

Apresentação

TVLLE

EMANUELA CAETANO
LEONARDO FABRÍCIO
LUCAS ALCÂNTARA
TALYTТА OLIVEIRA
VITOR BRANDÃO



TVLLE

CONSULTORIA ORIENTADA
POR DADOS

TEMA: MERCADO DE BEBIDAS
ASSUNTO: MERCADO DE VINHOS
MUNDIAL



Principais objetivos

IDEIA

Desenvolver uma empresa com base em nossos princípios para a prestação de consultoria para a transformação de dados em informações.

OBJETIVO

Cumprir os requisitos obrigatórios e desejáveis do trabalho com excelência.

JUSTIFICATIVA

Disponibilidade de dados gratuitos na internet e relevância do assunto para o tema.

Missão

Entregar aos nossos clientes análises objetivas e eficientes sobre a condição do respectivo segmento de mercado

Valores

Ser referência na área de dados como empresa que visa entregar o que é proposto com responsabilidade e qualidade

Propósito

Integridade, honestidade, responsabilidade, compromisso com os clientes, diversidade e inclusão, aprendizado, trabalho em equipe e qualidade

Datasets iniciais

Fontes:

Organização Internacional da Vinha
e do Vinho

Kaggle

Taylor & Francis Online

Datasets brutos



Google Cloud Storage

 [brazilian_wine_data.csv](#)

 [importacao-exportacao-producao...](#)

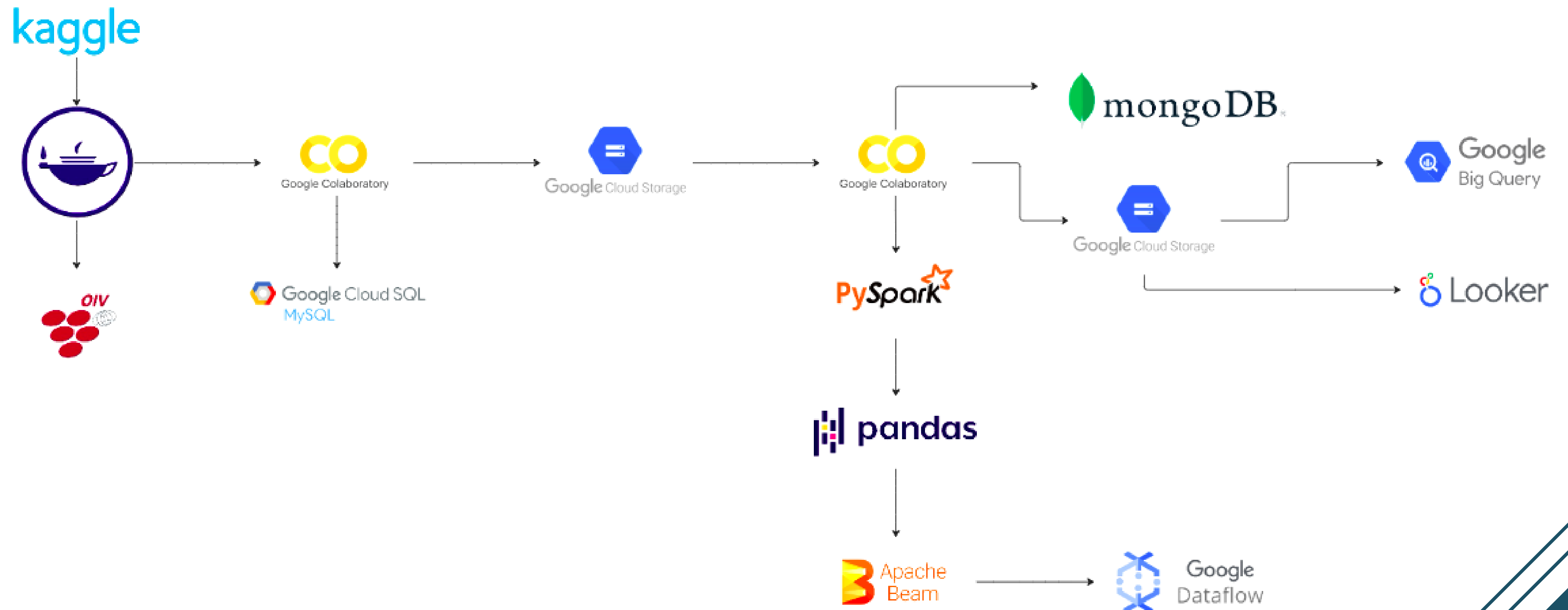
 [prod-BR1.csv](#)

 [prod-BR2.csv](#)

 [prod-BR3.csv](#)

 [winemag-data_first150k.csv](#)

WorkFlow



Conexão MySQL

Conexão

```
[ ] 1 #FUNÇÃO PARA CONECTAR AO BANCO
    2 def conexao_db(servidor, usuario, senha, nome_do_banco):
    3     conexao = None
    4     try:
    5         conexao = mysql.connector.connect(host=servidor, user=usuario, passwd=senha, database=nome_do_banco)
    6         print(f'Conexão ao banco {nome_do_banco} efetuada com sucesso')
    7     except:
    8         print(f'Erro ao conectar ao banco {nome_do_banco}')
    9
   10     return conexao

[ ] 1 con = conexao_db('34.162.75.223', 'root', 'root', 'mercado_vinho')

[ ] 1 pd.read_sql('show databases', con)
```

Envio MySQL

Realizar conexão com o servidor | criar uma SQLAlchemy engine para conectar com o MySQL | convertendo os dataframes em uma table do sql

```
[ ] 1 servidor = '34.162.75.223'
    2 nome_do_banco = 'mercado_vinho'
    3 usuario = 'root'
    4 senha = 'root'
    5
    6 engine = create_engine("mysql+pymysql://root:root@34.162.75.223/mercado_vinho"
    7                        .format(host=servidor, db=nome_do_banco, user=usuario, pw=senha))
    8

[ ] 1 df1.to_sql('dados_vinho_brasil', engine, index=True)
    2 df2.to_sql('producao_consumo_importacao_exportacao_uvas_vinhos_global', engine, index=True)
    3 df3.to_sql('avaliacoes-vinhos-global', engine, index=True)
    4 df4.to_sql('prod-vinho-BR1', engine, index=True)
    5 df5.to_sql('prod-vinho-BR2', engine, index=True)
    6 df6.to_sql('prod-vinho-BR3', engine, index=True)
```


Database MySQL

```
mysql> show databases;
+-----+
| Database |
+-----+
| Z_README_TO_RECOVER |
| information_schema |
| mercado_vinho |
| mysql |
| performance_schema |
| sys |
+-----+
6 rows in set (0.02 sec)
```

Tabela MySQL

```
mysql> use mercado_vinho;
Database changed
mysql> show tables;
+-----+
| Tables_in_mercado_vinho |
+-----+
| avaliacoes-vinhos-global |
| dados_vinho_brasil      |
| prod-vinho-BR1          |
| prod-vinho-BR2          |
| prod-vinho-BR3          |
| producao_consumo_importacao_exportacao_uvas_vinhos_global |
+-----+
6 rows in set (0.03 sec)
```

Pesquisa MySQL

```
mysql> select *  
-> from producao_consumo_importacao_exportacao_uvas_vinhos_global  
-> where Product = 'wine' and Variable = 'Consumption' and Year='2020';
```

index	Continent	Region/Country	Product	Variable	Year	Unit	Quantity
34451	Asia	Afghanistan	Wine	Consumption	2020	1000 hl	1
34557	Europe	Albania	Wine	Consumption	2020	1000 hl	205
34666	Africa	Algeria	Wine	Consumption	2020	1000 hl	455
34724	Oceania	American Samoa	Wine	Consumption	2020	1000 hl	0
34803	Africa	Angola	Wine	Consumption	2020	1000 hl	317
34884	America	Antigua and Barbuda	Wine	Consumption	2020	1000 hl	15
34990	America	Argentina	Wine	Consumption	2020	1000 hl	9430
35099	Asia	Armenia	Wine	Consumption	2020	1000 hl	105
35261	Oceania	Australia	Wine	Consumption	2020	1000 hl	5950
35370	Europe	Austria	Wine	Consumption	2020	1000 hl	2306
35479	Asia	Azerbaijan	Wine	Consumption	2020	1000 hl	108
35562	America	Bahamas	Wine	Consumption	2020	1000 hl	37
35643	Asia	Bahrain	Wine	Consumption	2020	1000 hl	15
35699	Asia	Bangladesh	Wine	Consumption	2020	1000 hl	1
35778	America	Barbados	Wine	Consumption	2020	1000 hl	26
35884	Europe	Belarus	Wine	Consumption	2020	1000 hl	982

Tratamentos com Pandas

Removendo colunas irrelevantes

```
[ ] 1 df1.drop(['id', 'type_id', 'vintage_type', 'region_id', 'region_country_name', 'region_country_most_used_grapes_0_has_detailed_info'\n2         'winery_id', 'taste_flavor_0_stats_mentions_count', 'taste_flavor_0_primary_keywords_0_id', 'taste_flavor_0_primary_keywords_0_count'\n3         'taste_flavor_0_primary_keywords_1_id', 'taste_flavor_0_primary_keywords_2_count', 'taste_flavor_0_stats_count'], axis=1, inplace=True)
```

Criando um dicionário para renomear as colunas e executando a função para renomear as colunas

```
[ ] 1 dict_rename_df1 = {'name': 'nome', 'seo_name': 'seo', 'is_natural': 'natural'\n2         'region_name': 'nome_regiao', 'region_country_most_used_grapes_0_name': 'uva_mais_utilizada'\n3         'winery_name': 'vinicula', 'taste_flavor_0_group': 'grupo_de_sabores', 'taste_flavor_0_stats_score': 'pontuacao_sabores'\n4         'taste_flavor_0_primary_keywords_0_name': 'sabor_1', 'taste_flavor_0_primary_keywords_1_name': 'sabor_2'\n5         'taste_flavor_0_primary_keywords_2_name': 'sabor_3'}
```

Traduzindo os valores da coluna continente

```
[ ] 1 df2['continente'] = df2['continente'].replace({'Asia': 'Asia', 'Europe': 'Europa', 'Africa': 'Africa', 'America': 'Amer
```

Tratamentos com Pandas

Traduzindo os valores da coluna regioao_pais

```
1 df2['regiao_pais'] = df2['regiao_pais'].replace(  
2     {'Afghanistan': 'Afeganistão', 'Algeria': 'Argélia', 'Antigua and Barbuda': 'Antígua e Barbuda', 'Azerbaijan': 'Azerbaijão',  
3     'Belarus': 'Bielorrússia', 'Belgium': 'Bélgica', 'Belgium and Luxemburg': 'Bélgica e Luxemburgo', 'Bhutan': 'Butão',  
4     'Bosnia and Herzegovina': 'Bósnia e Herzegovina', 'Botswana': 'Botsuana', 'Brazil': 'Brasil', 'Brunei Darussalam': 'Brunei',  
5     'Cambodia': 'Camboja', 'Cameroon': 'Camarões', 'Cayman Islands': 'Ilhas Cayman',  
6     'Central African Republic': 'República Centro-Africana', 'Chad': 'Chade', 'Comoros': 'Comores', 'Cook Islands': 'Ilhas Cook',  
7     'Côte d'Ivoire': 'Costa do Marfim', 'Croatia': 'Croácia', 'Cyprus': 'Chipre', 'Czech Republic': 'Tchéquia',  
8     'Denmark': 'Dinamarca', 'Dominican Republic': 'República Dominicana', 'Ecuador': 'Equador', 'Egypt': 'Egito',  
9     'Eswatini': 'Suazilândia', 'Ethiopia': 'Etiópia', 'Faroe Islands': 'Ilhas Faroe', 'Finland': 'Finlândia', 'France': 'França',  
10    'French Guiana': 'Guiana Francesa', 'French Polynesia': 'Polinesia Francesa', 'Gabon': 'Gabão', 'Germany': 'Alemanha',  
11    'Ghana': 'Gana', 'Greece': 'Grécia', 'Greenland': 'Groelândia', 'Grenada': 'Granada', 'Guadeloupe': 'Guadalupe', 'Guinea': 'Guiné',  
12    'Guinea-Bissau': 'Guiné-Bissau', 'Guyana': 'Guiana', 'Hungary': 'Hungria', 'Iceland': 'Islândia', 'Iran': 'Irã', 'Iraq': 'Iraque',  
13    'Ireland': 'Irlanda', 'Italy': 'Itália', 'Japan': 'Japão', 'Jordan': 'Jordânia', 'Kazakhstan': 'Cazaquistão', 'Kenya': 'Quênia',  
14    'Kiribati': 'Quiribati', 'Korea (the Republic of)': 'Coreia do Sul', 'Kyrgyzstan': 'Quirguistão',  
15    'Lao People's Democratic Republic': 'Laos', 'Latvia': 'Letônia', 'Lebanon': 'Libano', 'Lesotho': 'Lesoto', 'Libya': 'Libia',  
16    'Lithuania': 'Lituânia', 'Luxemburg': 'Luxemburgo', 'Macao': 'Macau', 'Malaysia': 'Malásia', 'Maldives': 'Maldivas',  
17    'Martinique': 'Martinica', 'Mauritius': 'Ilhas Maurício', 'Moldova': 'Moldávia', 'Morocco': 'Marrocos', 'Mozambique': 'Moçambique',  
18    'Netherlands': 'Países Baixos', 'Netherlands Antilles': 'Antilhas Neerlandesas', 'New Caledonia': 'Nova Caledônia',  
19    'New Zealand': 'Nova Zelândia', 'Niger': 'Nigéria', 'Norway': 'Noruega', 'Oman': 'Omã', 'Pakistan': 'Paquistão',  
20    'Papua New Guinea': 'Papua-Nova Guiné', 'Paraguay': 'Paraguai', 'Philippines': 'Filipinas', 'Poland': 'Polônia',  
21    'Qatar': 'Catar', 'Republic of North Macedonia': 'Macedônia do Norte', 'Republic of Türkiye': 'Turquia',
```

Tratamentos com Pandas

▼ Analisando o conteúdo da coluna produto

```
[ ] 1 pd.unique(df2['produto'])
```

```
array(['Dried Grapes', 'Table Grapes', 'Fresh Grapes', 'Vineyard', 'Wine'],  
      dtype=object)
```

▼ Traduzindo os valores da coluna produto

```
▶ 1 df2['produto'] = df2['produto'].replace({  
  2     'Dried Grapes': 'Uvas Secas', 'Table Grapes': 'Uvas de Mesa',  
  3     'Fresh Grapes': 'Uvas Frescas', 'Vineyard': 'Vinhas',  
  4     'Wine': 'Vinho'  
  5 })
```

Tratamentos com Pandas

Visualização inicial do df4



```
1 df4.head()
```



	Year	Imports amounta	Exports amounta	Imports-exports amounta	Imports valueb	Exports valueb	Import-exports valueb
0	1990	8.0	3.4	4.6	15.3	3.6	11.7
1	1991	8.0	4.3	3.7	14.1	4.2	9.9

```
] 1 df5.head()
```

	Year	RS total (in tons)	% of Brazilian production	PE total (in tons)	% of Brazilian production.1	Brazil total production
0	1990	538,705	66.94	14,483	1.80	804,774
1	1991	396,318	61.16	17,163	2.65	648,026

Tratamentos com Pandas

Fazendo o merge dos dfs 4 e 5

```
[ ] 1 merged_df = pd.merge(df4, df5, on='Year')
```

Analizando o novo dataframe

```
1 merged_df.head()
```



	Year	Imports amounta	Exports amounta	Imports- exports amounta	Imports valueb	Exports valueb	Import- exports valueb	RS total (in tons)	% of Brazilian production	PE total (in tons)	% of Brazilian production.1	Brazil total production
0	1990	8.0	3.4	4.6	15.3	3.6	11.7	538,705	66.94	14,483	1.80	804,774
1	1991	8.0	4.3	3.7	14.1	4.2	9.9	396,318	61.16	17,163	2.65	648,026
2	1992	6.0	7.4	-1.4	10.2	7.5	2.7	505,462	63.17	18,510	2.31	800,112
3	1993	12.0	20.2	-8.2	18.8	14.8	4.0	489,464	62.16	26,475	3.36	787,363
4	1994	21.4	14.8	6.6	35.0	12.7	22.3	479,034	59.32	30,821	3.82	807,520

Tratamentos com Pandas

+ Código

+ Texto

```
[ ] 1 df6.head()
```

		1990	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	:
0	Total wine production	200.00	209.00	224.00	255.00	274.00	200.00	228.0	198.00	260.0	319.0	329.00	263.00	291.00	232.00	357.0	271.00	217.00	318.00	335.00	
1	% of Brazilian total wine production	54.12	82.65	74.52	83.42	78.41	56.55	87.7	60.16	96.7	84.7	90.49	88.61	90.66	88.55	90.9	84.84	91.53	90.89	90.88	

Tratamentos com Pandas

Invertendo as colunas pelas linhas no df6 e checagem da operação

```
1 df6_transposed = df6.T  
2 df6_transposed.head()
```

	0	1	2
	Total wine production	% of Brazilian total wine production	Brazilian total wine production
1990	200.0	54.12	370.0
1991	209.0	82.65	253.0
1992	224.0	74.52	301.0
1993	255.0	83.42	306.0

Tratamentos com Pandas

Convertendo o tipo da coluna "Year" de string para int para que a junção com o df final possa ser bem sucedida

```
[ ] 1 df6_transposed['index'] = df6_transposed['index'].astype(str).astype(int)
```

```
▶ 1 df6_transposed.rename(columns={'index':'Year'}, inplace=True)
```

— Código — + Texto

```
[▶] 1 df6_transposed.head()
```

	Year	Total wine production	% of Brazilian total wine production	Brazilian total wine production
0	1990	200.0	54.12	370.0
1	1991	209.0	82.65	253.0
2	1992	224.0	74.52	301.0
3	1993	255.0	83.42	306.0
4	1994	274.0	78.41	349.0

Tratamentos com Pandas

Fazendo o merge do df6 com o df novo(junção dos dfs 4 e 5)

```
[ ] 1 merged_df = pd.merge(merged_df, df6_transposed, on='Year')
```

```
1 merged_df.head()
```

	Year	Imports amounta	Exports amounta	Imports- exports amounta	Imports valueb	Exports valueb	Import- exports valueb	RS total (in tons)	% of Brazilian production	PE total (in tons)	% of Brazilian production.1	Brazil total production	Total wine production	% of Brazilian total wine production	Brazilian total wine production
0	1990	8.0	3.4	4.6	15.3	3.6	11.7	538,705	66.94	14,483	1.80	804,774	200.0	54.12	370.0
1	1991	8.0	4.3	3.7	14.1	4.2	9.9	396,318	61.16	17,163	2.65	648,026	209.0	82.65	253.0
2	1992	6.0	7.4	-1.4	10.2	7.5	2.7	505,462	63.17	18,510	2.31	800,112	224.0	74.52	301.0
3	1993	12.0	20.2	-8.2	18.8	14.8	4.0	489,464	62.16	26,475	3.36	787,363	255.0	83.42	306.0
4	1994	21.4	14.8	6.6	35.0	12.7	22.3	479,034	59.32	30,821	3.82	807,520	274.0	78.41	349.0

Tratamentos com Pandas

Criando um dicionário para traduzir as colunas e executando a função para traduzir as colunas

```
[ ] 1 dict_rename_merged = {'Year': 'ano', 'Imports amounta': 'qtd_litros_importacao(milhoes)', 'Exports amounta': 'qtd_litros_exportacao(milhoes)'\n
2      'Imports valueb': 'valor_em_dolar_importacao(milhoes)', 'Exports valueb': 'valor_em_dolar_exportacao(milhoes)'\n
3      'RS total (in tons)': 'Producao_uvas_RS(toneladas)', '% of Brazilian production': '%_Producao_uvas_RS'\n
4      'PE total (in tons)': 'Producao_uvas_PE(toneladas)', '% of Brazilian production.1': '%_Producao_uvas_PE'\n
5      'Brazil total production': 'Producao_de_uvas_brasil', 'Total wine production': 'RS_producao_vinhos_litros(milhoes)'\n
6      '% of Brazilian total wine production': '%_Producao_de_vinhos_brasileiros'\n
7      'Brazilian total wine production': 'Producao_vinhos_brasil_litros(milhoes)'}

[ ] 1 merged_df.rename(columns=dict_rename_merged, inplace=True)
```

Tratamentos com Pandas

Verificando se as colunas foram traduzidas

```
[21] 1 merged_df.head()
```

	ano	qtd_litros_importacao(milhoes)	qtd_litros_exportacao(milhoes)	valor_em_dolar_importacao(milhoes)	valor_em_dolar_exportacao(milhoes)	Producao_uvas_RS(toneladas)
0	1990	8.0	3.4	15.3	3.6	538,705
1	1991	8.0	4.3	14.1	4.2	396,318
2	1992	6.0	7.4	10.2	7.5	505,462
3	1993	12.0	20.2	18.8	14.8	489,464
4	1994	21.4	14.8	35.0	12.7	479,034

Tratamentos com PySpark

Traduzindo os valores de inglês para português da coluna grupo_de_sabores

```
[ ] 1 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'non_oak', 'sem_barrica'))
    2 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'citrus_fruit', 'frutas_citricas'))
    3 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'earth', 'terra'))
    4 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'tree_fruit', 'arvore_frutifera'))
    5 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'oak', 'barricado'))
    6 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'red_fruit', 'frutas_vermelhas'))
    7 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'tropical_fruit', 'frutas_tropicais'))
    8 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'microbio', 'microbio'))
    9 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'black_fruit', 'frutas_negras'))
   10 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'spices', 'especiarias'))
   11 sparkDF1 = sparkDF1.withColumn('grupo_de_sabores', F.regexp_replace('grupo_de_sabores', 'dried_fruit', 'frutas_secas'))
```

Tratamentos com PySpark

Alteração do tipo da coluna "natural" de Boolean para String para realizarmos a renomeação dos valores

```
[ ] 1 sparkDF1.withColumn("natural", col("natural").cast(StringType()))
```

Verificando se há valores únicos dentro da coluna natural

```
[ ] 1 sparkDF1.select('natural').distinct().show()
```

```
+-----+  
|natural|  
+-----+  
|   true|  
|  false|  
+-----+
```

Alteração dos valores da coluna natural

```
[ ] 1 sparkDF1 = sparkDF1.withColumn('natural', F.regexp_replace('natural', 'false', 'Não'))  
    2 sparkDF1 = sparkDF1.withColumn('natural', F.regexp_replace('natural', 'true', 'Sim'))
```


Tratamentos com PySpark

Verificando se há valores únicos dentro da coluna "grupo_de_sabores"

```
1 sparkDF1.select('grupo_de_sabores').distinct().show(100, truncate=False)
```

```
+-----+  
|grupo_de_sabores|  
+-----+  
|non_oak          |  
|citrus_fruit     |  
|earth            |  
|oak              |  
|tree_fruit       |  
|red_fruit        |  
|NaN              |  
|tropical_fruit   |  
|microbio         |  
|black_fruit      |  
|spices           |  
|floral           |  
|vegetal          |  
|dried_fruit      |  
+-----+
```

Tratamentos com PySpark

Removendo vírgulas do dataframe

```
[ ] 1 sparkDF_merged = sparkDF_merged.withColumn("Producao_uvas_RS(toneladas)", F.regexp_replace('Producao_uvas_RS(toneladas)', ',', ''))
```

```
[ ] 1 sparkDF_merged = sparkDF_merged.withColumn("Producao_uvas_PE(toneladas)", F.regexp_replace('Producao_uvas_PE(toneladas)', ',', ''))
```

```
[ ] 1 sparkDF_merged = sparkDF_merged.withColumn("Producao_de_uvas_brasil", F.regexp_replace('Producao_de_uvas_brasil', ',', ''))
```

Pipeline

Criação de um modelo para a GCP

```
1 pipeline_options = {
2     'project': 'projeto-final-373619',
3     'runner': 'DataFlowRunner',
4     'region': 'us-east5',
5     'staging_location': 'gs://datasets-tvlle/staging/',
6     'temp_location': 'gs://datasets-tvlle/temp/',
7     'template_location': 'gs://datasets-tvlle/models/modelo_batch'
8 }
9
10 serviceAccount = '/content/projeto-final-373619-f6460a3b65bd.json'
11 os.environ['GOOGLE_APPLICATION_CREDENTIALS'] = serviceAccount
12
13
14 pipeline_options = PipelineOptions.from_dictionary(pipeline_options)
15 p1 = beam.Pipeline(options=pipeline_options)
16
17 vinho = (
18     p1
19     | 'Extrair do CSV' >> beam.io.ReadFromText('gs://datasets-tvlle/tratados/producao_consumo_importacao_exportacao_uvas_vinhos_trat.csv', skip_header_lines=1)
20     | 'Separador de dados' >> beam.Map(lambda record: record.split(','))
21     | 'Filtro vinho' >> beam.Filter(lambda record: record[3] == 'Vinho')
22     | 'Load final' >> beam.io.WriteToText('gs://datasets-tvlle/tratados/vinhotr', file_name_suffix='.csv')
23 )
24
25 p1.run()
```

DATABASES: 1 COLLECTIONS: 4

+ Create Database

Q Search Namespaces

db-MercadoBebidas

avaliacoes_vinhos_gl...

producao_consumo_i...

producao_vinhos_br_tr...

vinhos_brasil_tratado

db-MercadoBebidas.avaliacoes_vinhos_global_trat

STORAGE SIZE: 15.16MB LOGICAL DATA SIZE: 33.65MB TOTAL DOCUMENTS: 150930 INDEXES TOTAL SIZE: 5.09MB

Find Indexes Schema Anti-Patterns 0 Aggregation Search Indexes ●

FILTER { field: 'value' }

QUERY RESULTS: 1-20 OF MANY

_id: ObjectId('63bd805d5ccf64cd4a836f69')
index: 0
pais: "Estados Unidos"
designacao: "Martha's Vineyard"
pontos: 96
preco: 235
provincia: "California"
regiao1: "Napa Valley"
regiao2: "Napa"
variedade: "Cabernet Sauvignon"
adeiga: "Heitz"

_id: ObjectId('63bd805d5ccf64cd4a836f6e')
index: 5

Datasets finais

Avaliações de vinhos no mercado mundial;

Consumo, produção, importação e exportação de uvas e vinhos no mercado mundial;

Produção de vinhos no Brasil;

Avaliações de vinhos no mercado brasileiro;

Datasets tratados



Google Cloud Storage

 [avaliacoes_vinhos_global_trat.csv](#)

 [producao_consumo_importacao_expor](#)

 [producao_vinhos_brasil_trat.csv](#)

 [vinhos_brasil_tratado.csv](#)

Armazenamento

Storage da GCP (Bucket)

MongoDB

BigQuery

Google Cloud



 mongoDB

BigQuery & LookerStudio

OBJETIVO

- Representação dos dados

MOTIVO

- Otimização

Conclusão



Viabilidade, competitivo

Alta demanda do
mercado



Grande mercado
nacional e mundial



Crescimento

Investimento

Orçamento

1 – 17 de janeiro de 2023 (custo total) ?

R\$ 115,12

inclui R\$ 0,00 de crédito



R\$ 115,12 durante 15 – 31 de dezembro de 2022

1 – 31 de janeiro de 2023 (custo total previsto) ?

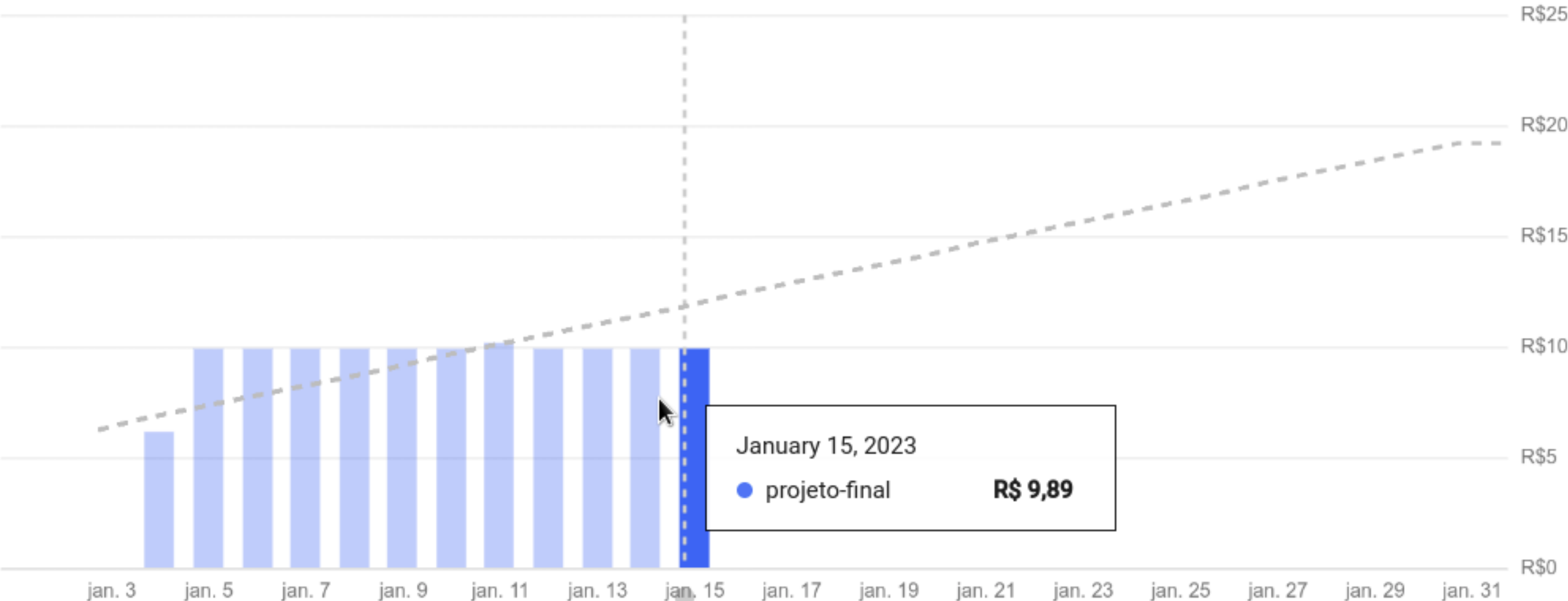
R\$ 369,89

inclui R\$ 0,00 em créditos



R\$ 369,89 durante 1 – 31 de dezembro de 2022

Diariamente ▼



orçamento vigente até a data de entrega do projeto (17/01/2023)

<input type="checkbox"/>	Nome do orçamento ↑	Período do orçamento	Tipo de orçamento	Aplicável a	Acionar alertas a	Valor dos gastos e do orçamento
<input type="checkbox"/>	<u>custos-projeto-final</u>	Mensal	Valor especificado	Projeto "p" ▼	10%, 30%, 70...	<div><div></div></div> R\$ 115,12/R\$ 250,00 Não inclui -R\$ 115,12 de crédito

TVLLE
CONSULTORIA
ORIENTADA POR DADOS



www.linkedin.com/in/vitor-brandao9506/



www.linkedin.com/in/emanuela-caetano/



www.linkedin.com/in/leonardo-fabricio-48bb60157/



www.linkedin.com/in/lucasalcantara/



www.linkedin.com/in/talytta/