

GERAÇÃO DISTRIBUÍDA DE ENERGIA NO BRASIL

Projeto Final - SoulCode Academy

Aurélia Covre

João Victor Guimarães

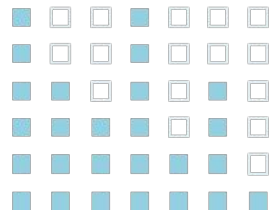
Kely Fernandes

Ricardo Rowedder

Robson Motta



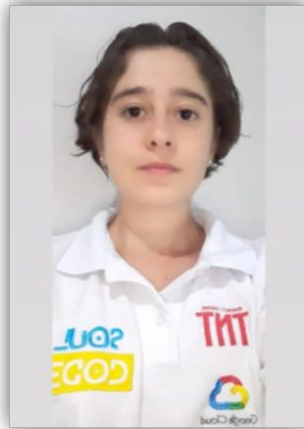
Equipe



Aurélia
Covre



João Victor
Guimarães



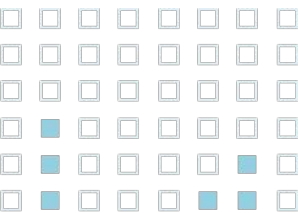
Kely
Fernandes



Ricardo
Rowedder

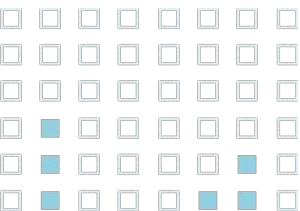
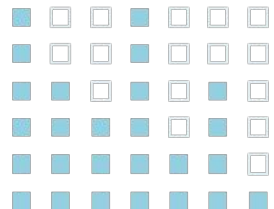


Robson
Motta

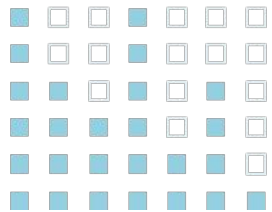


Conteúdo

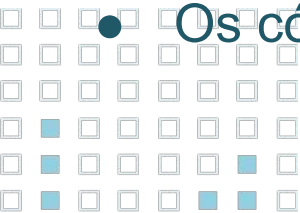
- Data Migration SQL to PySpark
- Geração Distribuída
- Etapas do processo
- Ferramentas
- Bancos de dados
- Tratamento dos dados
- Análises



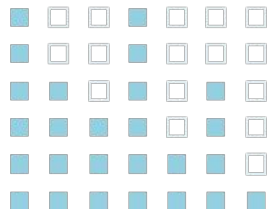
Data Migration SQL to PySpark



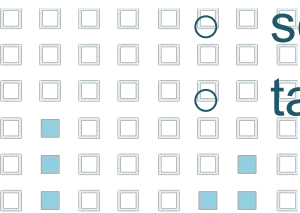
- O objetivo principal é criar uma réplica de um ambiente produtivo em uma estrutura de data lake em nuvem
- Criar um código em Python que leia o database, tratar e inserir os dados no data lake
- Utilizar o Spark para ler os dados e gerar resultados
- Os códigos em Python devem utilizar o paradigma orientado a objetos



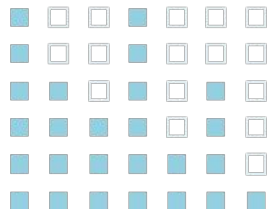
Datasets



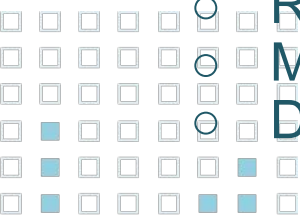
- Dados foram coletados de datasets públicos
- Quatro arquivos em formato CSV
 - dados.gov.br
 - aneel.gov.br
- Arquivos relacionados com geração distribuída
 - relação de empreendimentos
 - série histórica
 - tarifas



Geração Distribuída



- É o termo dado à energia elétrica gerada no local de consumo ou próximo a ele
- Sendo diversas fontes de energia renováveis
 - Energia solar
 - Energia eólica
 - Energia hídrica
- Seus benefícios estão entre
 - O baixo impacto ambiental
 - Redução no carregamento das redes de transmissão
 - Minimização das perdas
 - Diversificação da matriz energética



Tipos de geração

UFV
Central Geradora Solar
Fotovoltaica
7.818 MW - 97,6%



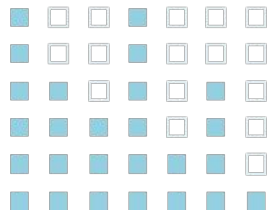
UTE
Usina Termelétrica
de Energia
113 MW - 1,41%

CGH
Central Geradora
Hidrelétrica
67 MW - 0,83%

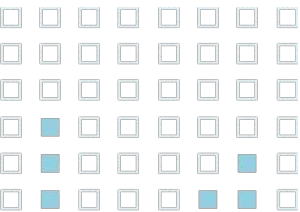


EOL
Central Geradora
Eólica
15 MW - 0,19%

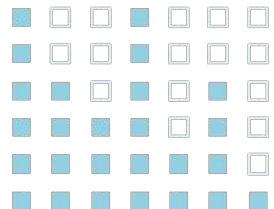
Microgeração X Minigeração



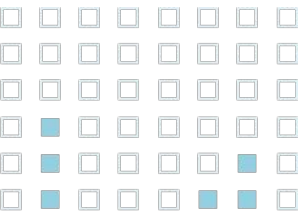
- Microgeração distribuída
 - Potência instalada menor ou igual a 75 kW
 - Custos extras são de responsabilidade da distribuidora de energia
- Minigeração distribuída
 - Potência instalada superior a 75 kW e menor ou igual a 5MW
 - Custos extras são de responsabilidade do consumidor
- Sistema de créditos de energia: 60 meses



Sistema de créditos



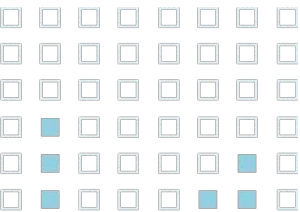
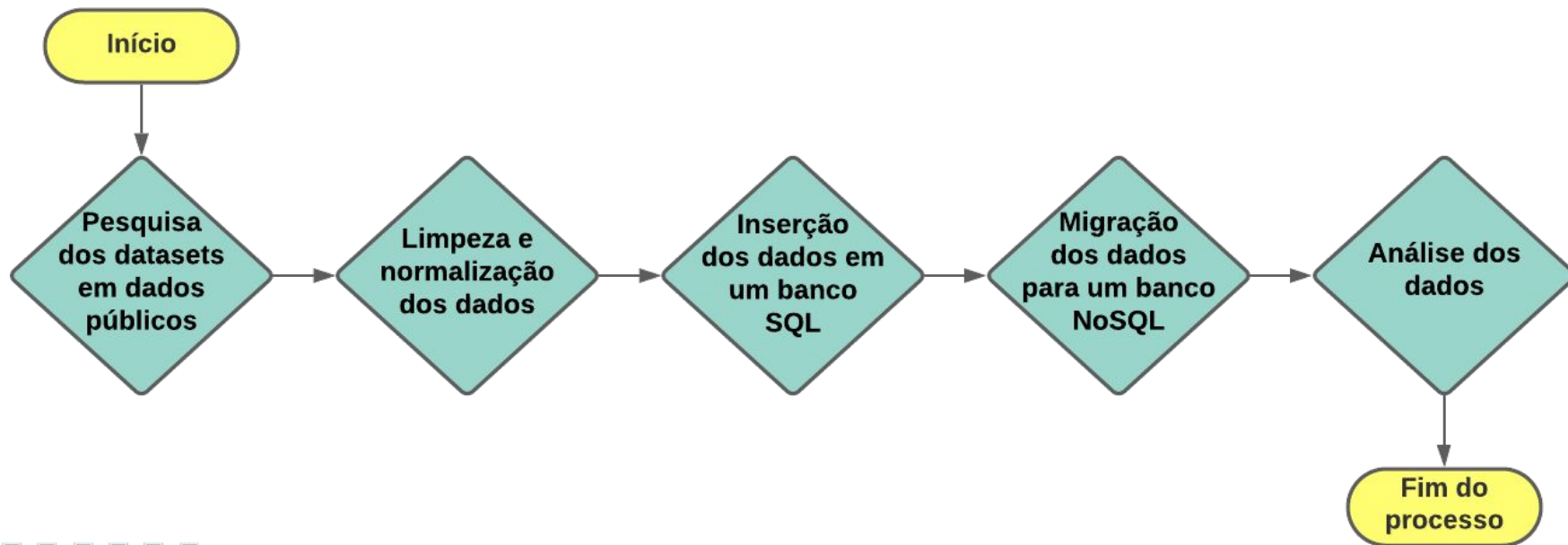
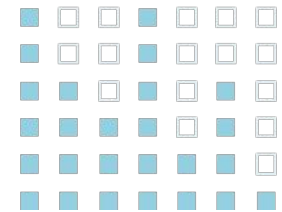
- Caso a energia injetada na rede seja superior à consumida, cria-se um “crédito de energia”
- Pode ser utilizado para abater o consumo
 - Própria unidade consumidora
 - Autoconsumo remoto
 - Geração em condomínio



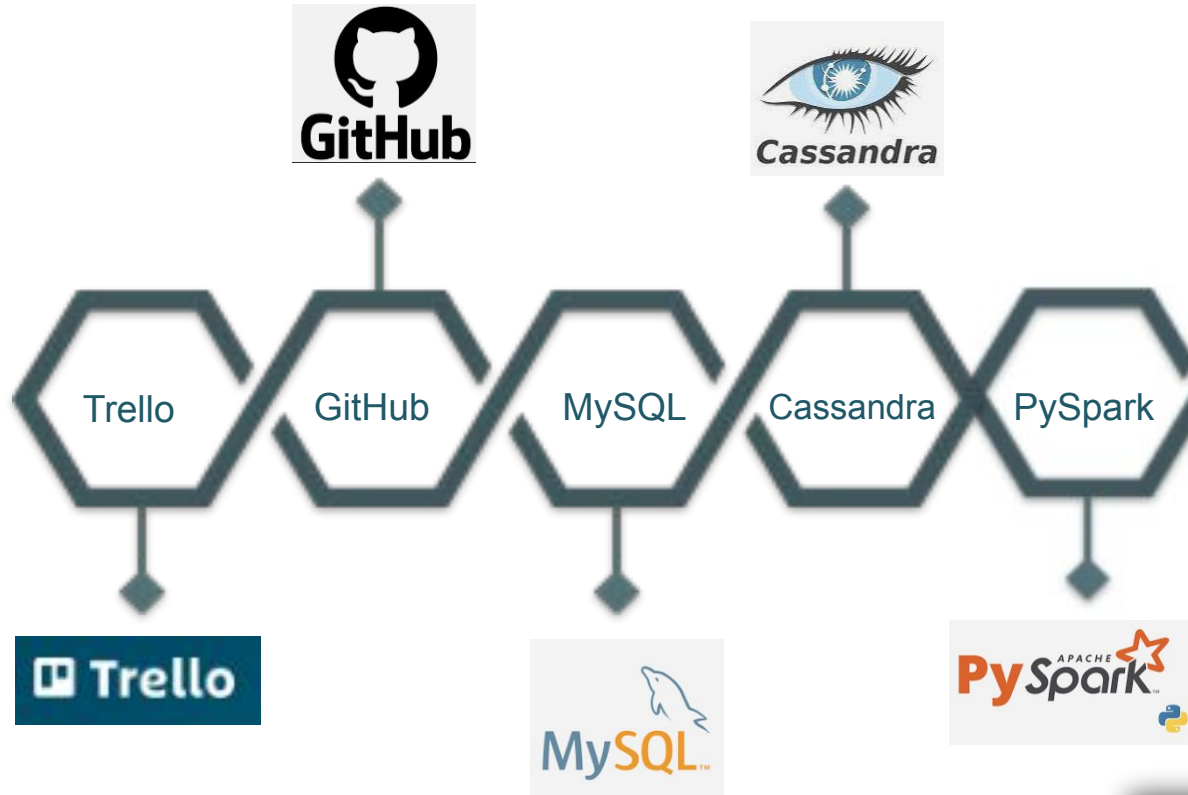
Sistema de créditos



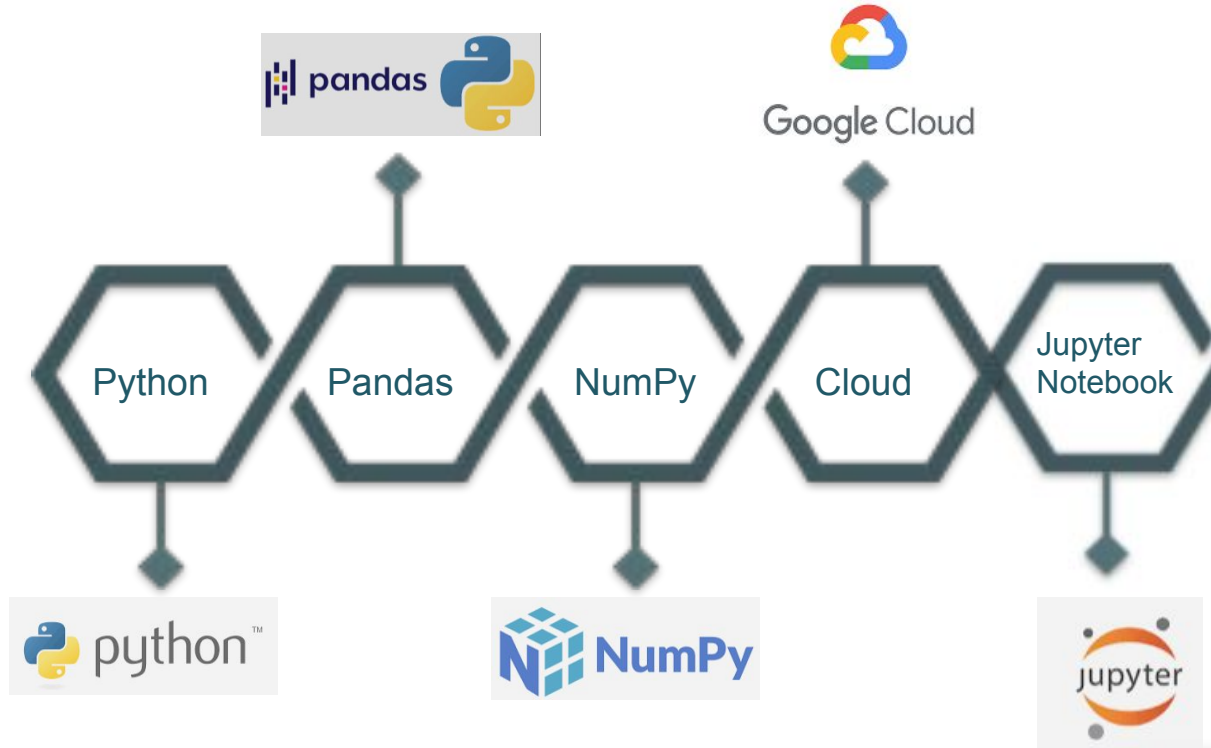
Etapas



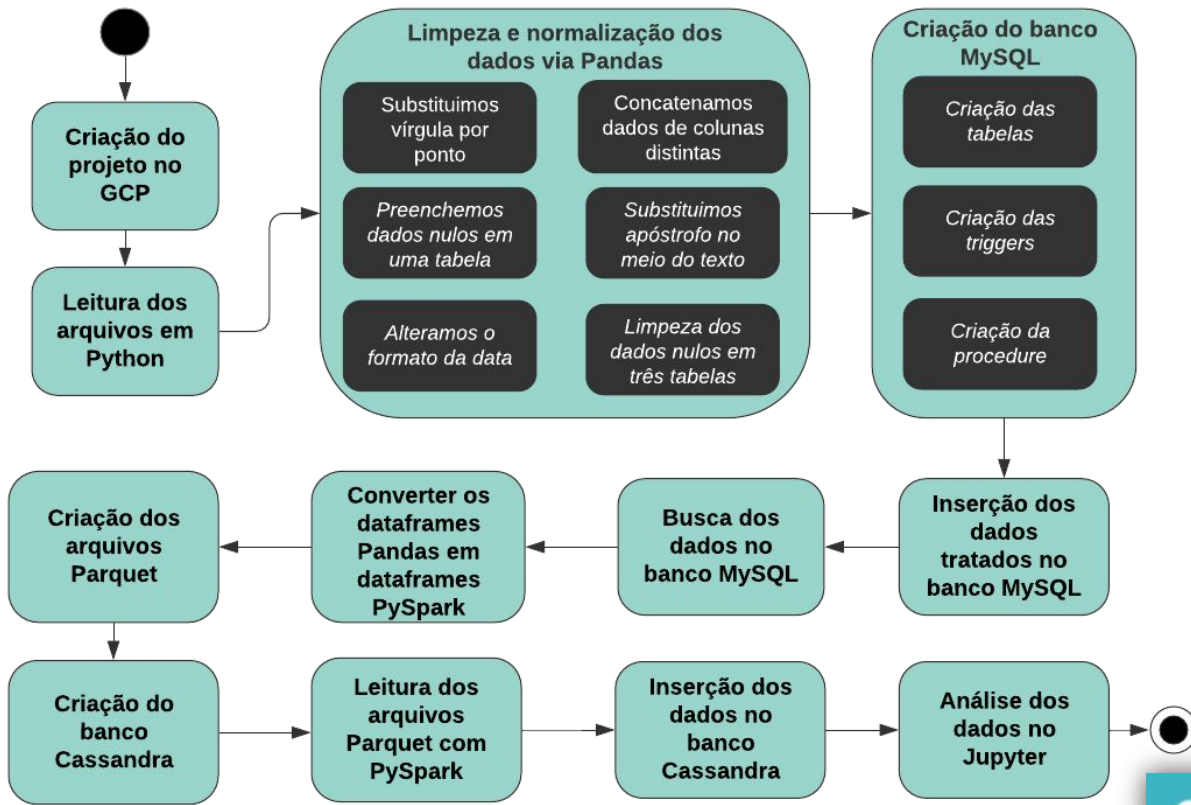
Ferramentas



Ferramentas



Processos



Banco de dados - SQL

```
create table if not exists empreendimentosGD(
```

```
create table if not exists tarifaMediaFornecimento (
```

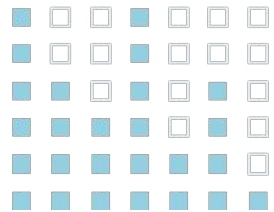
```
create table if not exists tarifaResidencial (
```

```
create table if not exists geracaoDistribuida(
```

```
create table if not exists logDados(
```

	usuario_log	data_log	tipo_registro
▶	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD
	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD
	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD
	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD
	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD
	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD
	root@%	2022-01-14 14:29:24	Item inserido na tabela empreendimentosGD

Banco de dados - NoSQL



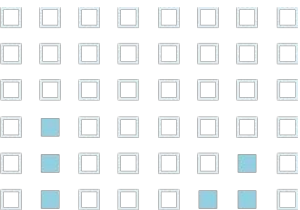
```
CREATE KEYSPACE IF NOT EXISTS projeto_final WITH replication = {'class' : 'SimpleStrategy', 'replication_factor' : 1};
```

```
create table if not exists empreendimentosGD(
```

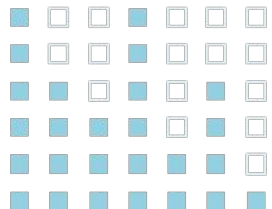
```
create table if not exists tarifaMediaFornecimento(
```

```
create table if not exists tarifaResidencial(
```

```
create table if not exists geracaoDistribuida(
```



Tratamento dos dados

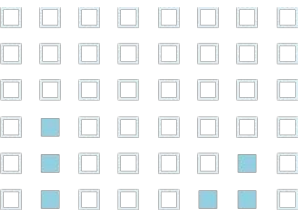


- Coluna de valor com separador decimal sendo vírgula

```
# Substituindo ',' por '.' na coluna de Potencia instalada
try:
    dados_emprego['PotenciaInstaladaKW'] = dados_emprego['PotenciaInstaladaKW'].str.replace(",", ".")
except Exception as e:
    print('Erro ao substituir virgula por ponto - empreendimento-geracao-distribuida:', e)
```

- Texto com apóstrofo

```
# Retirando os apóstrofes dos dados
try:
    dados_emprego['NomMunicipio'] = dados_emprego['NomMunicipio'].str.replace("'", "")
    dados_emprego['NomTitularUC'] = dados_emprego['NomTitularUC'].str.replace("'", "")
except Exception as e:
    print('Erro ao retirar os apóstrofes - empreendimento-geracao-distribuida:', e)
```

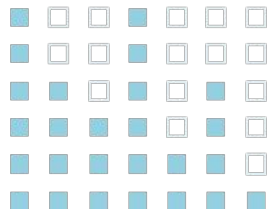


Tratamento dos dados

- Texto com vírgula

```
# Inserindo no banco MySQL
# Laço for para correção de anomalia encontrada na linha 14 do CSV
try:
    for coluna, linha in dados_tarifa_media.iterrows():
        if(linha[2] == " Serviços e Outras" or linha[2] == " esgoto e saneamento"):
            linha[1] = linha[1] + linha[2]
            linha[2] = linha[3]
            linha[3] = linha[4]
            linha[4] = linha[5]
            linha[5] = linha[6]
            linha[6] = linha[7]
            linha[7] = np.NaN
```

Tratamento dos dados

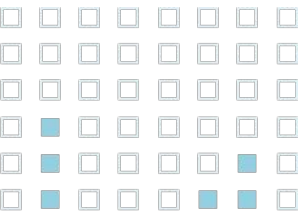


- Exclusão de linhas com campos vazios

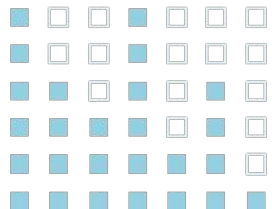
```
# Dropando linhas em que as colunas não possuem dados
try:
    dados_tarifa_residencial.dropna(axis=0, subset=["ideTarifaFornecimento", "nomConcessao", "SigDistribuidora",
except Exception as e:
    print('Erro ao dropar linhas com dados em branco', e)
```

- Preenchendo dados faltantes

```
# Preenchendo colunas vazias:
try:
    values = {"CNPJ_Distribuidora": 'ND', "SigAgente": 'ND', "NomAgente": 'ND', "MdaLatitude": 0, "MdaLongitude": 0}
    dados_empresa.fillna(value=values, inplace= True)
except Exception as e:
    print('Erro ao preencher colunas vazias - empreendimento-geracao-distribuida: ', e)
```



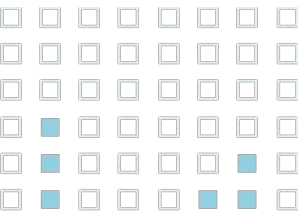
Tratamento dos dados



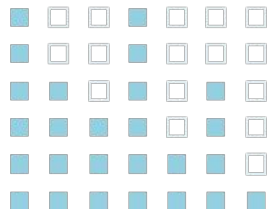
- Datas

```
# Transforma data da coluna 1 e 21 em DateTime
try:
    linha[1] = datetime.combine(datetime.strptime(linha[1], '%d/%m/%Y %H:%M').date(), datetime.strptime(linha[1], '%d/%m/%Y %H:%M').time())
    linha[21] = datetime.strptime(linha[21], '%d/%m/%Y').date() #altera o formato da data (coluna 21)
except Exception as e:
    print('Erro ao transformar em datetime - empreendimento-geracao-distribuida:', e)

# Transforma data da coluna 7 em DateTime
try:
    linha[7] = datetime.combine(datetime.strptime(linha[7], '%d/%m/%Y %H:%M').date(), datetime.strptime(linha[7], '%d/%m/%Y %H:%M').time())
except Exception as e:
    print('Erro ao transformar data em datetime - GeracaoDistribuida', e)
```

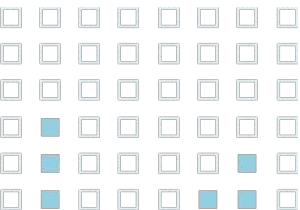


Tratamento dos dados



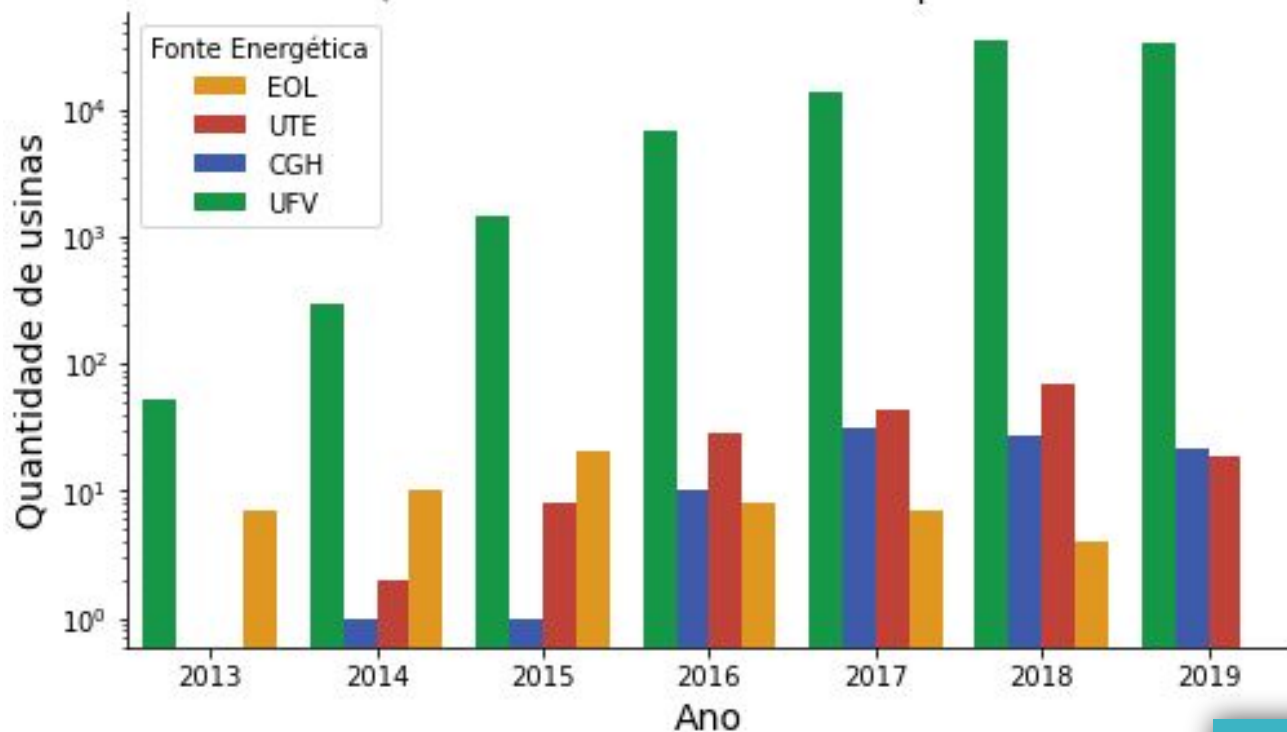
- Criação dos arquivos parquet

```
# -----CRIAÇÃO DOS ARQUIVOS PARQUET-----  
try:  
    print('Criando arquivos parquet')  
    spk_empresendimentos_gd.write.parquet(caminho + "empresendimentos_gd")  
    spk_geracao_distribuida.write.parquet(caminho + "geracaodistribuida")  
    spk_tarifa_residencial.write.parquet(caminho + "tarifamediaresidencial")  
    spk_tarifa_media.write.parquet(caminho + "tarifamediafornecimento")  
except Exception as e:  
    print('Erro ao criar arquivos Parquet: ', e)
```

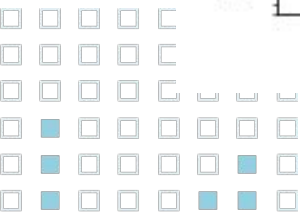
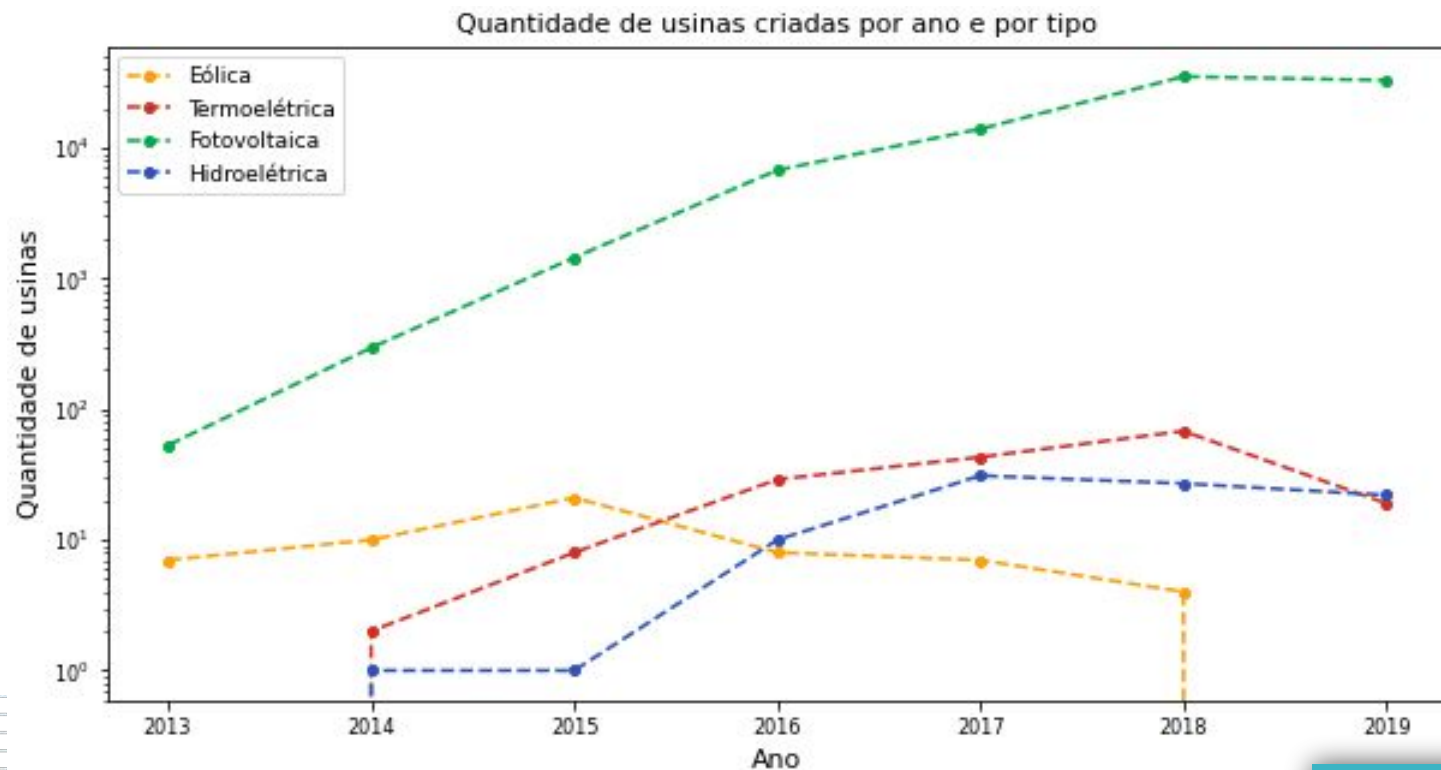
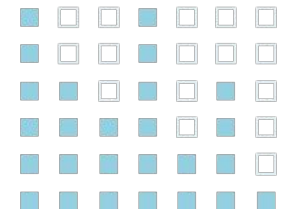


Análises

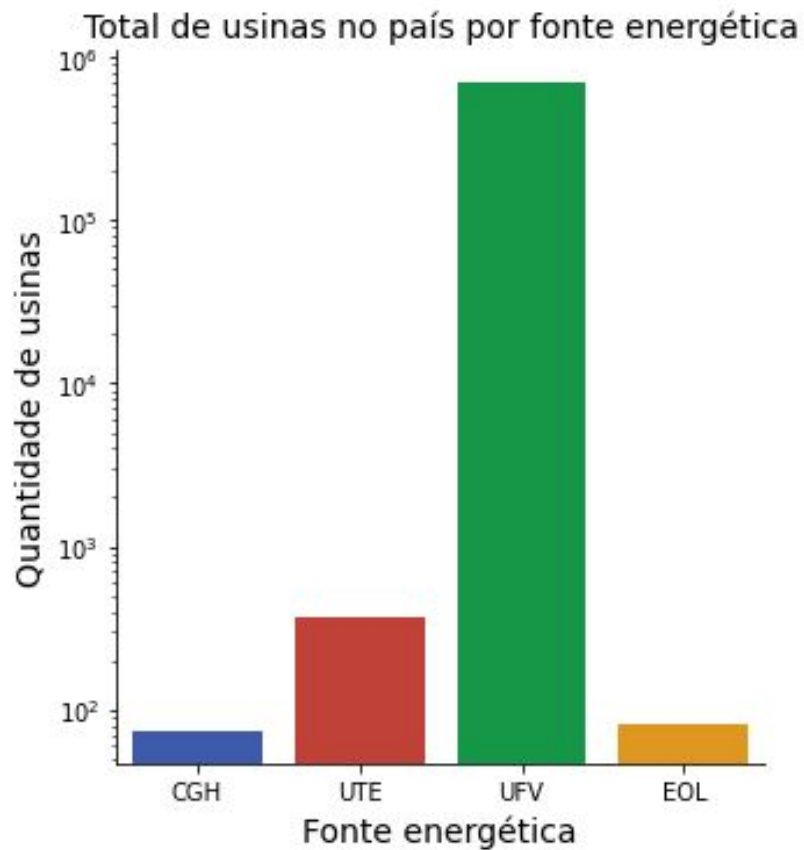
Quantidade de usinas criadas por ano



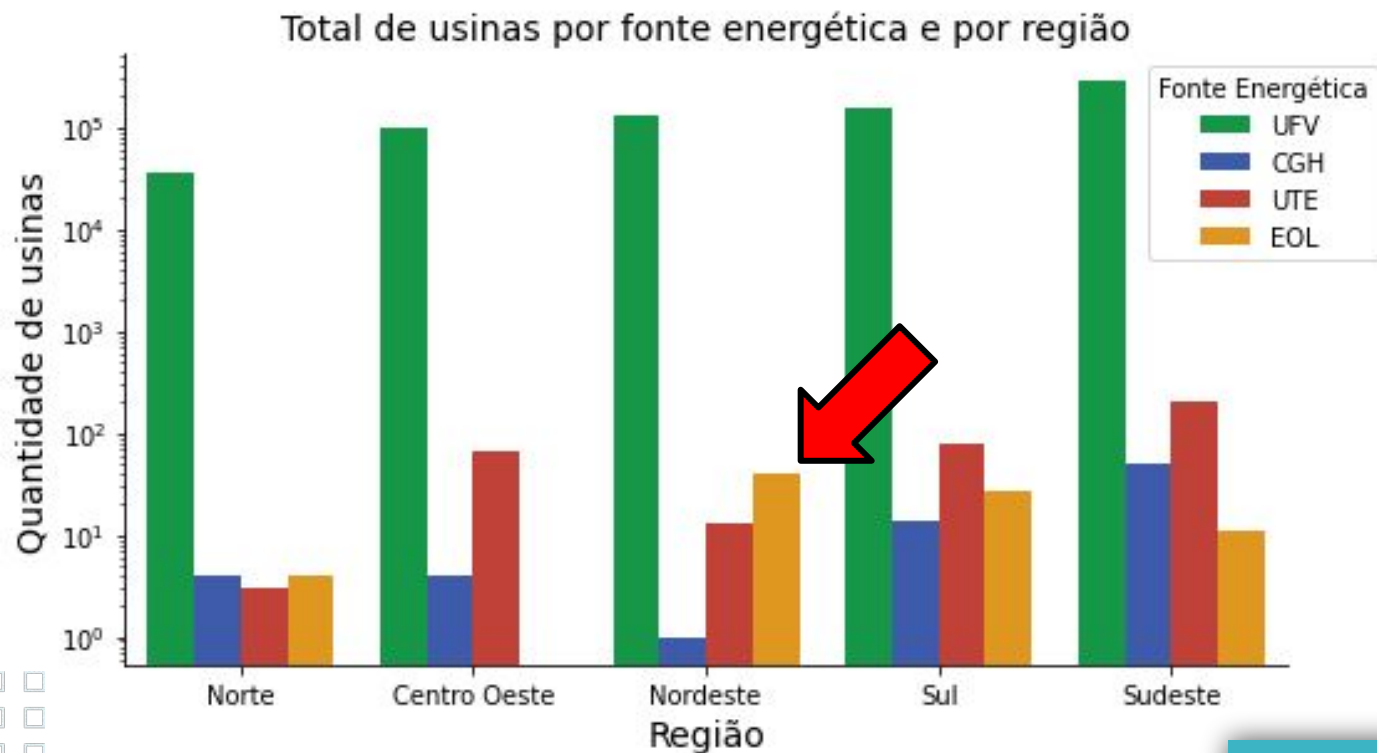
Análises



Análises

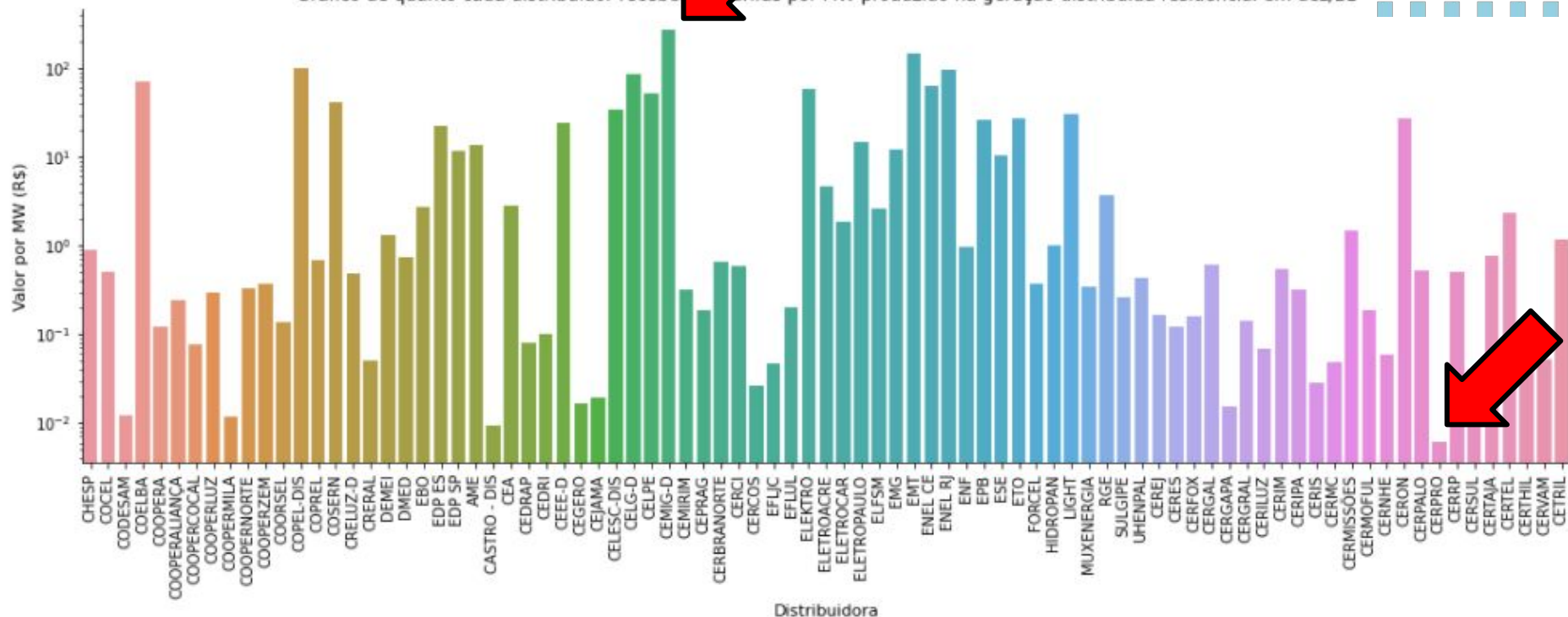


Análises



Análises

Gráfico de quanto cada distribuidor recebe tarifas por MW produzido na geração distribuída residencial em dez/21



GERAÇÃO DISTRIBUÍDA DE ENERGIA NO BRASIL

Obrigado!



Aurélia Covre

João Victor Guimarães

Kely Fernandes

Ricardo Rowedder

Robson Motta

SOUL CODE