Pandas e Esteganografia

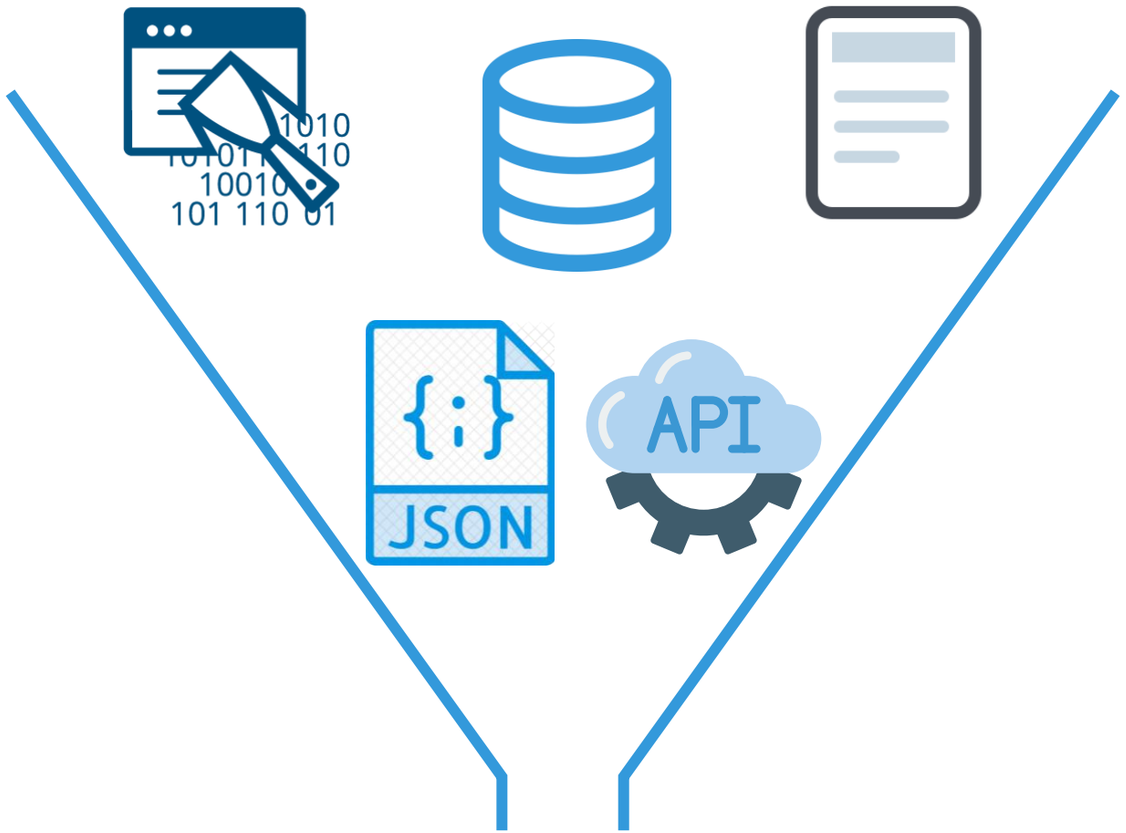


Imagem: https://clearbit.com/our-data

Professor: Alex Pereira

Transformação de Dados: Estudo de Caso do Autodiagnóstico da SGD

A	B	C	D	E	F	G
Qual das opç	Por favor, infc	3.1. Dados e	3.1.1.1. Relev	3.1.1.2. Prontidão Orgar	3.1.1.3. Recu	3.1.1.4. Segn
2.Área de TI (	Universidade Federal do Pa	2. INICIADO:	3. EMERGENTE: A Inst	3. EMERGEN	3. EMERGEN
2.Área de TI (	Universidade Federal de Sá	3. EMERGEN	3. EMERGENTE: A Inst	3. EMERGEN	3. EMERGEN
2.Área de TI (	Fundação Universidade Fe	3. EMERGEN	4. DESENVOLVIDO: A I	2. INICIADO:	4. DESENVC
3.Área de TI (	Centro Federal de Educaçá	1. NÃO INICIA	1. NÃO INICIADO: Parte	1. NÃO INICIA	1. NÃO INICIA
3.Área de TI (	Instituto Federal de Educaç	1. NÃO INICIA	1. NÃO INICIADO: Parte	2. INICIADO:	3. EMERGEN
2.Área de TI (	Fundação Universidade Fe	3. EMERGEN	2. INICIADO: A Instituiçá	1. NÃO INICIA	1. NÃO INICIA
1.Área de TI f	Instituto Nacional de Metro	3. EMERGEN	1. NÃO INICIADO: Parte	2. INICIADO:	3. EMERGEN
2.Área de TI (	Universidade Federal do Ri	4. DESENVC	3. EMERGENTE: A Inst	4. DESENVC	4. DESENVC
1.Área de TI f	Comissão Nacional de Ene	1. NÃO INICIA	2. INICIADO: A Instituiçá	2. INICIADO:	1. NÃO INICIA
1.Área de TI f	Fundação Nacional dos Po	1. NÃO INICIA	1. NÃO INICIADO: Parte	1. NÃO INICIA	1. NÃO INICIA
1.Área de TI f	Controladoria-Geral da Uniá	4. DESENVC	3. EMERGENTE: A Inst	4. DESENVC	2. INICIADO:
1.Área de TI f	Secretaria do Tesouro Nac	5. OTIMIZADO	3. EMERGENTE: A Inst	5. OTIMIZADO	1. NÃO INICIA
1.Área de TI f	Fundação Casa de Rui Bar	3. EMERGEN	3. EMERGENTE: A Inst	1. NÃO INICIA	1. NÃO INICIA
1.Área de TI f	Empresa Brasileira de Infra	4. DESENVC	3. EMERGENTE: A Inst	4. DESENVC	4. DESENVC

Principais Alterações

- Extrair o valor numérico das categorias
  - “1. NÃO INICIADO”, “2. INICIADO”, “3. EMERGENTE”, “4. DESENVOLVIDO”, “5. OTIMIZADO”
- Despivotar a tabela

area	orgao	Pergunta	Valor	ria_matur	iado_matu	d_pergun	ciado_per	ano
2.Área de	Universic	3.1.1.1. Re	2	INICIADO	Na Institu	3.1.1.1.	Relevânci	2023
2.Área de	Universic	3.1.1.1. Re	3	EMERGEN	Na Institu	3.1.1.1.	Relevânci	2023
2.Área de	Fundaçã	3.1.1.1. Re	3	EMERGEN	Na Institu	3.1.1.1.	Relevânci	2023
3.Área de	Centro Fe	3.1.1.1. Re	1	NAO INICI	Na Institu	3.1.1.1.	Relevânci	2023
3.Área de	Instituto	3.1.1.1. Re	1	NAO INICI	Na Institu	3.1.1.1.	Relevânci	2023
2.Área de	Fundaçã	3.1.1.1. Re	3	EMERGEN	Na Institu	3.1.1.1.	Relevânci	2023
1.Área de	Instituto	3.1.1.1. Re	3	EMERGEN	Na Institu	3.1.1.1.	Relevânci	2023
2.Área de	Universic	3.1.1.1. Re	4	DESENVOI	A Instituiç	3.1.1.1.	Relevânci	2023

Prompt como Compartilhamento de Conhecimento e Refactoring do Prompt

Processar arquivo 2024 ("/content/Autodiagnóstico 2024 Dados_Tratado_GD.xlsx"):

1. Renomear colunas especificadas por "3:" (ou seja df.columns[3:]) removendo o padrão r'(\d\.?)+\s?' do início.
2. Manter apenas o número (1 a 5) no início do texto das colunas especificadas por "3:" (ou seja df.columns[3:]), removendo o restante do texto.
3. Pivote as colunas especificadas por "3:" (ou seja df.columns[3:]) para uma coluna chamada Valor, criando o dataframe df_melted_2024.
4. Remover o padrão r'(\d\.?)+\s?' das colunas area e Pergunta.
5. Remover registros com valores nulos na coluna Valor.
6. Adicionar a coluna ano com o valor 2024.
7. Realizar merge com o arquivo /content/drive/MyDrive/empreender/ME/GovBr/Autodiagnostico/MapeamentoEixos.xlsx (contendo perguntas e eixos) e verificar se o resultado do inner join é igual ao outer join.

Prompt como Compartilhamento de Conhecimento e Refactoring do Prompt

Processar arquivo 2023 (/content/Resposta_40133199_results_survey998556.xlsx):

1. Renomear colunas especificadas por "3:" (ou seja df.columns[3:]) removendo o padrão r'(\d\.?)+\s?' do início.
2. Manter apenas o número (1 a 5) no início do texto das colunas especificadas por "3:" (ou seja df.columns[3:]), removendo o restante do texto.
3. Pivote as colunas especificadas por "3:" (ou seja df.columns[3:]) para uma coluna chamada Valor, criando o dataframe df_melted_2023.
4. Remover o padrão r'(\d\.?)+\s?' das colunas area e Pergunta.
5. Remover registros com valores nulos na coluna Valor.
6. Adicionar a coluna ano com o valor 2023.
7. Realizar merge com o arquivo /content/drive/MyDrive/empreender/ME/GovBr/Autodiagnostico/MapeamentoEixos.xlsx e verificar se alguma pergunta ficou sem eixo.

Finalizar:

1. Concatenar verticalmente df_melted_2023 e df_melted_2024.

Operação Join (Algebra Relacional)

Médico

CPF	Nome	Salario
11222731642	Jose Pereira	10
91498733332	Maria da Silva	20
81464221612	Pedro Martins	15

Professor

CPF	Nome	Salario
11222731642	Jose Pereira	6
91498733332	Maria da Silva	8
21564281600	Roberto Afonso	5

Join (ou inner join)

CPF	Nome	Salario_M	Salario_P
11222731642	Jose Pereira	10	6
91498733332	Maria da Silva	20	8

Operação Left Join (Algebra Relacional)

Médico

CPF	Nome	Salario
11222731642	Jose Pereira	10
91498733332	Maria da Silva	20
81464221612	Pedro Martins	15

Professor

CPF	Nome	Salario
11222731642	Jose Pereira	6
91498733332	Maria da Silva	8
21564281600	Roberto Afonso	5

Left Join

CPF	Nome	Salario_M	Salario_P
11222731642	Jose Pereira	10	6
91498733332	Maria da Silva	20	8
81464221612	Pedro Martins	15	--

Operação Right Join (Algebra Relacional)

Médico

CPF	Nome	Salario
11222731642	Jose Pereira	10
91498733332	Maria da Silva	20
81464221612	Pedro Martins	15

Professor

CPF	Nome	Salario
11222731642	Jose Pereira	6
91498733332	Maria da Silva	8
21564281600	Roberto Afonso	5

Right Join

CPF	Nome	Salario_M	Salario_P
11222731642	Jose Pereira	10	6
91498733332	Maria da Silva	20	8
21564281600	Roberto Afonso	--	5

Operação Outer Join (Algebra Relacional)

Médico

CPF	Nome	Salario
11222731642	Jose Pereira	10
91498733332	Maria da Silva	20
81464221612	Pedro Martins	15

Professor

CPF	Nome	Salario
11222731642	Jose Pereira	6
91498733332	Maria da Silva	8
21564281600	Roberto Afonso	5

Outer Join

CPF	Nome	Salario_M	Salario_P
11222731642	Jose Pereira	10	6
91498733332	Maria da Silva	20	8
21564281600	Roberto Afonso	--	5
81464221612	Pedro Martins	15	--

join (fundir/juntar)

- Faz o join de dois dataframes usando o índice
  - como chave de junção

```
In [70]: left2
Out[70]:
   Ohio  Nevada
a   1.0     2.0
c   3.0     4.0
e   5.0     6.0
In [71]: right2
Out[71]:
   Missouri  Alabama
b       7.0      8.0
c       9.0     10.0
d      11.0     12.0
e      13.0     14.0
```

```
In [73]: left2.join(right2, how='outer')
Out[73]:
   Ohio  Nevada  Missouri  Alabama
a   1.0     2.0       NaN      NaN
b   NaN     NaN       7.0      8.0
c   3.0     4.0       9.0     10.0
d   NaN     NaN      11.0     12.0
e   5.0     6.0      13.0     14.0
```

join (fundir/juntar)

- Com how='left' somente os registros do dataframe da esquerda
  - aparecem no resultado

```
In [70]: left2
Out[70]:
   Ohio  Nevada
a   1.0     2.0
c   3.0     4.0
e   5.0     6.0

In [71]: right2
Out[71]:
   Missouri  Alabama
b       7.0      8.0
c       9.0     10.0
d      11.0     12.0
e      13.0     14.0
```

```
left2.join(right2, how='left')
```

	Ohio	Nevada	Missouri	Alabama
a	1.0	2.0	NaN	NaN
c	3.0	4.0	9.0	10.0
e	5.0	6.0	13.0	14.0

merge (fundir/juntar)

- Semelhante ao join, mas você precisa informar a coluna de junção
  - pode ser inferida a partir do contexto da interseção entre as tabelas
    - ✓ Também pode ser especificada com o argumento **on** (Ex.: on='key')

```
In [37]: df1
Out[37]:
   data1 key
0      0   b
1      1   b
2      2   a
3      3   c
4      4   a
5      5   a
6      6   b
```

```
In [38]: df2
Out[38]:
   data2 key
0      0   a
1      1   b
2      2   d
```

```
In [39]: pd.merge(df1, df2)
Out[39]:
   data1 key  data2
0      0   b      1
1      1   b      1
2      6   b      1
3      2   a      0
4      4   a      0
5      5   a      0
```

Join vs Merge

- Ambos servem para combinar dataframes
- Join
  - Combina dataframes a partir dos seus indexes
    - ✓ Ou pode-se especificar uma coluna no dataframe onde se executa o método.
- Merge
  - Combina dataframes a partir de suas colunas
    - ✓ Pode validar o merge pelo tipo, com o argumento: validate
      - "1:1"
      - "1:m"
      - "m:1"
      - "m:m"

Maneiras de Armazenar vs Analisar os dados

Melhor para Armazenar

	Aluno	Disciplina	Objetiva	Discursiva
0	AlunoA	Portugues	8.5	6
1	AlunoA	Matematica	7.5	6.5
2	AlunoB	Geografia	9	7.5
3	AlunoB	HIstória	10	7

Melhor para Analisar

Disciplina	Geografia	HIstória	Matematica	Portugues
Aluno				
AlunoA	NaN	NaN	7.5	8.5
AlunoB	9	10	NaN	NaN

Reshaping / Pivoting (Pivotar)

- Método pivot
  - 3 argumentos: index, columns, values
    - ✓ `df.pivot(index='Aluno', columns='Disciplina', values='Objetiva')`
      - a função **`melt()`** faz a operação de despivotar

	Aluno	Disciplina	Objetiva	Discursiva
0	AlunoA	Portugues	8.5	6
1	AlunoA	Matematica	7.5	6.5
2	AlunoB	Geografia	9	7.5
3	AlunoB	HIstória	10	7

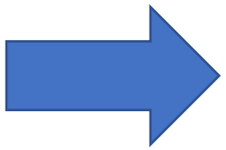
Pivotar

Disciplina	Geografia	HIstória	Matematica	Portugues
Aluno				
AlunoA	NaN	NaN	7.5	8.5
AlunoB	9	10	NaN	NaN

E quando houver valores repetidos ?

- Pivotar com o mesmo método pivot() gera exceção
  - Neste caso, use o método pivot_table
    - ✓ mean é a métrica padrão de cálculo sobre a de agregação

	Aluno	Disciplina	Objetiva	Discursiva
0	AlunoA	Portugues	8.5	6.0
1	AlunoA	Matematica	7.5	6.5
2	AlunoA	Geografia	9.0	7.5
3	AlunoA	Geografia	10.0	7.0
4	AlunoA	História	9.0	8.0
5	AlunoB	Portugues	8.5	8.5
6	AlunoB	Matematica	7.5	7.5
7	AlunoB	Geografia	9.0	9.0
8	AlunoB	História	10.0	10.0



Disciplina	Geografia	História	Matematica	Portugues
Aluno				
AlunoA	9.5	9.0	7.5	8.5
AlunoB	9.0	10.0	7.5	8.5

Reshaping / Pivoting com Índice Hierárquico

- Método stack/unstack (Pivotar com índice hierárquico)
  - stack = empilhar

state	number	
Ohio	one	0
	two	1
	three	2
Colorado	one	3
	two	4
	three	5

pivotar
unstack()

stack()
despivotar

	number	one	two	three
state				
Ohio		0	1	2
Colorado		3	4	5

ETL – Extract, Trasform and Load

- Extract (Extrair)
  - Coletar dados
- Transform (Transformar)
  - Transformar o dado num formato conveniente e útil
    - ✓ Limpeza / padronização
      - Sim/Não
      - YYYY-MM-DDThh:mm:ss.sTZ (ISO 8601)
    - ✓ Validação
    - ✓ Integração (join)
    - ✓ Reamostragem
- Load (Carregar/Inserir)
  - Inserir os dados numa base de dados de destino

Princípios de Gestão da Informação / Analogia com Princípios Orçamentários (PO)

- **Repositório de dados único**
  - Single Source Of Truth (SSOT)
    - ✓ PO: O orçamento deve ser uno.
- Dados Brutos/Originais
  - PO: Orçamento bruto.
    - ✓ Todas as parcelas da receita e da despesa devem aparecer no orçamento
      - em seus valores brutos, sem qualquer tipo de dedução.
- Utilidade
- Publicidade
  - Adequada
- Automação de Processos
- Controle Prévio

Analogia: https://www2.camara.leg.br/orcamento-da-uniao/cidadao/entenda/cursopo/principios

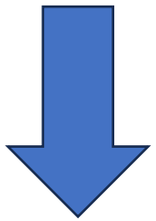
Cada tipo de problema tem uma ferramenta apropriada

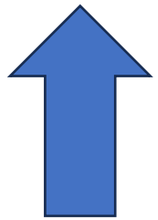
- "With great power, comes great responsibility"
  - Saber/conhecer (uma ferramenta)
    - ✓ não implica em dever usá-la
      - Coletar dados para experimento vs Coletar dados para um Dashboard
  - Simplicidade
    - ✓ a arte de maximizar a quantidade de trabalho não realizado--é essencial

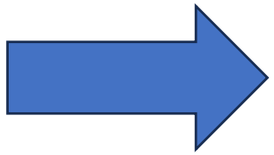
Apache
Airflow

Orquestrador de workflows e pipelines

Dashboard do Preço do Bitcoin

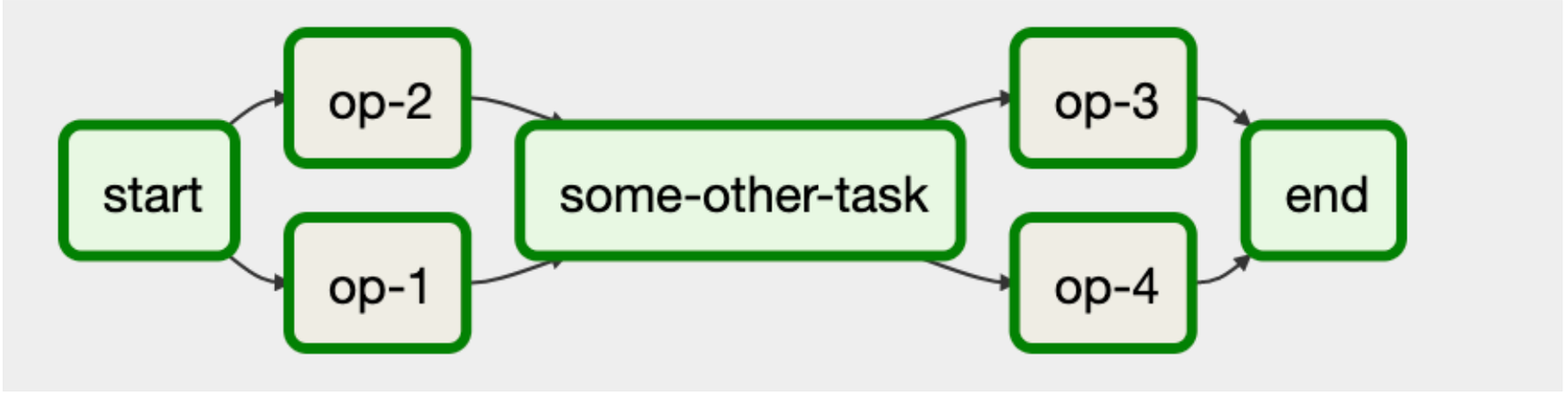






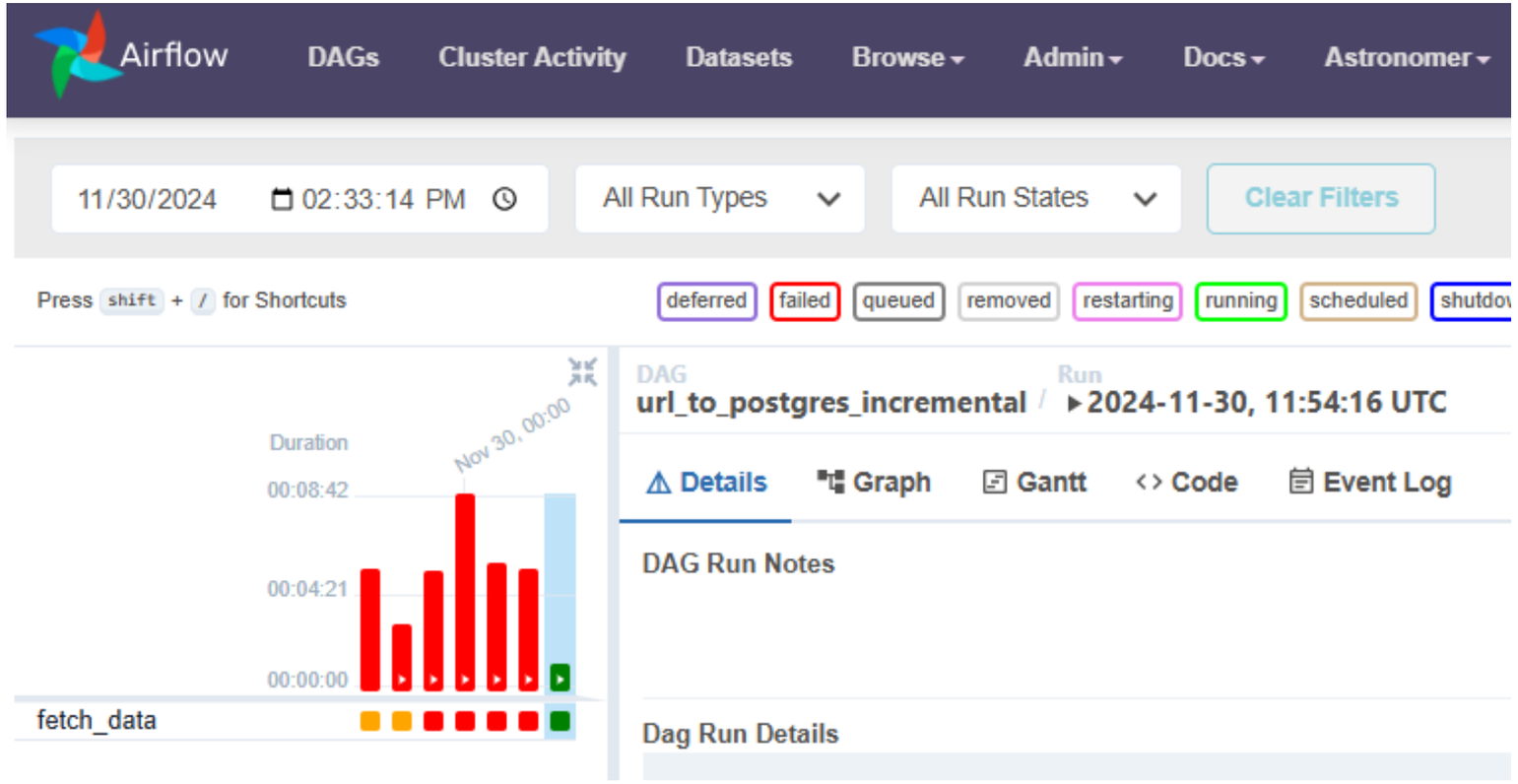
DAG (Directed Acyclic Graph)

- Grafo acíclico direto
  - Sem laços
    - ✓ Tarefas sequenciais e paralelizáveis
- Operators
  - Building blocks do Airflow
    - ✓ Contém a lógica / implementação dos requisitos; e
    - ✓ Templates prontos pra configurar e usar.



Task (Airflow)

- Uma instância de um Operator
  - Configuram parâmetros de contexto
    - ✓ Por exemplo, quando executar o Operator



Dag Run

Task

DAG simplificada

```
#### Extract
url = "https://github.com/alexlopespereira/dags/raw/refs/heads/main/data/pib_municipios.csv"
response = requests.get(url)
df = pd.read_csv(pd.io.common.StringIO(response.text), sep=';')

#### Transform
pivoted_data = df.melt(id_vars=["Cód.", "Município"],
    value_vars=["2007", "2009", "2011", "2013", "2015", "2017"],
    var_name="ano", value_name="populacao")

# Rename columns to align with the specified format
pivoted_data.rename(columns={"Cód.": "Codigo", "Município": "Municipio"}, inplace=True)

#### Load
pg_hook = PostgresHook(postgres_conn_id='postgres')
engine = pg_hook.get_sqlalchemy_engine()
pivoted_data.to_sql('pib_municipios', con=engine, if_exists='replace', index=False)
```

Dashboard do Preço do Bitcoin

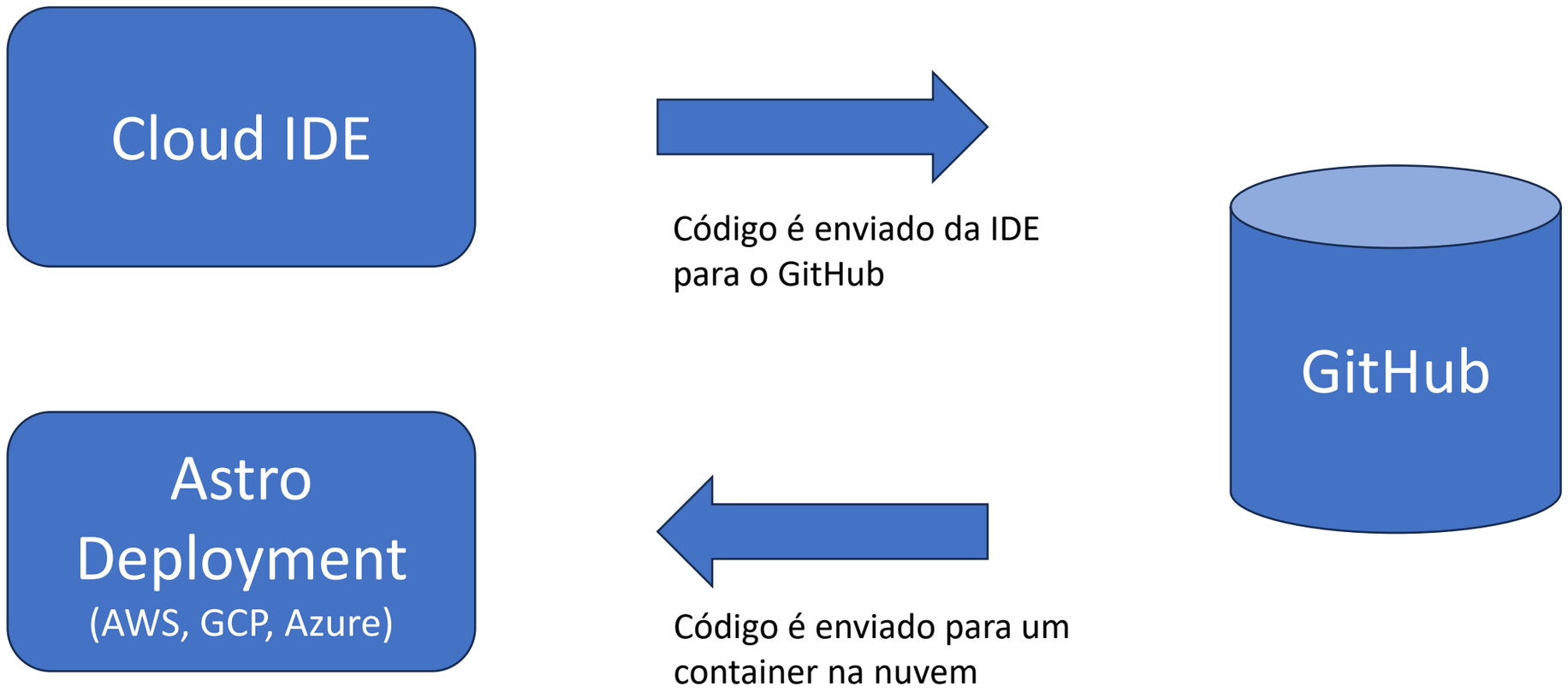
Airflow no Atronomer.io - Requisitos

- No astronomer.io
  - Workspace
  - Deployment
  - Projeto Cloud IDE
- No GitHub
  - Fork do repositório astro-dags-template
- Um banco de dados (ex. postgres na Digital Ocean)
- Acesso ao Google Looker Studio

astronomer.io

- Ferramenta online para gerenciar clusters do Airflow
- Trial de 14 dias
  - Ferramenta paga
    - ✓ Num contexto produtivo, o custo é efetivo

Deploy com Cloud IDE e GitHub



Astronomer.io e Airflow

- Tutorial sobre como https://youtu.be/sw-hATdQrBU
  - Configurar o Airflow no Astronomer;
  - Integrar com o github para fazer deploy automatizado;
  - Geração de código de DAGs
- Haverá um exercício valendo nota para
  - Sua DAG em código python;
  - Sua conta no Astronomer configurada;
  - Seu dashboard no Google Looker Studio

Prática no Colab Notebook

- Faça os exercícios da aula
  - A IA ainda não está boa para inferir os argumentos das funções do pandas que lêem arquivos e transformam num dataframe.
  - Você precisará, a priori, descobrir quais são esses argumentos e solicitar que a IA os utilize para ler os arquivos.