

A laptop screen is shown in the background, displaying a data analytics interface. At the top, there is a line chart with a blue line showing fluctuations. Below it, a pie chart is visible, with a legend indicating 'New Visitor' and 'Returning Visitor'. The screen is slightly out of focus, and the overall image has a dark, muted color palette.

RELATO DE EXTENSÃO

# Tópicos de Big Data em Python

# Identificação dos Alunos



JAMILY AGUIAR

**202201088745**



LUCAS RAFAEL

**202309855046**



LYCON HEDA

**202303247419**



THIAGO BRUNO

**202202965731**

# Identificação dos Parceiros

## **Centro de Ensino Vinho Novo**

Instituição parceira localizada no bairro Redenção, com atuação na área de Educação Infantil e reforço escolar, atendendo crianças da comunidade local.

**Elizabeth Aguiar Ribeiro e Jonas Sá Ribeiro**, fundadores e gestores do Centro de Ensino Vinho Novo. Desde 2019, desenvolvem ações sociais e educacionais com foco no acolhimento e desenvolvimento de crianças da comunidade.



# Identificação do Problema

	Nome	DIA	FEV	MAR	ABR	MAI	JUN	JULH	AGOS	SE
1	CARLOS MIGUEL	05	350,00	350,00	350,00	350,00				
2	DAVID RAMOM	05	350,00	350,00	350,00	350,00				
3	DANIEL SOARES	15	440,00	440,00	440,00	440,00				
4	ESTHER LUIZA	05	440,00	440,00	440,00	440,00				
5	EFRAIM MORAES	07	400,00	400,00	400,00	400,00				
6	GUILHERME	05	440,00	440,00	440,00	440,00				
7	JOSEPH NASC	05	400,00	400,00	400,00	400,00				
8	KENNEDY	12	440,00	440,00	440,00	440,00				
9	MIRELLA NEVES	05	440,00	440,00	440,00	440,00				
10	RAYLA VITORIA	05	350,00	350,00	350,00	350,00				
11	ALICE NICOLES	05	440,00	440,00	440,00	440,00				
12	Arthur Rulon	05	440,00	440,00	440,00	440,00				

FLUXO DE CAIXA			
	ENTRADA	SAÍDA	SALDO
R\$	-	R\$	-R\$
		87.687,76	87.687,76

DATA	MOVIMENTAÇÃO	DESCRIÇÃO	VALOR
01/05/2025	Saída	Parcela Empréstimo bemol	R\$ 1.676,68
01/05/2025	Saída	Fatura do cartão Carrefour	R\$ 1.063,00
03/05/2025	Saída	Resisão Eva	R\$ 250,00
03/05/2025	Saída	Edy Contadora	R\$ 500,00
03/05/2025	Saída	Cartão empresa	R\$ 1.760,00
05/05/2025	Saída	Conta de energia escola	R\$ 2.183,97
06/05/2025	Saída	Aluguel da casa Hileia	R\$ 2.200,00
06/05/2025	Saída	última parcela recisão Izhadora	R\$ 748,26
05/05/2025	Saída	Empréstimo mercado pago	R\$ 1.469,92

21/05/2025	Saída	Riachuelo	R\$ 138,00
21/05/2025	Saída	Aniel	R\$ 1.000,00
21/05/2025	Saída	energia alvorada	R\$ 816,10
21/05/2025	Saída	Aluguel auditório alvorada dia das mães	R\$ 250,00
23/05/2025	Saída	Carrefour	R\$ 1.063,00
29/05/2025	Saída	Edy contadora	R\$ 1.000,00
29/05/2025	Saída	Mercado	R\$ 200,00
29/05/2025	Saída	Parcela do carro	R\$ 1.776,77
28/05/2025	Saída	Parcela Bemol	R\$ 343,78
28/05/2025	Saída	Negociação empréstimo bemol	R\$ 859,83
28/05/2025	Saída	Parcela Otica	R\$ 348,25
28/05/2025	Saída	Receita federal abril 2024/ Atraso	R\$ 325,09
30/05/2025	Saída	Darf receita federal	R\$ 526,82
30/05/2025	Saída	Taxa do cartório	R\$ 74,37

# Objetivos

- **Organizar** os dados financeiros existentes da escola (entradas e saídas) no ambiente do Google Colab, separando corretamente as informações por unidade (Redenção e Alvorada), para facilitar a visualização e compreensão dos gastos.
- **Fornecer** à equipe gestora um relatório claro e visual com os resultados obtidos a partir da análise, apoiando a tomada de decisões estratégicas e a definição de metas específicas para o equilíbrio financeiro entre as unidades.
- **Analisar** os dados financeiros organizados com base em critérios definidos em conjunto com os gestores da escola, identificando padrões de consumo, desequilíbrios e oportunidades de melhoria.

# Demonstrações

```
✓ [94] # Importa a biblioteca Pandas com apelido pd
Ds import pandas as pd
```

```
✓ [94] # Importa a biblioteca Pandas com apelido pd
Ds import pandas as pd
```

```
✓ [96] # Exibe o nome das planilhas existentes no arquivo
Ds print(df.sheet_names)
```

```
⇒ ['JAN', 'FEV', 'MAR', 'ABR', 'MAI', 'JUN', 'JUL', 'AGO', 'SET', 'OUT', 'NOV', 'DEZ']
```

```
✓ [97] # Armazena na variável 'meses_preenchidos' os meses que possuímos os dados
Ds meses_preenchidos = ['JAN', 'FEV', 'MAR', 'ABR', 'MAI']
```

```
# para identificar a qual mês os dados pertencem
df_valido = pd.concat(
    [pd.read_excel(df, sheet_name=nome, header=4).assign(Mês=nome) for nome in meses_preenchidos],
    ignore_index=True)
```

```
✓ [105] # Usa a primeira coluna DATA como base e preenche os valores da primeira com
Ds # com os valores da segunda
if len(data_cols) > 1:
    data_completa = df_valido.iloc[:, data_cols[0]]
    for i in data_cols[1:]:
        data_completa = data_completa.combine_first(df_valido.iloc[:, i])
```

```
✓ [106] # Substitui os valores da coluna DATA do dataframe df_valido pelos valores data_completa
Ds df_valido.drop(columns=df_valido.columns[data_cols], inplace=True)
df_valido['DATA'] = data_completa
```

```
✓ [107] # Substitui as colunas DATA pelo resultado unificado
Ds df_valido = df_valido.drop(columns='DATA')
df_valido.insert(0, 'DATA', data_completa)
```

```
✓ [108] # Resultado após a unificação das colunas DATA
Ds print(df_valido.columns)
```

```
⇒ Index(['DATA', 'MOVIMENTAÇÃO', 'DESCRIÇÃO', 'VALOR', 'OBSERVAÇÃO', 'MÊS'], dtype='object')
```

```
✓ [101] # Remove colunas com nome 'Unnamed'
Ds df_valido = df_valido.loc[:, ~df_valido.columns.str.contains('^Unnamed')]
```

```
✓ [102] # Uniformiza os nomes do dataframe, removendo os espaços em brancos dos lados
Ds # e convertendo todos os caracteres para maiúsculo
df_valido.columns = df_valido.columns.str.strip().str.upper()
```

```
✓ [103] # Demonstrar o resultado após a remoção da coluna 'Unnamed' e padronização dos nomes
Ds # das colunas
print(df_valido.columns)
```

```
⇒ Index(['DATA', 'MOVIMENTAÇÃO', 'DESCRIÇÃO', 'VALOR', 'OBSERVAÇÃO', 'MÊS',
        'DATA'],
        dtype='object')
```

```
✓ [104] # Cria uma lista com índices das colunas com nome DATA
Ds data_cols = [i for i, col in enumerate(df_valido.columns) if col == 'DATA']
```

```
[99] # Mostra as 5 primeiras linhas do dataframe df_valido
print(df_valido.head())
```

```
⇒
```

	DATA	MOVIMENTAÇÃO	DESCRIÇÃO	VALOR \
0	NaN	Saída	Comprar HD	200.00
1	NaN	Saída	conta energia/ ref dezembro escola	3245.98
2	NaN	Saída	agua/ novembro 2024	1180.97
3	NaN	Saída	Material de expediente	132.06
4	NaN	Saída	J s Ribibeiro / grafica	150.00

	OBSERVAÇÃO	Mês	Unnamed: 0	DATA
0	pix	JAN	NaN	NaN
1	pix	JAN	NaN	NaN
2	pix	JAN	NaN	NaN
3	pix	JAN	NaN	NaN
4	pix	JAN	NaN	NaN

```
[100] # Exibe os nomes das colunas do dataframe df_valido
print(df_valido.columns)
```

```
⇒ Index([' DATA ', 'MOVIMENTAÇÃO', 'DESCRIÇÃO', 'VALOR', 'OBSERVAÇÃO', 'Mês',
        'Unnamed: 0', 'DATA'],
        dtype='object')
```



# Demonstrações

```
[116] # Elimina as linhas na qual a coluna valor está ausente
df_valido = df_valido.dropna(subset=['VALOR'])
```

```
[118] # Importa a biblioteca matplotlib para criação de gráficos
import matplotlib.pyplot as plt
```

```
[119] # Cria uma lista com os meses ordenados
meses_ordenados = ['JAN', 'FEV', 'MAR', 'ABR', 'MAI']
```

```
[120] # Converte a coluna MÊS em uma categoria ordenada segundo a lista meses_ordenados
df_valido['MÊS'] = pd.Categorical(df_valido['MÊS'], categories=meses_ordenados, ordered=True)
```

```
[121] # Agrupa os dados por mês e soma os valores da coluna VALOR para cada mês
gastos_por_mes = df_valido.groupby('MÊS')['VALOR'].sum()
```

```
[113] # Preenche os dados vazios da coluna DATA com a str 'Data não informada'
df_valido['DATA'] = df_valido['DATA'].fillna("Data não informada")
```

```
df_valido.head()
```

	DATA	DESCRIÇÃO	VALOR	MÊS
0	Data não informada	Comprar HD	200.00	JAN
1	Data não informada	conta energia/ ref dezembro escola	3245.98	JAN
2	Data não informada	agua/ novembro 2024	1180.97	JAN
3	Data não informada	Material de expediente	132.06	JAN
4	Data não informada	J s Ribeiro / grafica	150.00	JAN

Próximas etapas: [Gerar código com df\\_valido](#) [Ver gráficos recomendados](#) [New interactive sheet](#)

```
[115] # Verifica quantos dados ausentes há na coluna VALOR
df_valido['VALOR'].isna().sum()
```

```
np.int64(2)
```

```
[109] # Copia o dataframe df_valido para df_backup
df_backup = df_valido.copy()
```

```
# Salva o dataframe em um novo arquivo excel
df_valido.to_excel('/content/drive/MyDrive/df_valido_tratado.xlsx', index=False)
```

```
# Cria um gráfico de barras com dados dos gastos de cada mês
gastos_por_mes.plot(kind='bar', color='steelblue')
```

```
plt.title('Gastos por Mês')
plt.xlabel('Mês')
plt.ylabel('Valor Total (R$)')
plt.xticks(rotation=0)
plt.show()
```

Gastos por Mês



## Gastos por Mês

