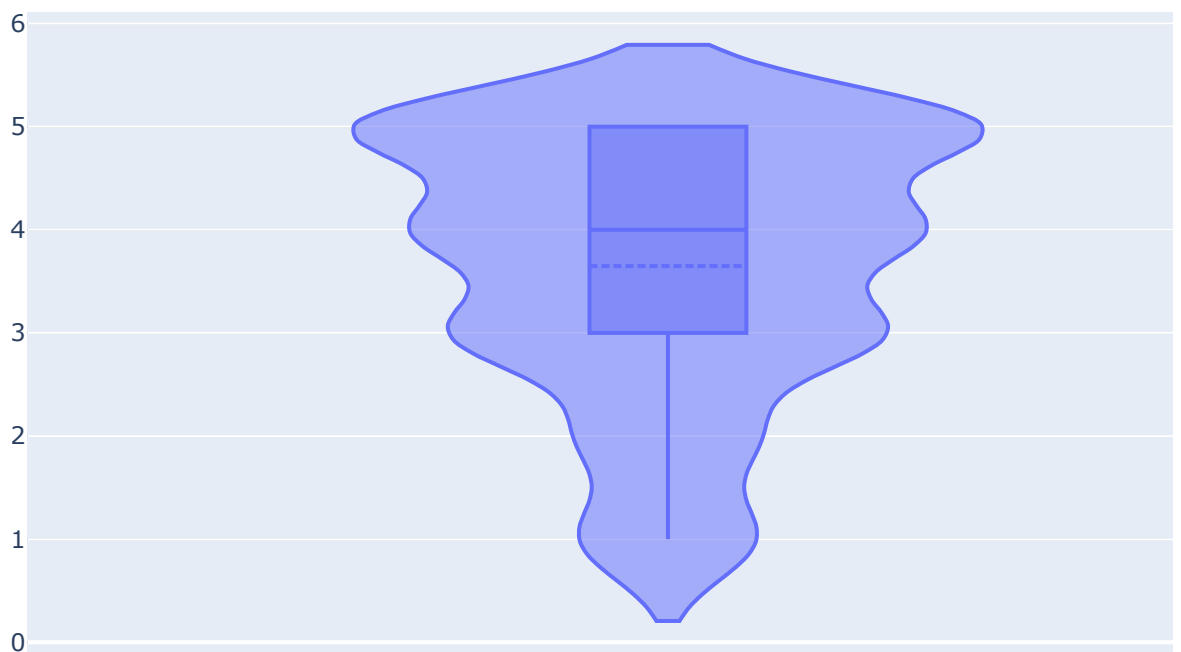


3.5. Na sua opinião, as LISTAS INDIVIDUAIS COM EXERCÍCIOS DISTINTOS, com pequenas alterações embaralhados (enviadas para o email do aluno) ajudaram a DIMINUIR O PLÁGIO na disciplina, log aprendizado, oferecendo avaliações mais JUSTAS?



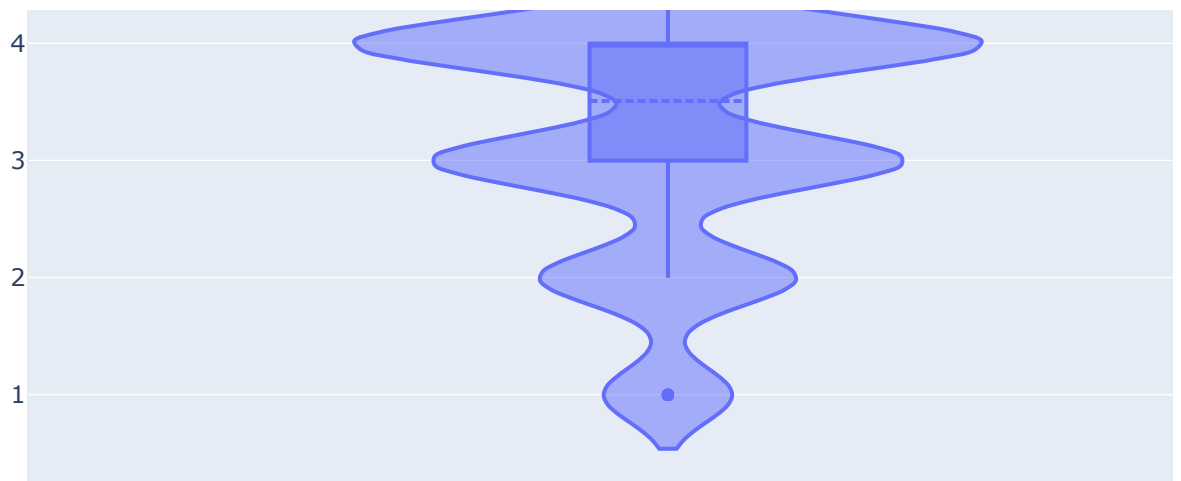


3.6. Na sua opinião, se aumentar o número de questões a serem sorteadas diminuiria o PLÁGIO? Por uma lista de 5 questões de múltipla-escolha, existe estudo afirmando que o ideal seria sortear 5 um banco de 25 questões distintas [ref: <https://www.tandfonline.com/doi/abs/10.1080/0260293>]

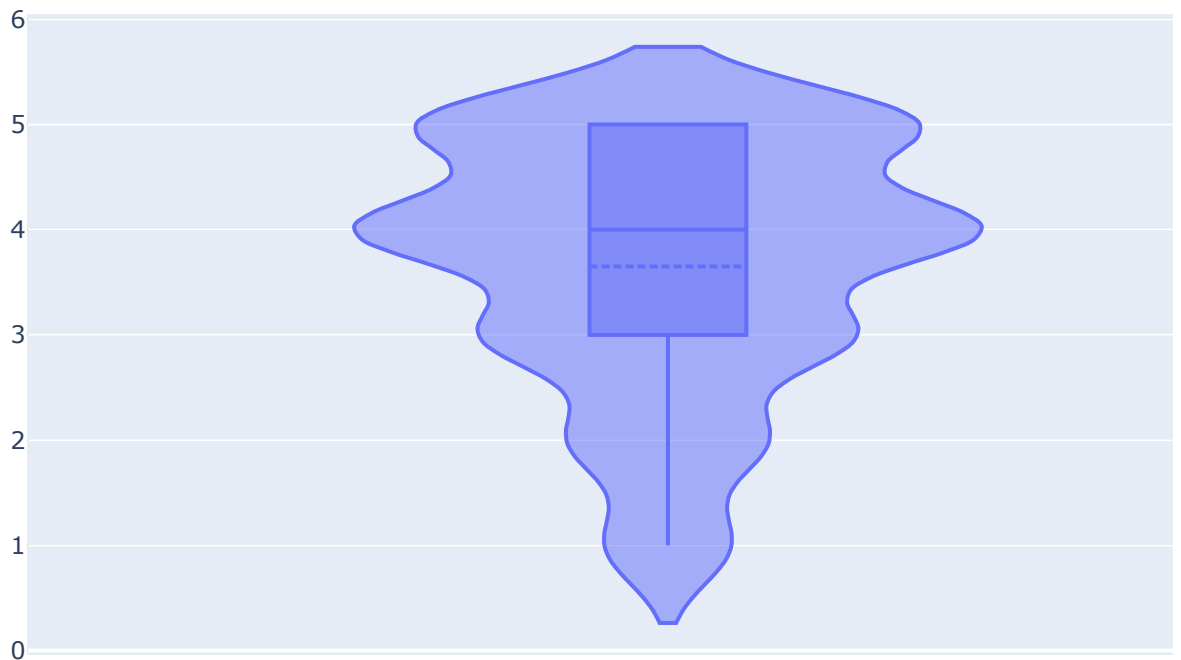


4.1. Qual é a sua avaliação GERAL para essa oferta do BCC em 2020-QS, quanto aos recursos GERAIS utilizados nas aulas (Slides, Notebooks, Vídeos, etc.)?

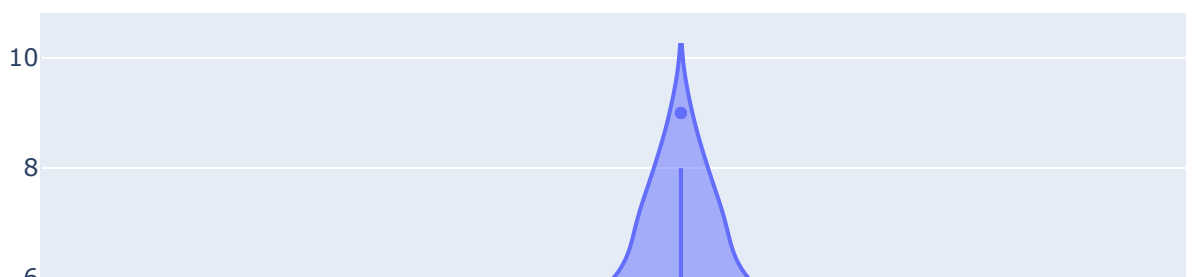


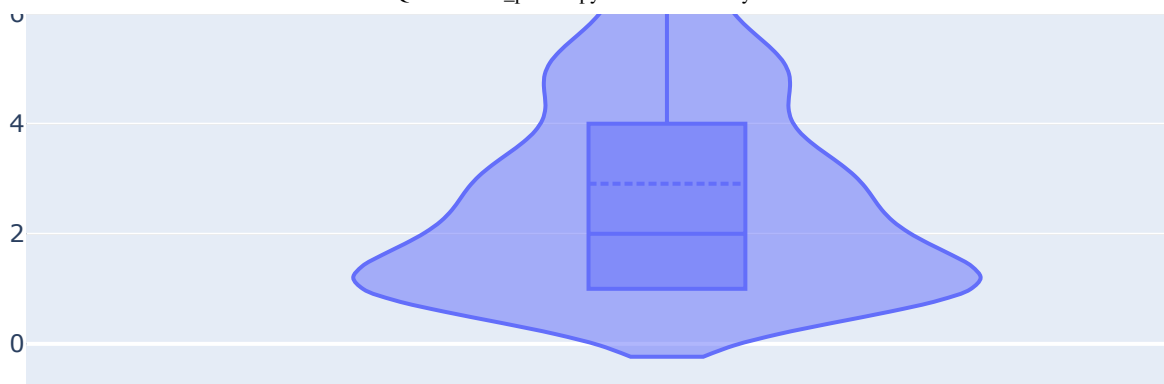


4.2. Qual é a sua avaliação GERAL para essa oferta do BCC em 2020-QS, quanto aos recursos u AVALIAÇÕES (Moodle, Listas, Avaliação Automática, etc.)?

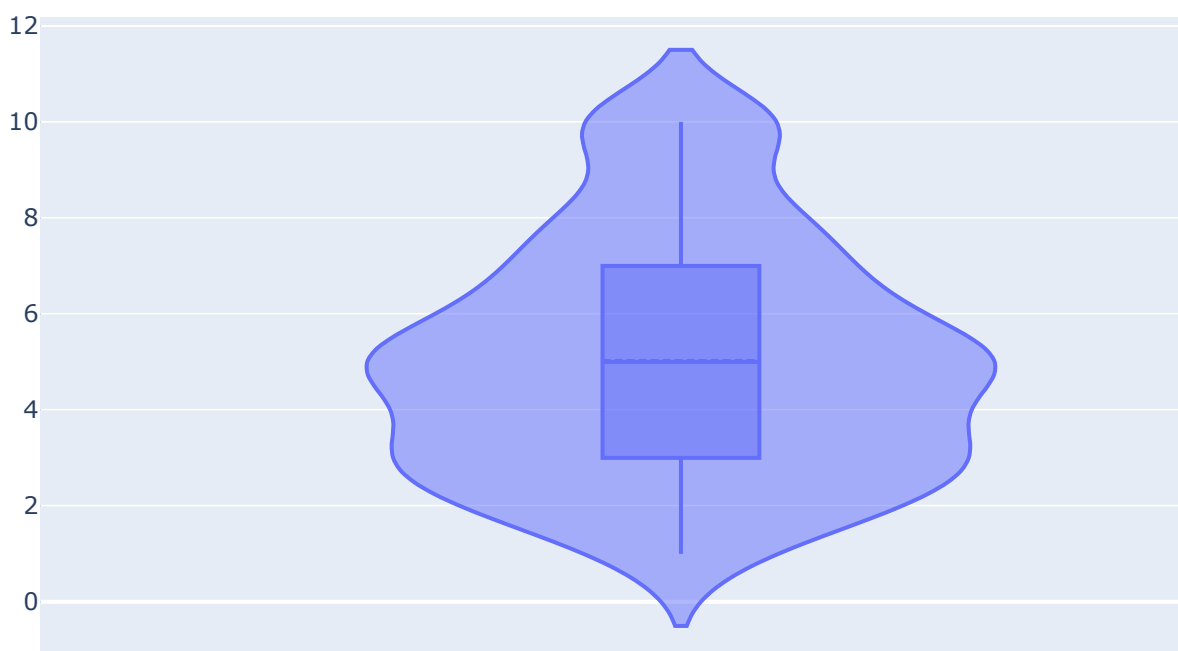


3.4. DO SEU CONHECIMENTO, qual foi a porcentagem de PLÁGIO existente nas entregas das listas Moodle? Copiar o código de colegas e fazer pequenas alterações (em comentários, em espaçament das variáveis) É PLÁGIO, porém, tirar dúvidas com colegas e monitores NÃO É PLÁGI

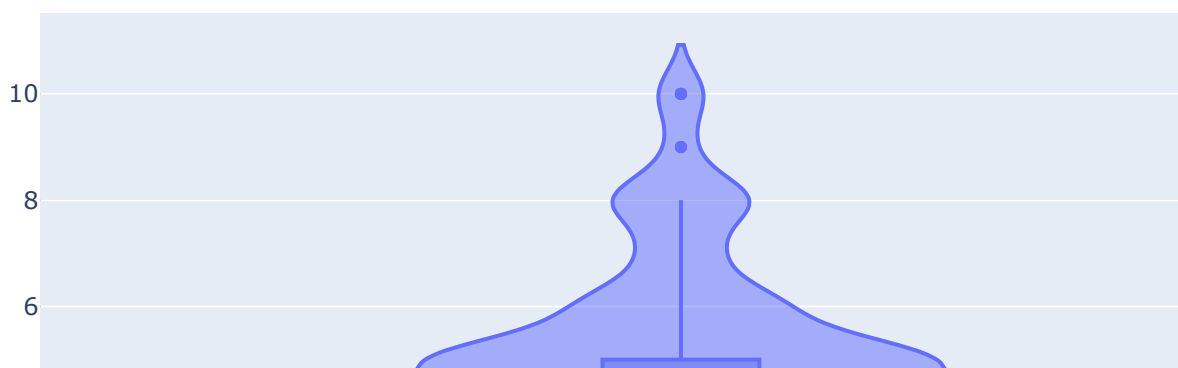




3.7. Na sua opinião, qual seria o melhor PESO atribuído às LISTAS no conceito final?



3.8. Na sua opinião, qual seria o melhor PESO atribuído à(s) PROVA(S) - se adotada(s) na sua conceito final? Sem considerar a prova de recuperação.



▼ Gráfico de pizza

Para as questões com categorias

```

0
1 # quando um aluno responde em mais de uma opção
2 def addCadehoria(df,col):
3     categorias = df[col].value_counts().index.tolist()
4     v = df[col].value_counts().tolist()
5     dictCateg = {}
6     for key, value in enumerate(categorias):
7         s = str(value).split(', ')
8         if len(s)==1:
9             dictCateg[value] = int(v[key])
10    #print(dictCateg)
11    for value in df[col].tolist():
12        s = str(value).split(', ')
13        if len(s)>1:
14            for v in s:
15                v = v.strip()
16                try:
17                    dictCateg[v] += 1
18                except KeyError:
19                    dictCateg[v] = 1
20                #print('###',v,dictCateg[v])
21    #print(dictCateg)
22    return(dictCateg)
23
24 def drawPiePlot(df,rows,col):
25     plt.clf()
26
27     dictCateg = addCadehoria(df,col)
28
29     categorias = []
30     v = []
31     for key in dictCateg:
32         if dictCateg[key] < 5:
33             categorias.append('')
34             print('###',key,dictCateg[key])
35         else:
36             categorias.append(key)
37             v.append(dictCateg[key])
38
39     myVal = np.array(v)
40     r = np.zeros_like(myVal)
41     r[myVal==myVal.max()]=0.2
42
43     def myFunc(pct, allvals):
44         val = pct/100.*np.sum(allvals)
45         if pct>1.5:
46             return "{:.0f}%\n{:.0f}".format(pct, val)
47         else:
48             return ''

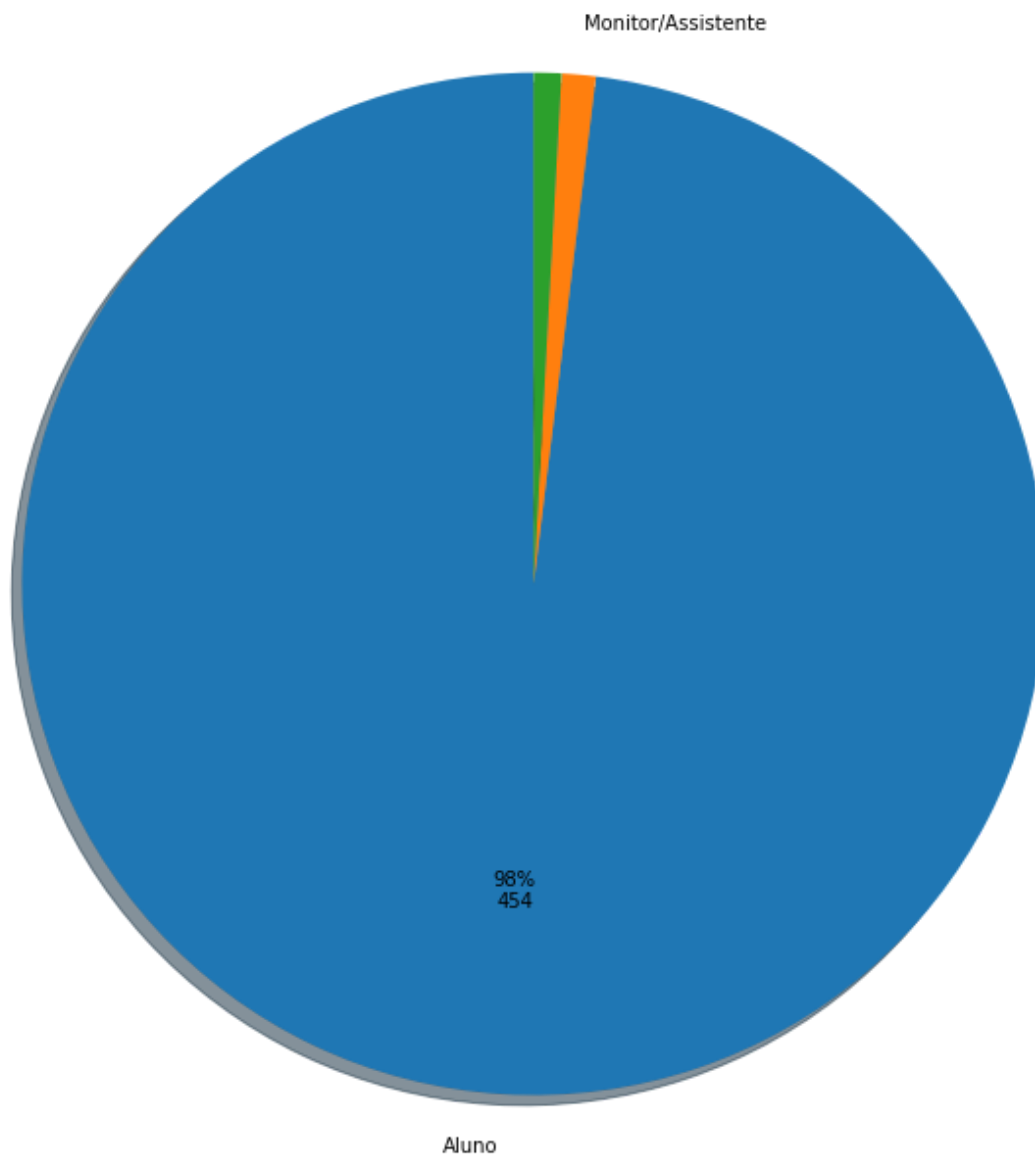
```

```
49
50 %matplotlib inline
51
52 categorias = ['\n'.join(textwrap.wrap(categorias[c], 33)) for c in range(len
53 fig, ax = plt.subplots(figsize=(16, 12))
54 ax.pie(v, labels=categorias, explode=r, autopct=lambda pct: myFunc(pct, myV
55 #ax.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circl
56 s = '\n'.join(textwrap.wrap(rows[col], 110)) + '\n\n(TOTAL = ' + str(np arra
57 plt.title(s)
58 plt.show()
59 plt.close()
60
61 if SAVE_FIGS_PAPER:
62     file = 'fig_col_'+ str(col) + '.png'
63     plt.tight_layout()
64     extent = ax.get_window_extent().transformed(fig.dpi_scale_trans.inverted()
65     fig.savefig(file, bbox_inches=extent, dpi=DPI_resolution)
66
67     files.download(file)
68
69 for i in colsCateg[2:-2]:
70     drawPiePlot(df, rowsSTR, i)
```

Professor 4

1.3. Você é um

(TOTAL = 463 respostas em 3 categorias)

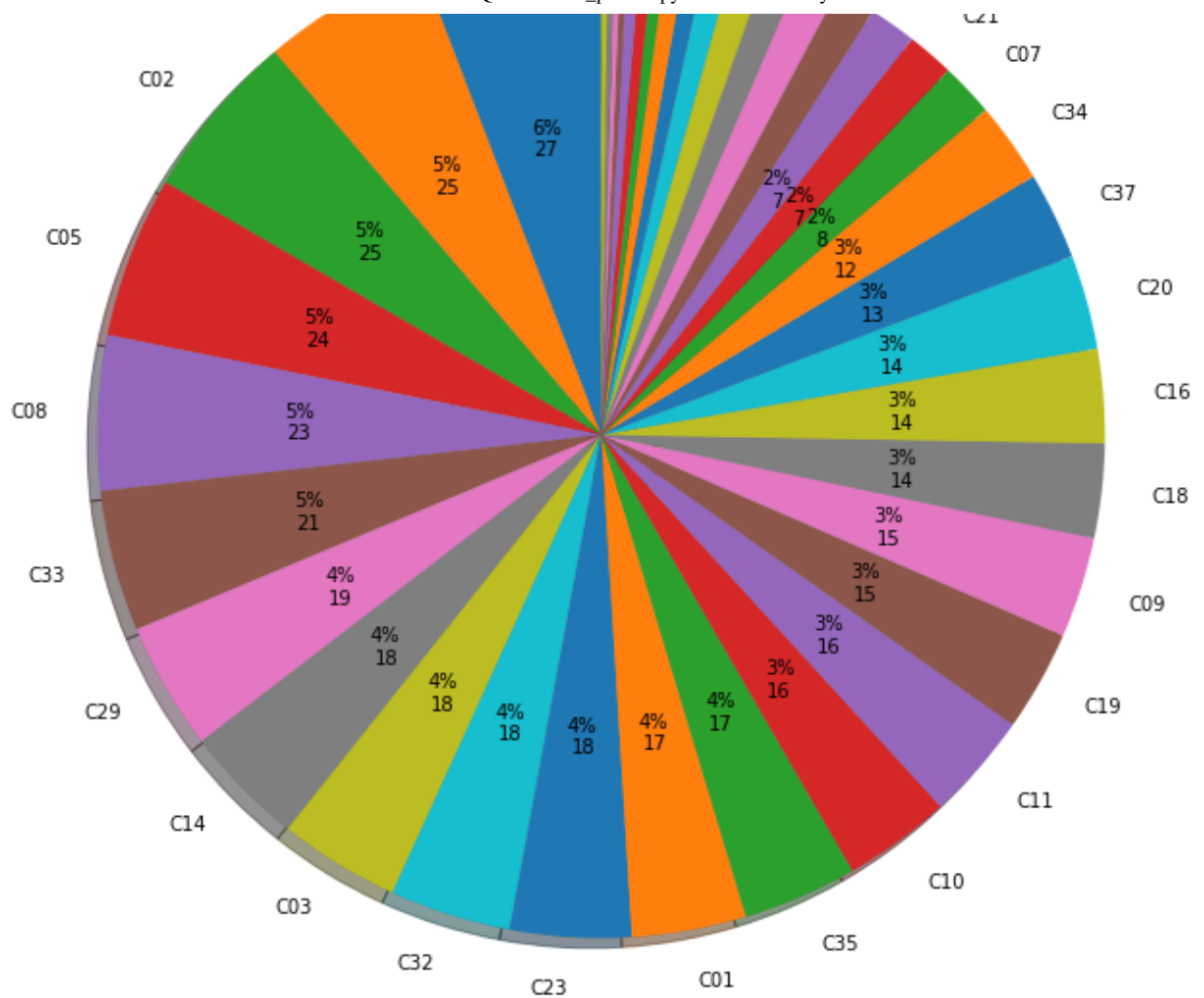


- ### C31 4
- ### C22 3
- ### C38 3
- ### C25 2
- ### C04 2
- ### C28 2
- ### C12 1
- ### C27 1
- ### C17 1
- ### C13 1

1.4. Sua(s) Turma(s) [Sua(s) turma(s)]

(TOTAL = 463 respostas em 39 categorias)

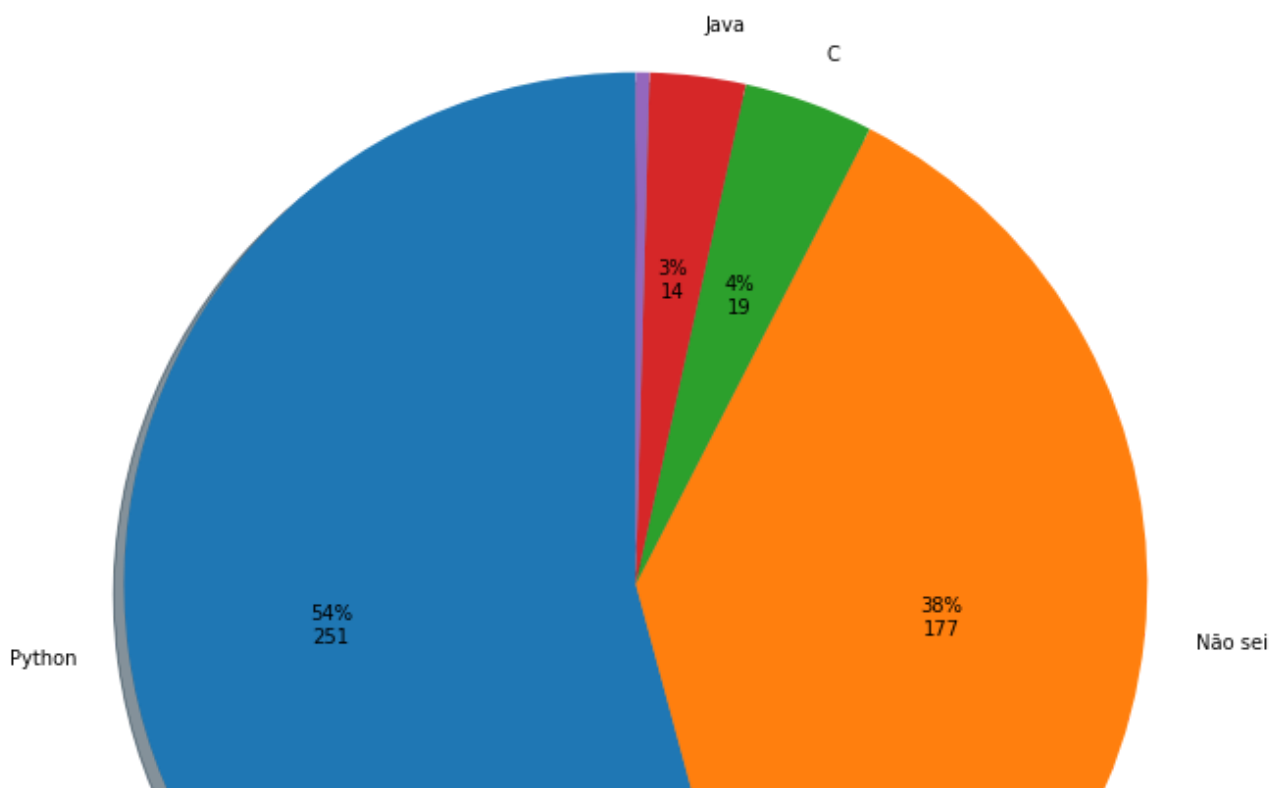




Basic 2

2.2. Para você, qual é a melhor LINGUAGEM capaz de lidar com toda a ementa (base de dados, gráficos, estatística descritiva, correlação-regressão, lógica de programação, e modelagem-simulação)?

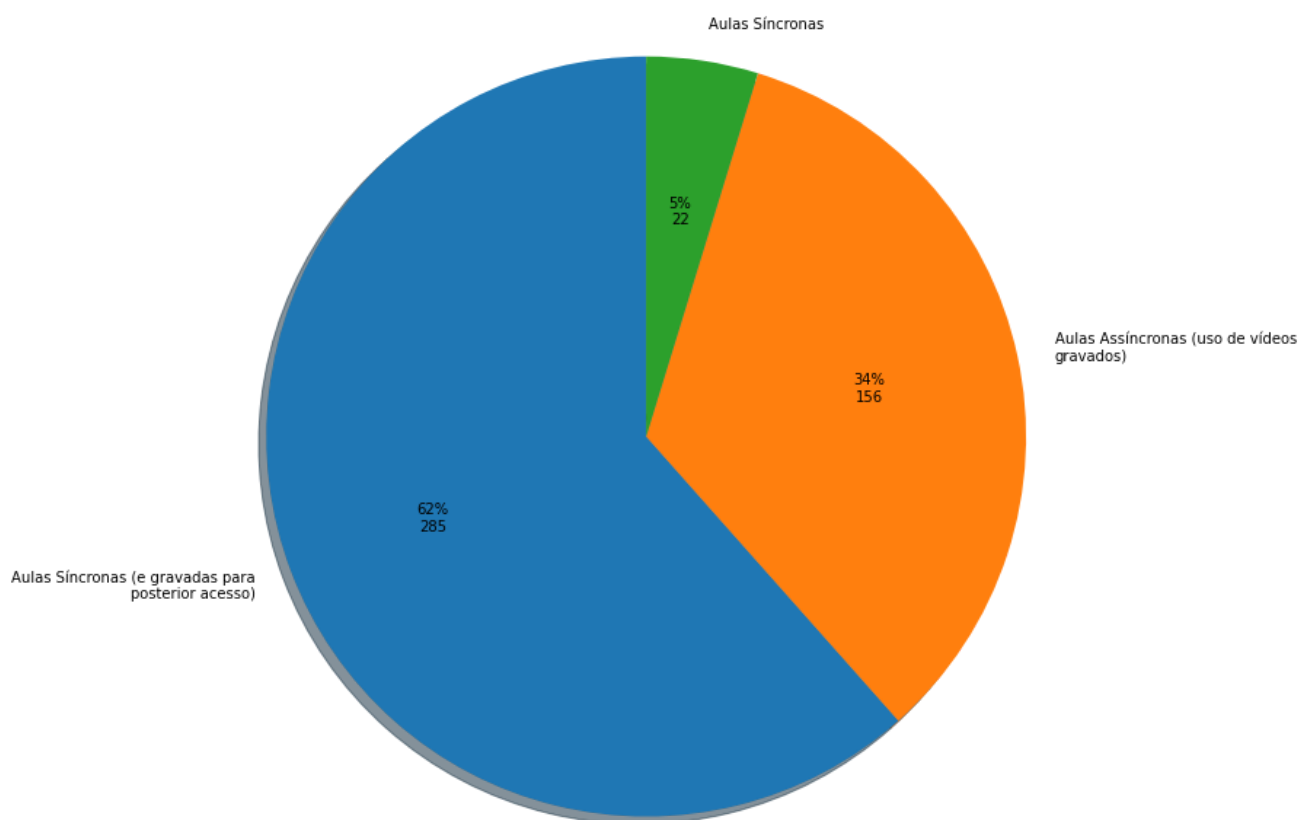
(TOTAL = 463 respostas em 5 categorias)





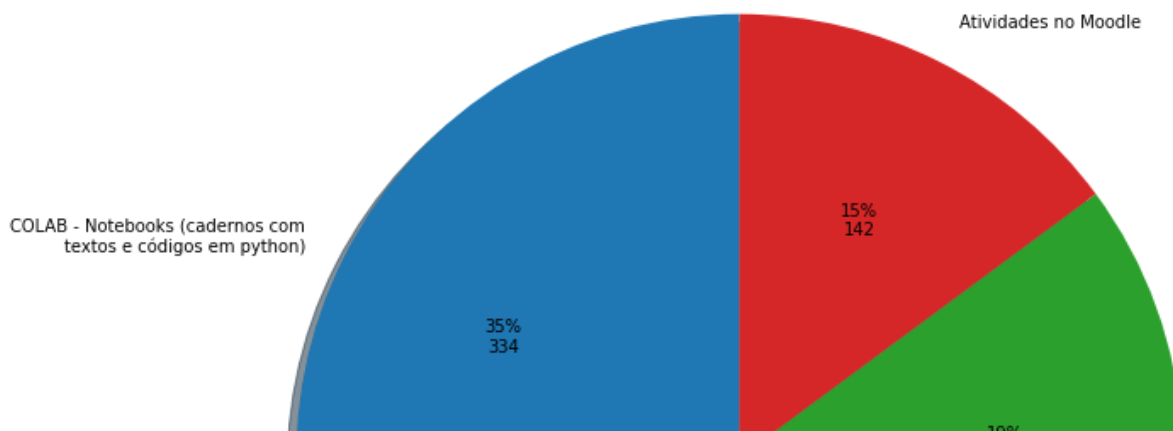
2.3. E quais as melhores METODOLOGIAS de ensino utilizadas?

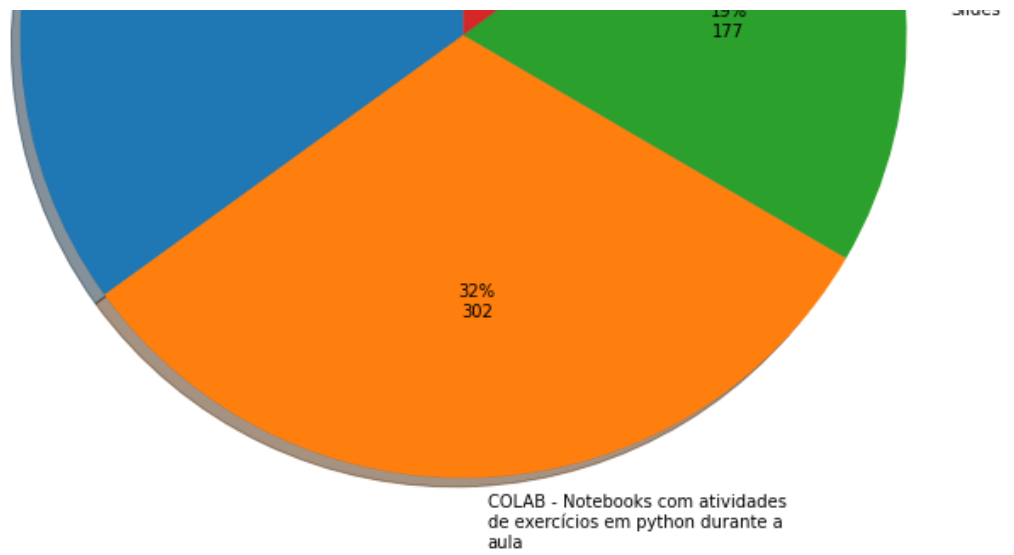
(TOTAL = 463 respostas em 3 categorias)



2.4. Quais os melhores RECURSOS utilizados DURANTE as aulas síncronas?

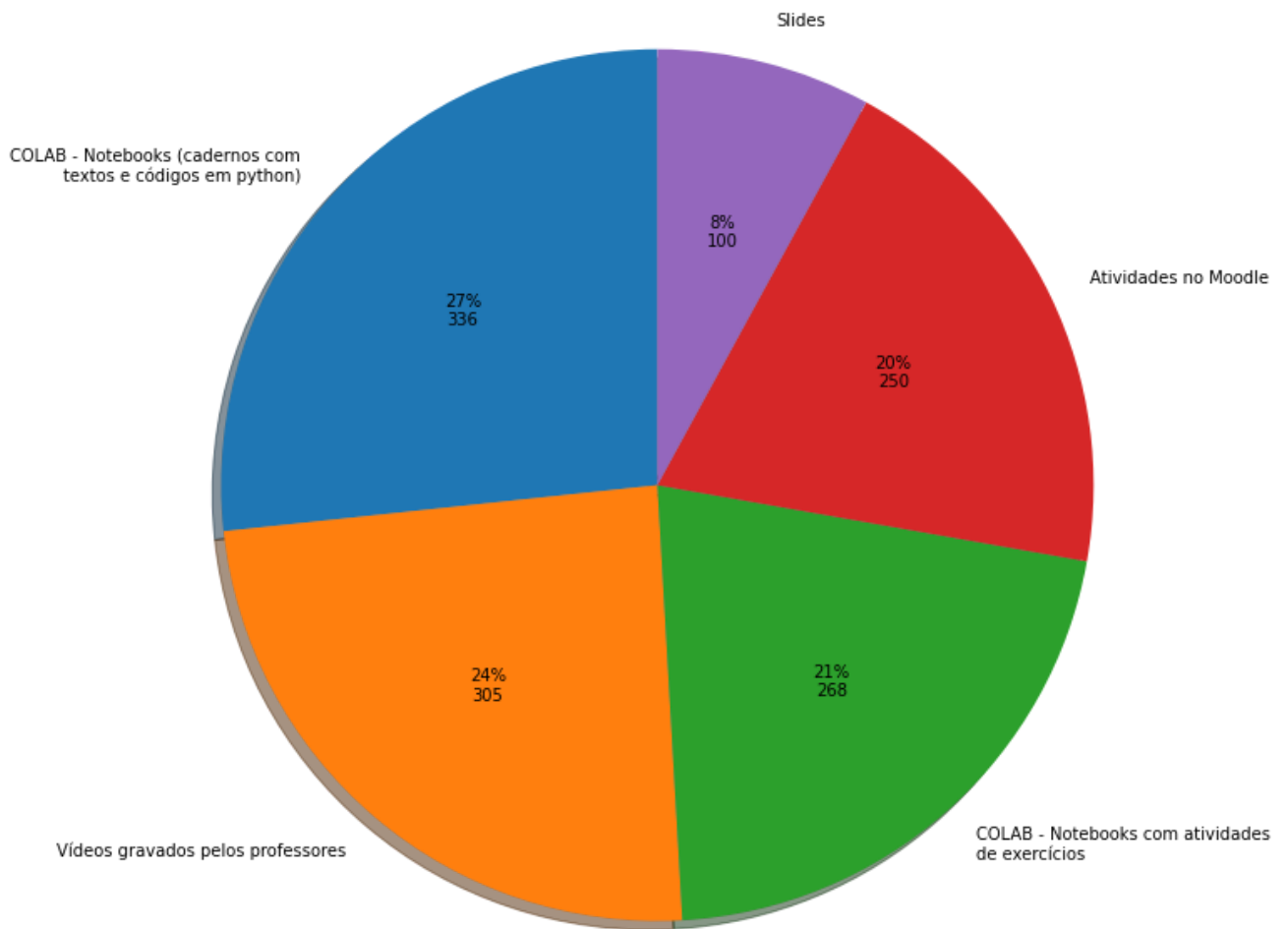
(TOTAL = 955 respostas em 4 categorias)





2.5. E quais os melhores RECURSOS utilizados para estudo FORA do horário da aula?

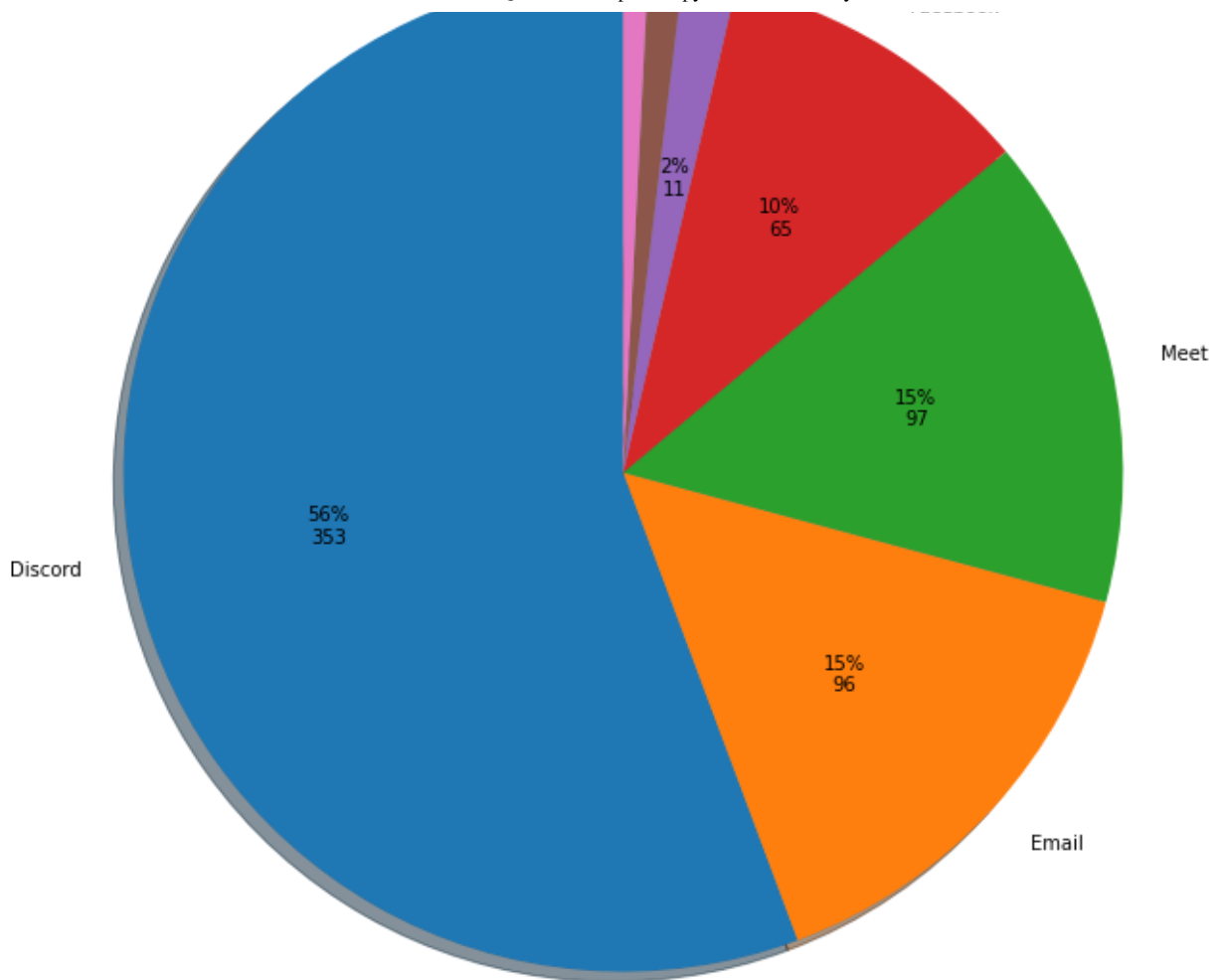
(TOTAL = 1259 respostas em 5 categorias)



2.8. Qual(is) foi(ram) o(s) melhor(es) recurso(s) utilizado(s) para COMUNICAÇÃO (por exemplo, para ter ajuda dos monitores para resolver as listas)?

(TOTAL = 634 respostas em 7 categorias)





▼ Questão Aberta

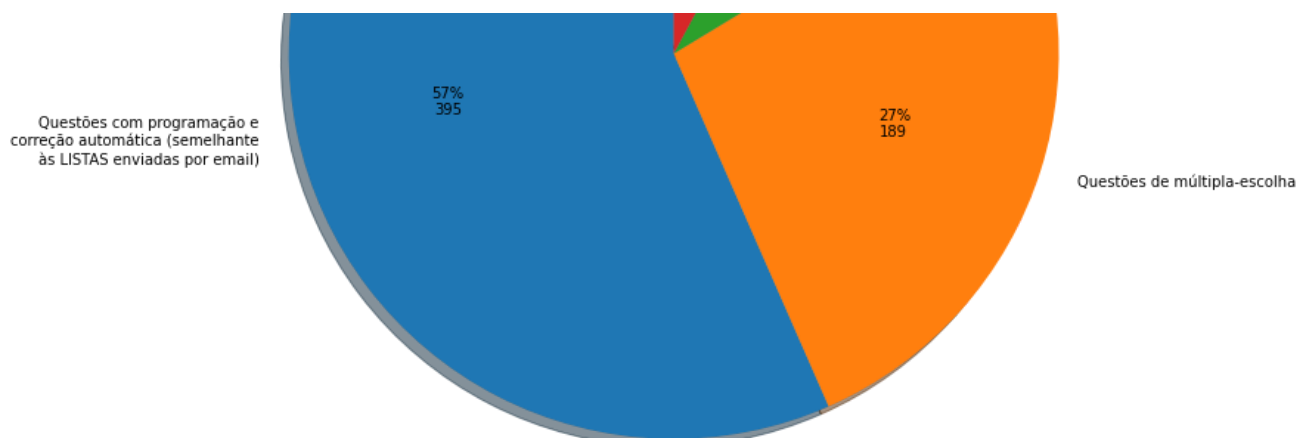
mínimo de 72h para entrega?

```

1 key = 0
2 for value in df[24]:
3     try:
4         if len(value)>3:
5             key += 1
6             s = value.replace('\n', ' ')
7             print('\n%s'%str(key).zfill(3),'\n'.join(textwrap.wrap(s, 80)))
8     except:
9         pass

```

1



✓ 0s conclusão: 10:19

