Cast House and Consumer Analysis

Table of Contents

1. Problem Statement

When discussing **Cast House and Process Analysis** in the context of the steel or iron industry, the two topics are often intertwined, as the cast house plays a critical role in producing the raw materials that various downstream consumers need for their own processes. A cast house in a steel or iron plant produces molten iron or steel in forms such as **ingots, slabs, billets, or blooms**, which are then sold to a wide range of consumers for further processing or manufacturing.

Cast House Overview

The **cast house** is the part of the plant where molten iron or steel is cooled and solidified into a form that can be transported and used by various industries. In iron and steel production, cast houses are crucial because they convert molten metal from a blast furnace or electric arc furnace (EAF) into products that are easier to handle, store, and ship to end users.

Key Products from the Cast House:

1. **Pig Iron**: A primary product of the blast furnace, used for further refining into steel or castings.
2. **Slabs**: Used as feedstock for flat products (e.g., sheet metal).
3. **Billets**: Typically used for the production of long products like rods, bars, or rails.
4. **Blooms**: Larger sections of steel used to make structural beams, heavy sections, or railway lines.
5. **Ingot**: A traditional form of casting, still used in some areas for further processing into various steel products.

Cast House Consumer Analysis

Consumer analysis in this context refers to identifying the industries or sectors that rely on the products cast in the cast house. These consumers can be broken down into different sectors based on the forms of cast products they require and the type of manufacturing processes they are involved in.

Consumer Preferences and Trends in the Cast House:

- **Quality and Consistency**: Consumers typically demand high-quality materials with consistent properties, particularly in industries like automotive and aerospace, where strength, ductility, and toughness are crucial.
- **Product Customization**: Different industries require different steel grades, chemical compositions, and physical properties. Cast houses need to provide flexibility in terms of product customization to meet specific consumer requirements.
- **Cost Sensitivity**: While some sectors (like aerospace) are less price-sensitive, most sectors such as construction and automotive are highly cost-sensitive. The cast house must balance the need for high-quality products with efficient, cost-effective production.
- **Sustainability**: Increasingly, consumers are demanding more sustainable production processes. This includes reducing CO_2 emissions, recycling, and using environmentally friendly materials. Some cast houses are working to reduce their carbon footprint by adopting cleaner technologies like electric arc furnaces (EAFs) and employing carbon capture solutions.

Impact of Market Demand on the Cast House:

- **Capacity Planning**: Depending on the demand from consumers (e.g., higher demand for automotive steel or construction materials), cast houses may need to adjust their production volumes and product types. For example, a rise in infrastructure projects can lead to higher demand for long products like beams and rebar.

- **Supply Chain Coordination**: Efficient coordination between the cast house and downstream consumers is crucial for maintaining the flow of materials. Delays or inefficiencies in the cast house can have a ripple effect on industries that depend on steel or iron products.
- **Technology Investments**: Modern consumers often require high-quality and specialized steel. Cast houses may need to invest in **advanced casting technologies** (like continuous casting machines) to meet these demands.

Conclusion:

The **Cast House** is a vital component of the steelmaking process, serving as the final link between the molten metal production and its end-users. **Consumer analysis** helps cast houses understand the demands of various industries (such as automotive, construction, heavy equipment, energy, and more) and tailor their production processes to meet these needs effectively. By aligning production with consumer requirements, a cast house can ensure its products are used optimally and remain competitive in a global market. The relationships between the cast house and its consumers are fundamental to ensuring a seamless and efficient supply chain in the steel and iron industries.

---

2. Importing Libraries

In [1]:

```
import math as ma
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import scipy as sp
#---------------------------------------------------------------------------------------------------------------------------
from wordcloud import WordCloud                    # To create wordclouds
from wordcloud import STOPWORDS                    # To import a list of stopwords that remove unnecessary unicode text
#---------------------------------------------------------------------------------------------------------------------------
from urllib.parse import quote_plus
from sqlalchemy import create_engine
from sqlalchemy import text
#---------------------------------------------------------------------------------------------------------------------------
from PIL import Image                                                # To import an image
```

---

3. Loading & Description Data

In [2]:

```
pass_ = input("Password")

# establish connection with the database
# engine = create_engine( f"""mysql+mysqlconnector://root:{quote_plus(pass_)}@localhost:3306/______""")
engine = create_engine( f"""mysql+mysqlconnector://root:{quote_plus(pass_)}@localhost:3306/Company""")
```

In [3]:

```
# read table data using sql query
cast_house = pd.read_sql_query(text("SELECT * FROM company.cast_house;"), con=engine.connect())
cast_house
```

Out[3]:

	Date	TAPNO	HOTMETALSTART	HOTMETALEND	SLAGSTART	CastHouse	Shift	DrillBitDIAMM	TapHoleLength	Mass	Pushed	Plu
0	01-01-2024	125743	22:54	0:24	23:31	A	C	45	2.5	VVEL 1 : SHIN 1	160	
1	01-01-2024	125744	1:44	3:05	2:20	A	C	45	2.5	VVEL 1 : SHIN 1	160	
2	01-01-2024	125745	4:20	5:40	5:06	A	C	45	2.5	VVEL 1 : SHIN 1	160	
3	01-01-2024	125746	7:05	8:35	7:35	A	A	45	2.5	VVEL 1 : SHIN 1	160	
4	01-01-2024	125747	20:10	21:54	21:23	A	B	45	2.7	VVEL 1 : SHIN 1	160	
...	...	...	...	...	...	...	...	...	...	...	...	
252	31-01-2024	125995	11:19	12:20	11:49	A	A	45	2.2	VVEL 2 : SHIN 1	160	
253	31-01-2024	125996	14:08	15:55	14:40	A	A	45	2.3	CHUS 1 : VVEL 2	160	
254	31-01-2024	125997	16:54	18:02	17:22	A	B	45	2.3	CHUS 1 : VVEL 2	160	
255	31-01-2024	125998	19:30	21:00	20:14	A	B	45	2.3	CHUS 1 : VVEL 2	160	
256	31-01-2024	125999	22:20	23:45	22:45	A	B	45	2.4	CHUS 1 : VVEL 2	160	

257 rows × 14 columns

In [4]:

```
process_analysis = pd.read_sql_query(text("SELECT * FROM company.process_analysis;"), con=engine.connect())
process_analysis
```

Out[4]:

	Date	TAPNO	CastHouse	Shift	TEMP	C	Si	S	P	Mn	CaO	MgO	SiO2	Al2O3	S.1	FeO
0	01-01-2024	125743	A	C	1479.0	3.72	0.99	0.041	0.054	0.06	38.24	10.65	29.86	17.92	0.96	0.68
1	01-01-2024	125744	A	C	1419.0	3.63	0.84	0.052	0.058	0.05	38.69	9.66	31.24	17.03	0.89	0.77
2	01-01-2024	125745	A	C	1440.0	3.59	1.00	0.061	0.057	0.06	37.75	10.20	31.50	16.91	0.80	1.14
3	01-01-2024	125746	A	A	1434.0	3.35	0.89	0.043	0.095	0.06	37.72	10.13	31.26	16.99	0.83	1.34
4	01-01-2024	125747	A	B	1449.0	3.97	1.75	0.035	0.076	0.07	38.73	9.93	30.45	17.84	0.87	0.61
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
252	31-01-2024	125995	A	A	1434.0	4.31	0.97	0.059	0.086	0.06	37.73	10.16	31.95	16.45	0.95	1.03
253	31-01-2024	125996	A	A/B	1469.0	3.81	1.47	0.045	0.109	0.07	37.21	9.99	30.85	18.93	0.98	0.33
254	31-01-2024	125997	A	B	1500.0	4.65	1.28	0.032	0.100	0.06	39.00	10.19	30.37	17.89	0.99	0.16
255	31-01-2024	125998	A	B	1458.0	4.30	0.85	0.058	0.103	0.06	37.58	11.42	30.65	17.47	1.04	0.36
256	31-01-2024	125999	A	B/C	1499.0	4.92	1.64	0.020	0.086	0.07	37.55	11.63	30.16	17.99	1.07	0.25

257 rows × 18 columns

In [5]:

```
production = pd.read_sql_query(text("SELECT * FROM company.production;"), con=engine.connect())
production
```

Out[5]:

	Date	Prodution
0	01-01-2024	1331
1	02-01-2024	2304
2	03-01-2024	2324
3	04-01-2024	2346
4	05-01-2024	2314
5	06-01-2024	2337
6	07-01-2024	2346
7	08-01-2024	2313
8	09-01-2024	2356
9	10-01-2024	2286
10	11-01-2024	2250
11	12-01-2024	2314
12	13-01-2024	2322
13	14-01-2024	2331
14	15-01-2024	2287
15	16-01-2024	2268
16	17-01-2024	2210
17	18-01-2024	2278
18	19-01-2024	2288
19	20-01-2024	985
20	21-01-2024	2210
21	22-01-2024	2375
22	23-01-2024	2375
23	24-01-2024	2376
24	25-01-2024	2380
25	26-01-2024	2360
26	27-01-2024	2338
27	28-01-2024	2261
28	29-01-2024	2358
29	30-01-2024	2376
30	31-01-2024	2322

Here's a breakdown of the additional dataset features based on the provided IDs:

ID	Feature Name	Feature Description
Date	Date	The specific date when the event or measurement was recorded.
TAPNO	Tap Number	The identifier for a particular tapping event, often used in steelmaking to track individual batches of metal tapped from a furnace.
HOT METAL START	Hot Metal Start	The time or event when the tapping of hot metal (molten iron) begins.
HOT METAL END	Hot Metal End	The time or event when the tapping of hot metal ends.
SLAG START	Slag Start	The time or event when the slag (byproduct of the smelting process) starts being removed from the furnace.
Cast House	Cast House	The specific casting facility where the molten metal is poured into molds or cast into shapes.
Shift	Shift	The work shift (e.g., day, night) during which the operations occurred.
Drill Bit DIA MM.	Drill Bit Diameter (mm)	The diameter of the drill bit used, likely referring to the size of the tap hole drill used to release molten metal or slag.

ID	Feature Name	Feature Description
Tap Hole Length	Tap Hole Length	The length of the tap hole through which the molten metal or slag flows out from the furnace.
Mass	Mass	The mass of material (metal or slag) involved in the tapping or other process.
Pushed	Pushed	A term likely referring to the number of furnace pushes, i.e., actions taken to drive or remove material through the furnace or during tapping.
Plugging Pressure	Plugging Pressure	The pressure used to plug or seal the tap hole after tapping is complete, critical for maintaining furnace integrity.
SGP	SGP	Could refer to "Specific Gravity of the Product," related to the density of the material, or another term specific to the process—clarification may be needed.
DSP A/B	DSP A/B	Refers to the status or identification of two units (A and B) in the process, possibly part of a twin-furnace or twin-tap setup.
PRODUCTION (TON)	Production (Tons)	The total production quantity of the tapped metal or material in tons during a specific process or time period.

These features seem to relate to the production, measurement, and management of metallurgical processes, particularly during the tapping phase where molten metal and slag are removed from a furnace. The features help track operational efficiency, materials, and key process timings.

Chemical composition and other related features:

Here's a breakdown of the chemical composition and related features based on the IDs you've listed:

ID	Feature Name	Feature Description
TEMP	Temperature	The temperature of the material or system. Important for controlling chemical reactions and transformations.
C	Carbon (C)	Represents the concentration of carbon, often measured in %wt, crucial for steel and alloy production.
Si	Silicon (Si)	Silicon content, usually in %wt. Important in steel to improve strength and resist oxidation.
S	Sulfur (S)	Sulfur content in the material, typically undesirable in metals as it makes them brittle.
P	Phosphorus (P)	Phosphorus content, which can affect the hardness and strength of metals but in excess, can lead to brittleness.
Mn	Manganese (Mn)	Manganese is used to improve hardness and strength and reduce the effects of sulfur in steel.
CaO	Calcium Oxide (CaO)	Commonly known as quicklime, used in refining processes, helps remove impurities in steelmaking.
MgO	Magnesium Oxide (MgO)	A refractory material often used to protect furnace linings and as a flux in steelmaking.
SiO2	Silicon Dioxide (SiO2)	Also known as silica, a key component in slag formation in metallurgy, helps remove impurities.
Al2O3	Aluminum Oxide (Al2O3)	Used as a refractory material, also forms part of the slag in metal purification processes.
S.1	Sulfur (S) in another context	Likely an additional sulfur measurement under different conditions or at a different process stage.
FeO	Iron Oxide (FeO)	Represents the presence of iron in its oxidized state, commonly present in slags and influences the oxidation-reduction reactions.
BAS2	Basicity 2 (CaO/SiO2)	The ratio of CaO to SiO2, used to evaluate the slag's basicity, which affects slag formation and refining ability in metallurgy.
BAS4	Basicity 4	Another basicity ratio, possibly involving other compounds like MgO or Al2O3, which also impact slag performance in metallurgical processes.

These features are often related to metallurgical and refining processes, particularly in steelmaking, where controlling the composition of elements and oxides is essential for achieving desired properties in the final product.

In [6]:

```
price_consumable = pd.read_sql_query(text("SELECT * FROM company.price_consumable;"), con=engine.connect())
price_consumable
```

Out[6]:

	Name_Item	Chuson	VVEL	Shinogowa	MISUS	Vesuvius	Drill_bit_35mm	Drill_bit_40mm	Drill_bit_45mm	Drill_Rod	...	Thermocoupl
0	Qty	MT	MT	MT	MT	MT	Nos	Nos	Nos	Nos	...	N
1	Price	100000	60000	100000	82000	60000	630	635	640	2200	...	

2 rows × 21 columns

This dataset seems to contain information about various consumables, their quantities, and corresponding prices. Here's a breakdown of the features and what they represent:

1. **Name_Item**: The list of items or consumables used in a specific operation, possibly in a casting house (e.g., Chuson, VVEL, Shinogowa, etc.).
2. **Qty**: The unit of quantity for each item. For most of the items, it appears to be measured in metric tons (MT), while some items are measured in numbers (Nos) or meters (mtrs).
3. **Price**: The cost of each item in the dataset. This is most likely the unit price for each consumable, expressed in a currency like USD or another local currency.

Example Breakdown:

- **Chuson**: Measured in MT (Metric Tons) with a unit price of 100,000.
- **Drill_bit_35mm**: Measured in Nos (numbers) with a price of 630 per piece.
- **Ladle_covering_compound**: Measured in MT with a unit price of 5400.

Possible Analysis:

You can analyze this dataset in various ways:

1. **Cost Efficiency**: Analyze which items are most expensive based on price per unit (either per ton, number, or meter).
2. **Consumption Trends**: If you have historical data on quantities used, you could identify which consumables are used most frequently or contribute the most to total costs.
3. **Cost Allocation**: If you have production data, you could allocate these costs to specific production activities or products.

In [7]:

```
cast_house_consumable = pd.read_sql_query(text("SELECT * FROM company.cast_house_consumable;"), con=engine.connect())
cast_house_consumable
```

Out[7]:

	Date	Chuson	VVEL	Shinogowa	MISUS	Vesuvius	Drill_bit_35mm	Drill_bit_40mm	Drill_bit_45mm	Drill_Rod	...	Thermocouple_tip	Ladl
0	01-01-2024	0	1	2	0	0	0	0	9	8	...	200	
1	02-01-2024	0	1	2	0	0	0	0	9	8	...	0	
2	03-01-2024	0	2	2	0	0	0	0	9	8	...	0	
3	04-01-2024	0	1	2	0	0	0	0	9	8	...	0	
4	05-01-2024	0	1	2	0	0	0	4	8	8	...	0	
5	06-01-2024	0	1	2	0	0	0	4	8	9	...	0	
6	07-01-2024	0	2	2	0	0	0	0	8	8	...	0	
7	08-01-2024	0	1	2	0	0	0	0	8	9	...	0	
8	09-01-2024	0	1	1	1	0	0	0	9	8	...	0	
9	10-01-2024	0	2	2	0	0	0	0	9	9	...	0	
10	11-01-2024	0	1	2	0	0	0	0	6	8	...	0	
11	12-01-2024	0	2	1	0	0	0	6	6	9	...	0	
12	13-01-2024	0	2	1	0	0	0	6	6	8	...	0	
13	14-01-2024	1	1	0	0	0	0	0	8	9	...	0	
14	15-01-2024	2	1	0	0	0	0	0	8	7	...	0	
15	16-01-2024	2	2	0	0	0	0	0	9	8	...	0	
16	17-01-2024	1	2	0	1	0	0	8	9	8	...	0	
17	18-01-2024	0	2	1	0	0	0	8	0	8	...	300	

	Date	Chuson	VVEL	Shinogowa	MISUS	Vesuvius	Drill_bit_35mm	Drill_bit_40mm	Drill_bit_45mm	Drill_Rod	...	Thermocouple_tip	Ladl
18	19-01-2024	0	2	2	0	0	0	0	0	8	...	0	
19	20-01-2024	0	2	1	0	0	0	0	8	8	...	0	
20	21-01-2024	0	2	2	0	0	0	0	8	8	...	0	
21	22-01-2024	0	2	2	0	0	0	8	0	8	...	0	
22	23-01-2024	0	2	2	0	0	0	8	0	8	...	0	
23	24-01-2024	0	2	2	0	0	0	0	8	8	...	0	
24	25-01-2024	0	2	0	2	0	0	0	8	8	...	0	
25	26-01-2024	0	2	0	3	0	0	0	8	8	...	0	
26	27-01-2024	0	2	1	2	0	0	0	8	8	...	0	
27	28-01-2024	0	2	0	1	0	0	0	8	8	...	0	
28	29-01-2024	0	2	0	0	0	0	0	8	8	...	0	
29	30-01-2024	0	2	0	0	0	0	0	8	8	...	0	
30	31-01-2024	0	2	1	0	0	0	0	8	8	...	0	

31 rows × 21 columns

The second dataset you've provided seems to represent daily consumption data of various items (consumables) in a cast house for the month of January 2024. Here's a breakdown of its structure:

Dataset Features:

1. **Name_Item**: Represents various consumables, similar to the first dataset.
2. **Date**: Daily records from January 1st to January 31st, 2024.
3. **Consumption Values**: Quantities consumed on each day for each item, measured in their respective units (e.g., Metric Tons, Nos, meters). For each item, a numeric value indicates the quantity consumed, and zero (0) means no consumption on that day.

Notable Features:

- **Drill_bit_35mm, 40mm, 45mm, Drill_Rod, Pocking_Rod**: These consumables are generally represented in numbers (Nos).
- **Ladle_covering_compound, Rice_Husk, Bamboo, Ecombetec_120**: These are in MT (Metric Tons).
- **Lancing_pipe_8_mm, 15_mm, 20_mm**: These are consumed in meters (mtrs).

- **Thermocouple_tip, Leackage_arrester**: These are consumed in Nos.
- **Sand**: Measured in MT.

Possible Analysis:

1. **Daily Consumption Trends**:
   - Analyze the daily usage patterns for each consumable. For instance, you can identify items with consistent usage (e.g., **Drill_bit_45mm** consumed on all days) versus items with occasional usage (e.g., **Chuson** or **MISUS**).
2. **High and Low Consumption**:
   - Calculate the total consumption for each item throughout the month and identify which items are used most frequently (e.g., high consumption of **Pocking_Rod**, **Drill_Rod**, etc.) and which are used less frequently (e.g., **MISUS**, **Chuson**).
3. **Cost Correlation**:
   - Cross-reference the consumption data with the price data provided earlier to calculate the total cost for each consumable over the month.
   - Identify high-cost consumables by multiplying the daily consumption by the price per unit for each item.
4. **Days of Peak Consumption**:
   - Find days where consumption spikes for certain items and analyze if there are operational or production-related reasons for this.
5. **Consumption by Category**:
   - Group consumables by their category (e.g., drilling equipment, pipes, chemicals) and analyze the total usage per category.

Would you like me to perform any specific analysis based on this dataset, such as calculating total consumption, identifying high-cost items, or visualizing the trends?

---

4. Exploratory Data Analysis

---

1. Data Analysis

In [8]:

```
cast_house1 = pd.read_sql_query(text("SELECT Date,TAPNO,CastHouse,Shift,DrillBitDIAMM,TapHoleLength,Mass,PluggingPressure FROM company.cast_
cast_house1
```

Out[8]:

	Date	TAPNO	CastHouse	Shift	DrillBitDIAMM	TapHoleLength	Mass	PluggingPressure
0	01-01-2024	125743	A	C	45	2.5	VVEL 1 : SHIN 1	146
1	01-01-2024	125744	A	C	45	2.5	VVEL 1 : SHIN 1	190
2	01-01-2024	125745	A	C	45	2.5	VVEL 1 : SHIN 1	181
3	01-01-2024	125746	A	A	45	2.5	VVEL 1 : SHIN 1	141
4	01-01-2024	125747	A	B	45	2.7	VVEL 1 : SHIN 1	166
...	...	...	...	...	...	...	...	...
252	31-01-2024	125995	A	A	45	2.2	VVEL 2 : SHIN 1	215
253	31-01-2024	125996	A	A	45	2.3	CHUS 1 : VVEL 2	229
254	31-01-2024	125997	A	B	45	2.3	CHUS 1 : VVEL 2	180
255	31-01-2024	125998	A	B	45	2.3	CHUS 1 : VVEL 2	211
256	31-01-2024	125999	A	B	45	2.4	CHUS 1 : VVEL 2	205

257 rows × 8 columns

In [9]:

```
cast_house1.describe()
```

Out[9]:

	TAPNO	DrillBitDIAMM	TapHoleLength	PluggingPressure
count	257.000000	257.000000	257.000000	257.000000
mean	125871.000000	44.883268	2.334241	213.276265
std	74.333707	0.756477	0.142778	35.207054
min	125743.000000	40.000000	1.800000	80.000000
25%	125807.000000	45.000000	2.200000	190.000000
50%	125871.000000	45.000000	2.400000	214.000000
75%	125935.000000	45.000000	2.400000	241.000000
max	125999.000000	45.000000	2.700000	291.000000

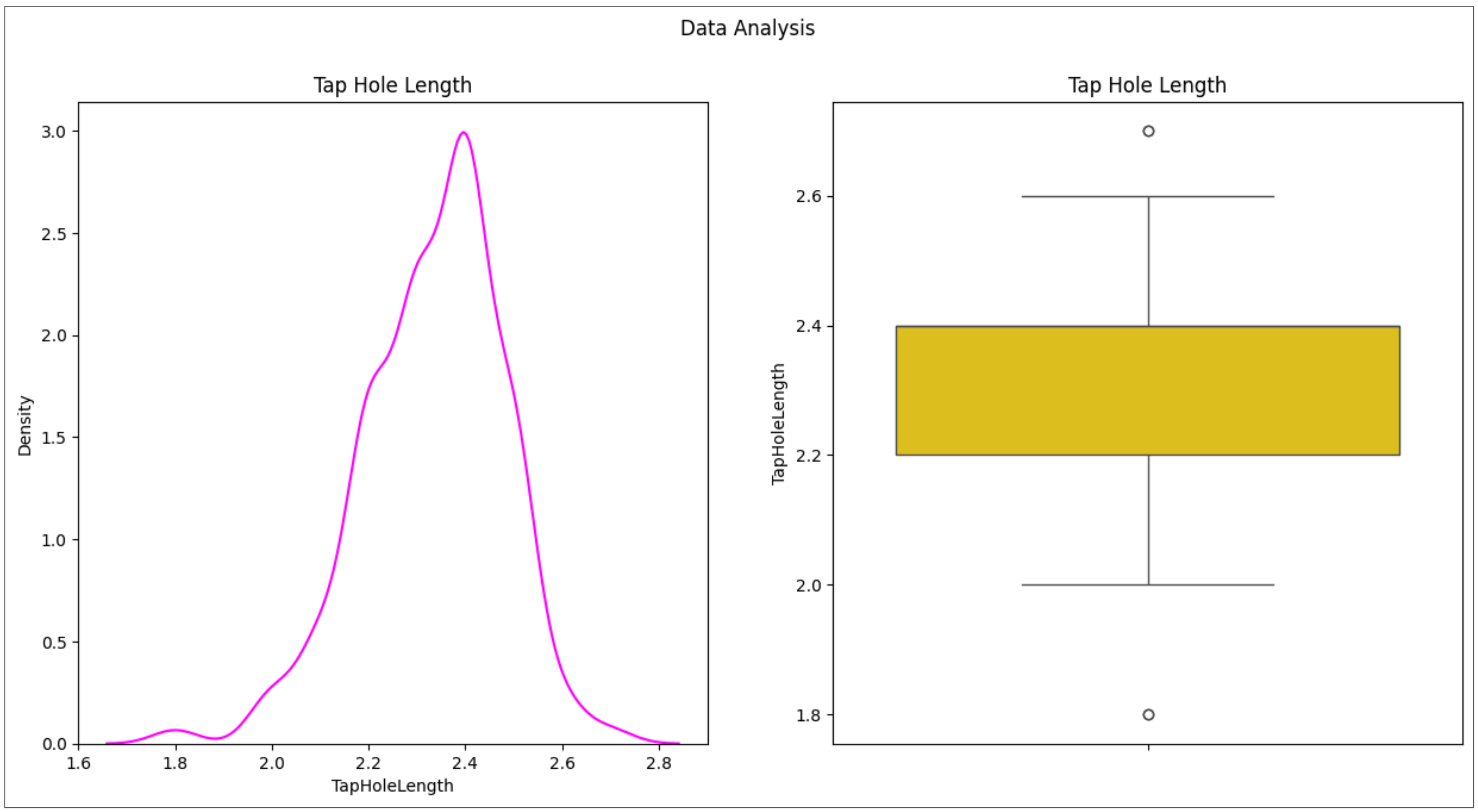
In [10]:

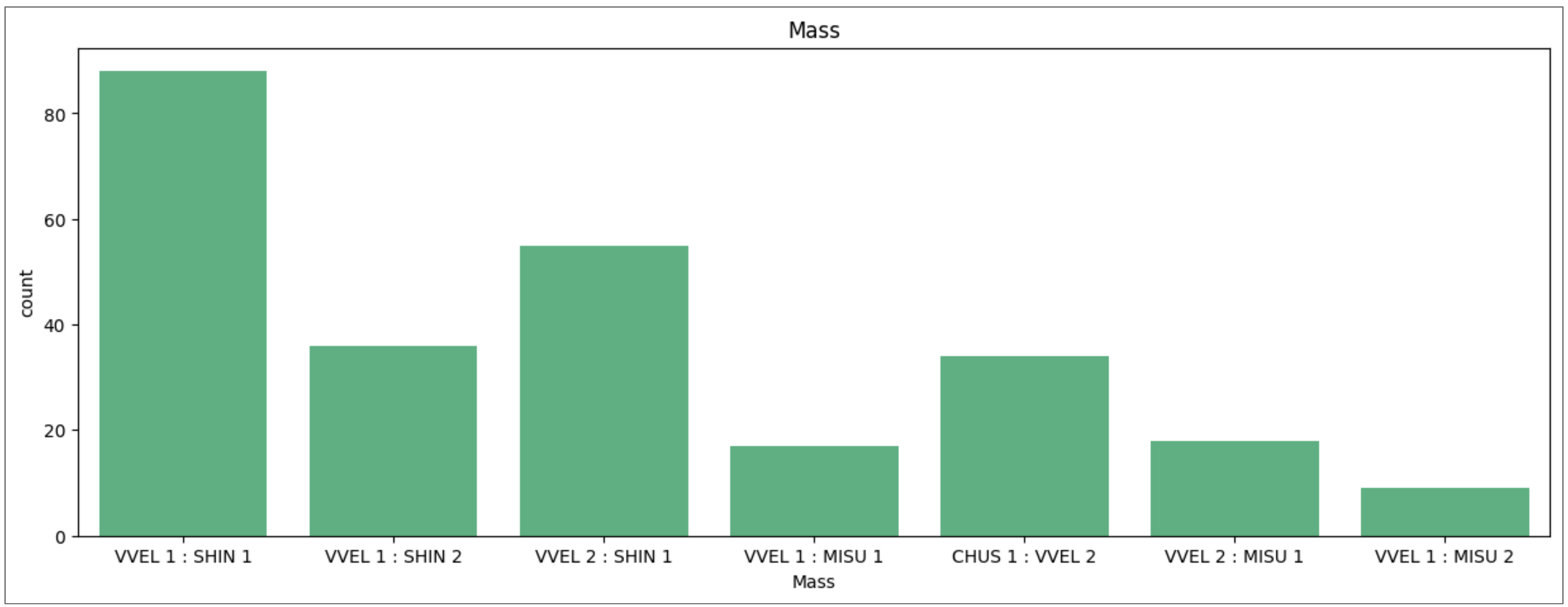
```
cast_house1.info()
```

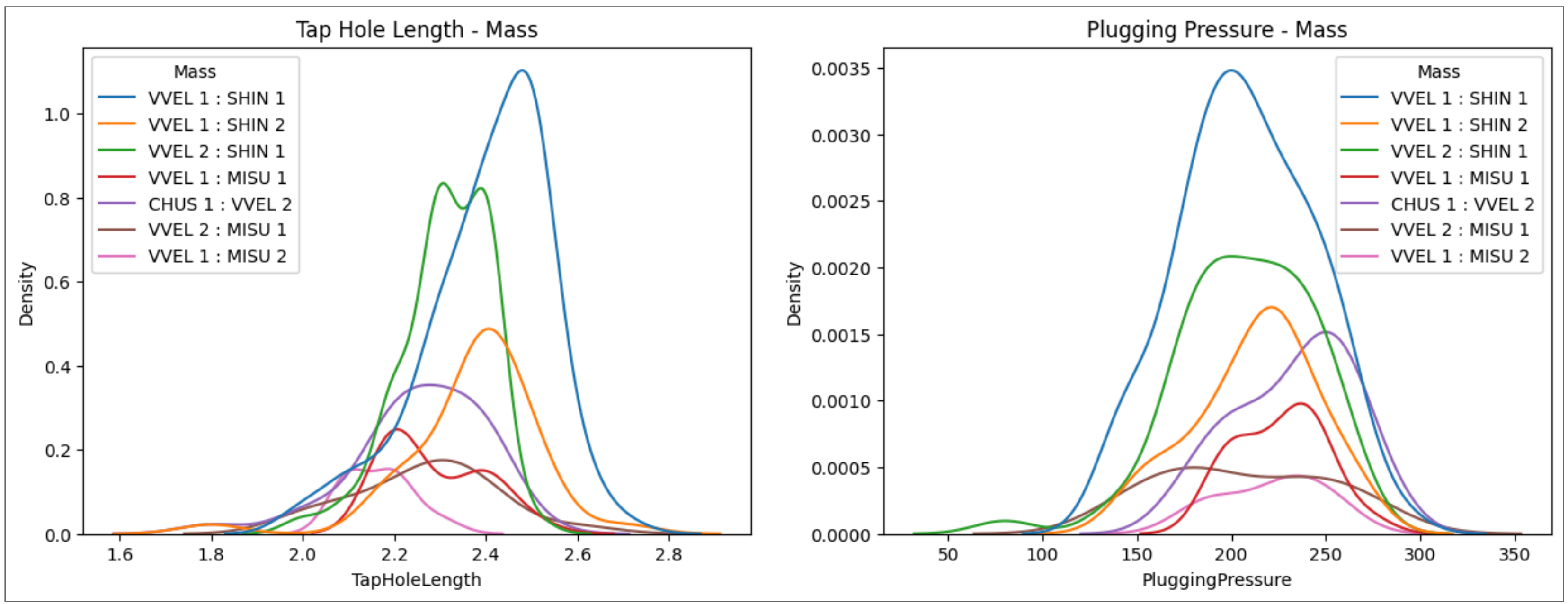
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 257 entries, 0 to 256
Data columns (total 8 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   Date              257 non-null    object 
 1   TAPNO             257 non-null    int64  
 2   CastHouse         257 non-null    object 
 3   Shift             257 non-null    object 
 4   DrillBitDIAMM     257 non-null    int64  
 5   TapHoleLength     257 non-null    float64
 6   Mass              257 non-null    object 
 7   PluggingPressure  257 non-null    int64  
dtypes: float64(1), int64(3), object(4)
memory usage: 16.2+ KB
```

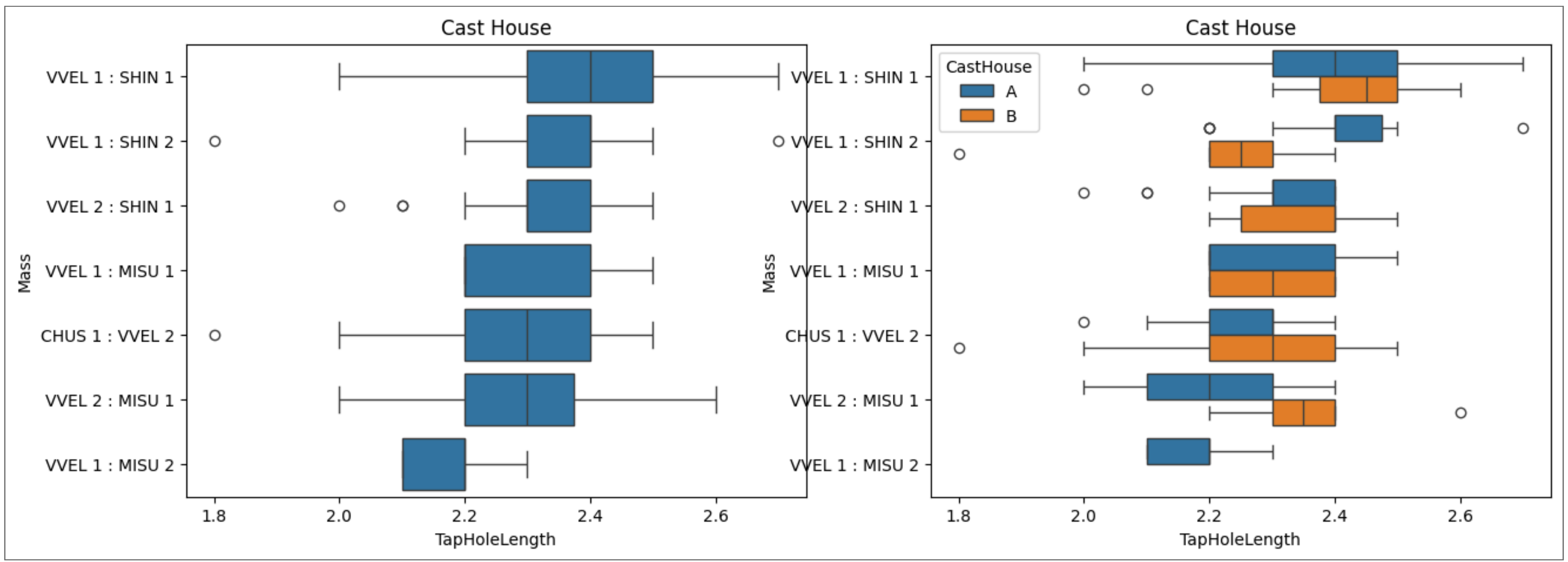
In [11]:

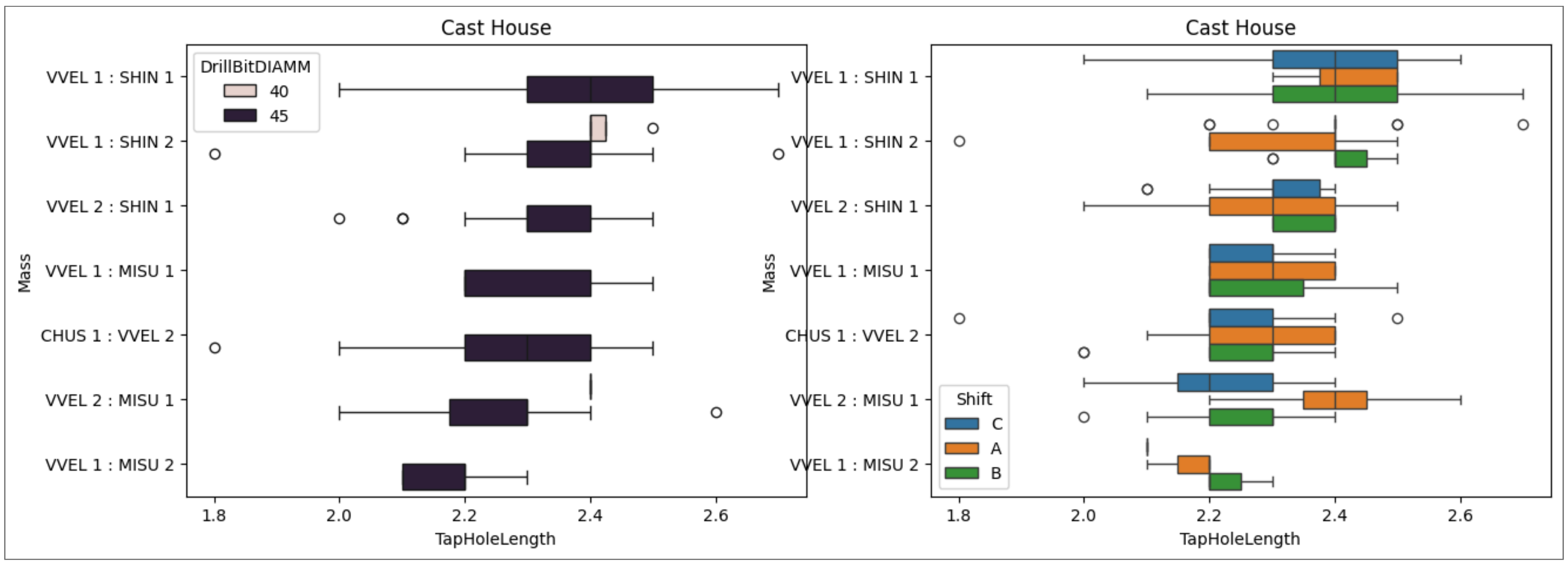
```
plt.figure(figsize=(15,7))
plt.subplot(1,2,1)
sns.kdeplot(cast_house1['TapHoleLength'], color='magenta')
plt.title("Tap Hole Length")

plt.subplot(1,2,2)
sns.boxplot(cast_house1['TapHoleLength'], color='gold')
plt.title("Tap Hole Length")
plt.suptitle(t='Data Analysis')

fig = plt.figure(figsize=(15,5))
sns.countplot(data=cast_house1, x='Mass',color='#52be80')
plt.title("Mass")

fig = plt.figure(figsize=(15,5))
plt.subplot(1,2,1)
sns.kdeplot(data=cast_house1,x="TapHoleLength",hue="Mass")
plt.title('Tap Hole Length - Mass')

plt.subplot(1,2,2)
sns.kdeplot(data=cast_house1,x="PluggingPressure",hue="Mass")
plt.title('Plugging Pressure - Mass')

plt.figure(figsize=(15,5))
plt.subplot(1,2,1)
sns.boxplot(data=cast_house1,x='TapHoleLength', y='Mass')
plt.title('Cast House')
plt.subplot(1,2,2)
sns.boxplot(data=cast_house1,x='TapHoleLength', y='Mass',hue="CastHouse")
plt.title('Cast House')

plt.figure(figsize=(15,5))
plt.subplot(1,2,1)
sns.boxplot(data=cast_house1,x='TapHoleLength', y='Mass',hue="DrillBitDIAMM")
plt.title('Cast House')
plt.subplot(1,2,2)
sns.boxplot(data=cast_house1,x='TapHoleLength', y='Mass',hue="Shift")
plt.title('Cast House')

fig = plt.figure(figsize=(15,5))
sns.heatmap(cast_house1[["TAPNO","DrillBitDIAMM","TapHoleLength","PluggingPressure"]].corr(),annot=True)
plt.title('Correlation')


plt.show()
```

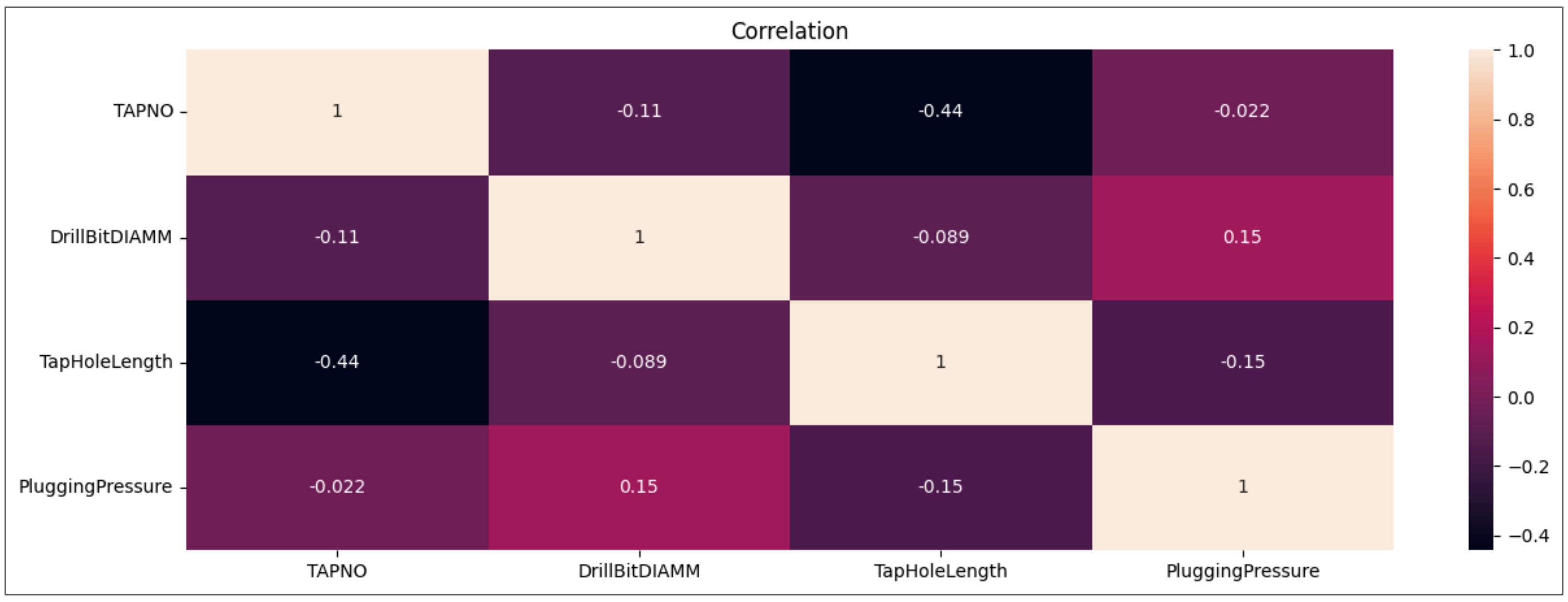

Data Analysis
Tap Hole Length
Density
TapHoleLength
Tap Hole Length
TapHoleLength


Mass
count
VVEL 1 : SHIN 1
VVEL 1 : SHIN 2
VVEL 2 : SHIN 1
VVEL 1 : MISU 1
CHUS 1 : VVEL 2
VVEL 2 : MISU 1
VVEL 1 : MISU 2
Mass


Tap Hole Length - Mass
Mass
VVEL 1 : SHIN 1
VVEL 1 : SHIN 2
VVEL 2 : SHIN 1
VVEL 1 : MISU 1
CHUS 1 : VVEL 2
VVEL 2 : MISU 1
VVEL 1 : MISU 2
Density
TapHoleLength
Plugging Pressure - Mass
Density
PluggingPressure


Cast House
Mass
TapHoleLength
Cast House
CastHouse
A
B
Mass
TapHoleLength





Conclusion

The dataset "Data1.csv" contains records of tapping operations, showcasing various parameters like tap hole length, mass, and plugging pressure. The data spans from 2024-01-01 to 2024-01-31, with multiple records per day, showcasing the tap hole length, mass, and plugging pressure for different tapping operations. The data highlights variations in these parameters, with a focus on different "VVEL" and "SHIN" types, and also "MISU" type.

Summarization

Key Parameters:

- **Date:** Represents the day when the tapping operation took place.
- **TAPNO:** Unique identifier for each tapping operation.
- **CastHouse:** Identifies the casting house where the operation was performed (A or B).
- **Shift:** Indicates the shift during which the operation took place (A, B, C, or combinations).
- **DrillBitDIAMM:** Represents the diameter of the drill bit used in the tapping operation.
- **TapHoleLength:** Indicates the length of the tap hole created.
- **Mass:** Describes the specific type of mass (e.g., "VVEL 1 : SHIN 1").
- **PluggingPressure:** Represents the pressure required to plug the tap hole.

Data Trends:

- The data suggests variations in plugging pressure depending on factors like tap hole length, mass type, and shift.
- There are different types of masses (e.g., "VVEL 1 : SHIN 1", "VVEL 2 : SHIN 1", "VVEL 1 : MISU 1") which could indicate distinct material types or casting processes.
- The data also shows variations in the plugging pressure within specific mass types, indicating potential factors like process control, material variability, or other operational parameters.

Actionable Insights

- **Plugging Pressure Analysis:** Analyze the plugging pressure data to identify patterns and correlations with other parameters like tap hole length, mass type, and shift. This analysis can reveal the impact of these factors on plugging pressure and potentially optimize the process.
- **Mass Type Comparison:** Compare the plugging pressure performance across different mass types (e.g., "VVEL 1 : SHIN 1" vs. "VVEL 2 : SHIN 1"). This comparison might highlight variations in material properties or process

control.

- **Shift Impact Assessment:** Examine how plugging pressure is affected by different shifts. This analysis can identify potential differences in operator skill, equipment maintenance, or other factors that contribute to plugging pressure.
- **Further Data Collection:** Consider collecting additional data points that could influence plugging pressure, such as casting temperature, cooling rate, and material composition. This will enhance the analysis and provide a more comprehensive understanding of the factors affecting plugging pressure.

By analyzing this dataset and incorporating further data collection as necessary, actionable insights can be generated to improve the tapping process and optimize plugging pressure for different mass types and operational conditions.

2. Data Analysis

In [12]:

```
cast_house2 = pd.read_sql_query(text("SELECT p.Date,p.Prodution, SUM(c.SGP) AS SGP, SUM(c.DSP) AS DSP FROM company.cast_house AS c LEFT JOIN
con=engine.connect())
```

In [13]:

```
cast_house2["Total SGP"]=cast_house2.loc[:,"SGP":"DSP"].sum(axis=1)
a=cast_house2["Total SGP"]
b=cast_house2["Prodution"]
n=0
for i in a:
    for j in b:
        n= a/b*1000
cast_house2["Production/Slag ratio"]=n
cast_house2
```

Out[13]:

	Date	Prodution	SGP	DSP	Total SGP	Production/Slag ratio
0	01-01-2024	1331	540.0	20.0	560.0	420.736289
1	02-01-2024	2304	930.0	140.0	1070.0	464.409722
2	03-01-2024	2324	790.0	60.0	850.0	365.748709
3	04-01-2024	2346	960.0	70.0	1030.0	439.045183
4	05-01-2024	2314	1020.0	NaN	1020.0	440.795160
5	06-01-2024	2337	980.0	40.0	1020.0	436.456996
6	07-01-2024	2346	1030.0	40.0	1070.0	456.095482
7	08-01-2024	2313	1050.0	NaN	1050.0	453.955901
8	09-01-2024	2356	1000.0	30.0	1030.0	437.181664
9	10-01-2024	2286	990.0	50.0	1040.0	454.943132
10	11-01-2024	2250	740.0	100.0	840.0	373.333333
11	12-01-2024	2314	770.0	50.0	820.0	354.364736
12	13-01-2024	2322	925.0	30.0	955.0	411.283376
13	14-01-2024	2331	880.0	50.0	930.0	398.970399
14	15-01-2024	2287	850.0	70.0	920.0	402.273721
15	16-01-2024	2268	810.0	100.0	910.0	401.234568
16	17-01-2024	2210	850.0	60.0	910.0	411.764706
17	18-01-2024	2278	900.0	70.0	970.0	425.812116
18	19-01-2024	2288	910.0	90.0	1000.0	437.062937
19	20-01-2024	985	390.0	NaN	390.0	395.939086
20	21-01-2024	2210	950.0	60.0	1010.0	457.013575
21	22-01-2024	2375	960.0	NaN	960.0	404.210526
22	23-01-2024	2375	1040.0	20.0	1060.0	446.315789
23	24-01-2024	2376	940.0	NaN	940.0	395.622896
24	25-01-2024	2380	950.0	NaN	950.0	399.159664
25	26-01-2024	2360	950.0	10.0	960.0	406.779661
26	27-01-2024	2338	800.0	90.0	890.0	380.667237
27	28-01-2024	2261	720.0	170.0	890.0	393.631137
28	29-01-2024	2358	760.0	120.0	880.0	373.197625
29	30-01-2024	2376	1070.0	NaN	1070.0	450.336700
30	31-01-2024	2322	910.0	120.0	1030.0	443.583118

In [14]:

```
cast_house2.describe()
```

Out[14]:

	Prodution	SGP	DSP	Total SGP	Production/Slag ratio
count	31.000000	31.000000	24.000000	31.000000	31.000000
mean	2242.612903	882.741935	69.166667	936.290323	417.158876
std	296.522138	147.805270	40.423483	144.302736	30.424972
min	985.000000	390.000000	10.000000	390.000000	354.364736
25%	2282.000000	805.000000	40.000000	900.000000	397.454743
50%	2322.000000	925.000000	60.000000	960.000000	411.764706
75%	2351.000000	970.000000	92.500000	1030.000000	442.189139
max	2380.000000	1070.000000	170.000000	1070.000000	464.409722

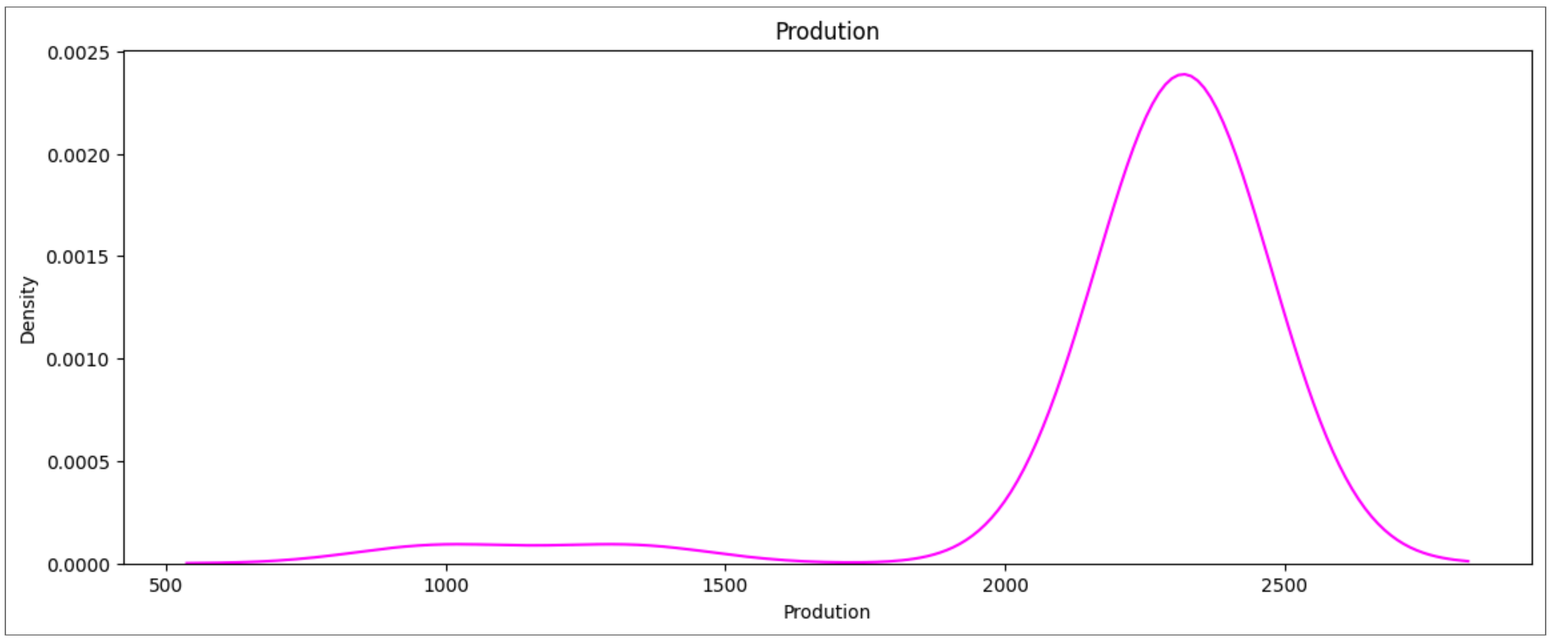
In [15]:

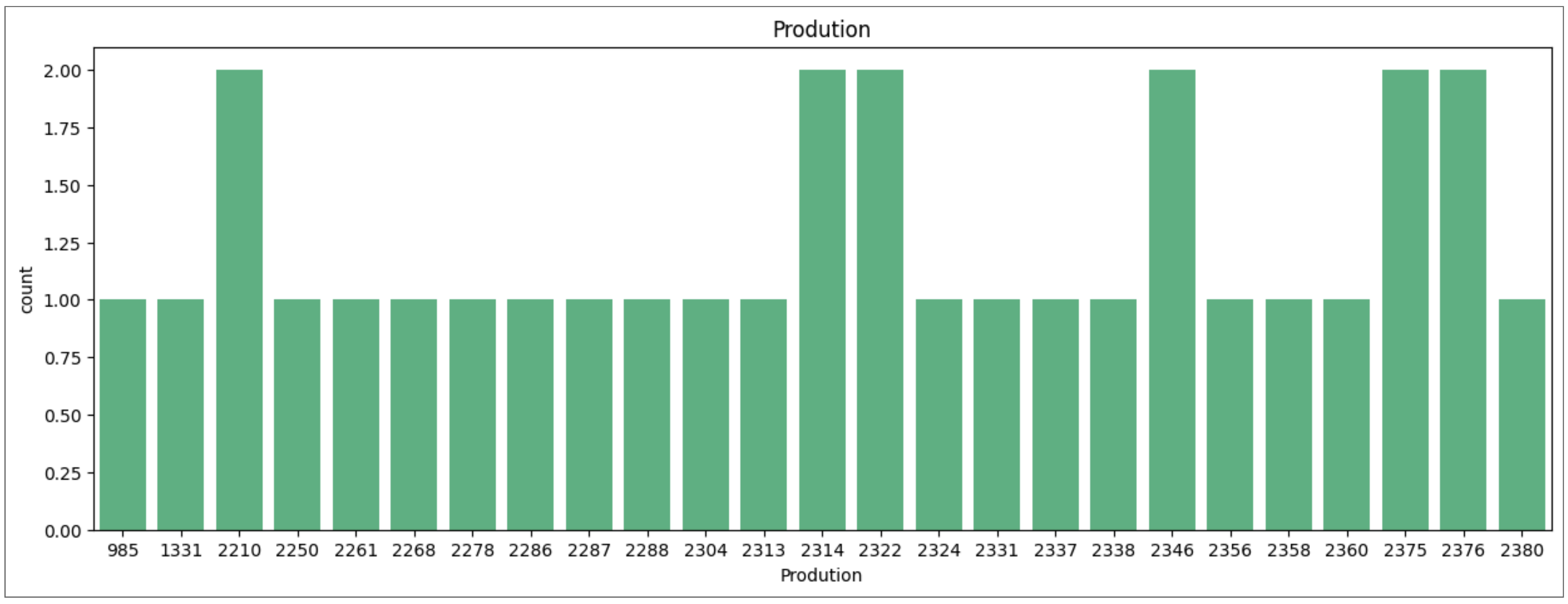
```
cast_house2.info()
```

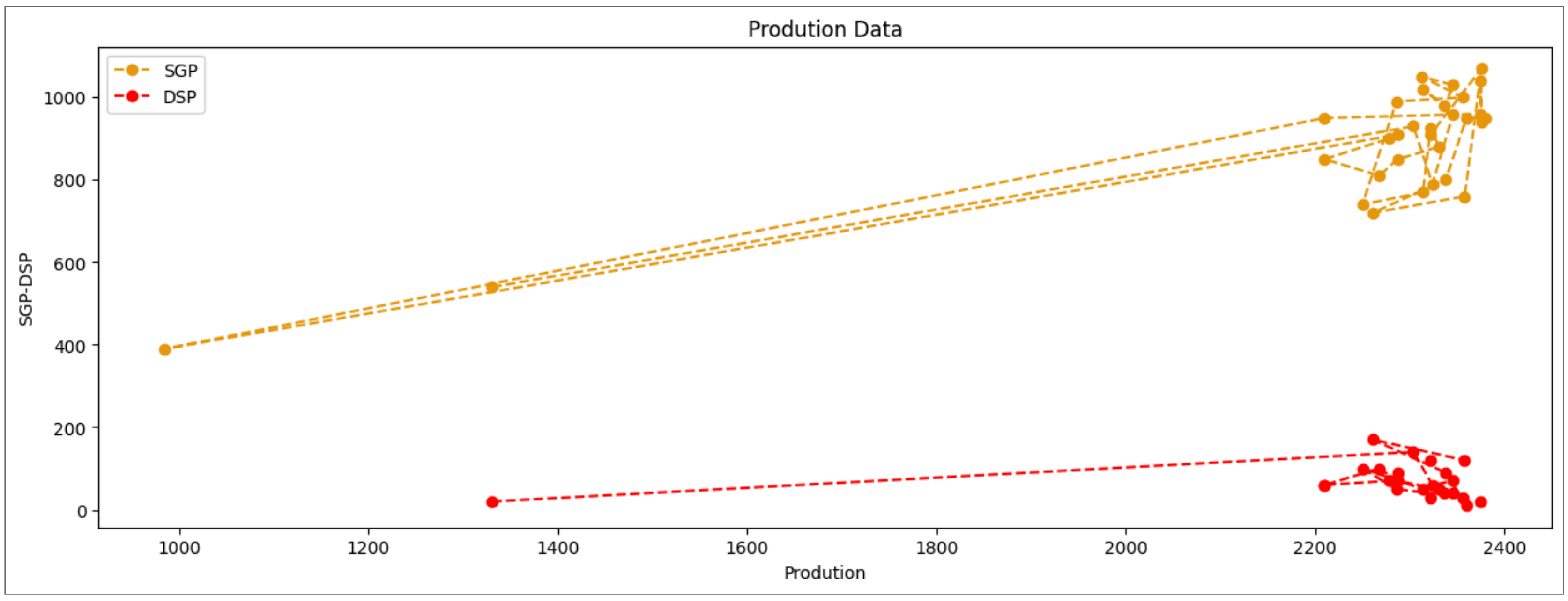
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 31 entries, 0 to 30
Data columns (total 6 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   Date                   31 non-null     object
 1   Prodution              31 non-null     int64
 2   SGP                    31 non-null     float64
 3   DSP                    24 non-null     float64
 4   Total SGP              31 non-null     float64
 5   Production/Slag ratio  31 non-null     float64
dtypes: float64(4), int64(1), object(1)
memory usage: 1.6+ KB
```

In [16]:

```
plt.figure(figsize=(30,5))
plt.subplot(1,2,1)
sns.kdeplot(cast_house2['Prodution'], color='magenta')
plt.title("Prodution")

fig = plt.figure(figsize=(15,5))
sns.countplot(data=cast_house2, x='Prodution',color='#52be80')
plt.title("Prodution")

x=cast_house2['Prodution']
y1=cast_house2['SGP']
y2=cast_house2['DSP']
plt.figure(figsize=[15,5])
plt.plot(x,y1,color="#E79A0C",label='SGP',linestyle="dashed",marker="o") #linestyle="dashed"
plt.plot(x,y2,color="red",label='DSP',linestyle="dashed",marker="o") #marker="o"
plt.xlabel("Prodution")
plt.ylabel("SGP-DSP")
plt.title("Prodution Data")
plt.legend()
plt.show()
```







Conclusion

The data presents daily records for January 2024 in a casting house, focusing on production metrics and slag quantities (SGP and DSP) along with their ratios. It shows relatively consistent production levels, with minor fluctuations, and a slight variance in slag quantities across days. The mean production of 2242.6 and mean slag of 936.3 align closely, reflecting a fairly efficient slag management process overall. The standard deviation in production and slag metrics suggests moderate consistency, though the data shows some daily variability in slag management.

Summarization

- **Production**: Daily production levels are mostly stable, with a mean around 2242.6, a minimum of 985, and a maximum of 2380. Most production days fall within the interquartile range (2282–2351).
- **Slag (SGP and DSP)**: Mean slag amounts are around 936.3, with DSP adding variability. The data shows that DSP is occasionally missing, suggesting the need for attention to capture these values for accuracy.
- **Production/Slag Ratio**: This ratio remains largely stable, with a mean of 417.2, signaling efficient slag utilization across production days.

Actionable Insights

1. **Enhance Data Accuracy**: The data contains missing values, especially for DSP, on several days. Ensuring complete data collection will improve monitoring and analysis accuracy.
2. **Optimize Low Production Days**: The significant drop in production on January 20 (985) is worth investigating. Identifying the cause could help in avoiding similar dips, improving overall productivity.
3. **Review High DSP Days**: There's a noticeable spike in DSP on days like January 28. Understanding what led to these increases could reveal opportunities to reduce slag production or improve processes.
4. **Focus on Ratios for Performance Insights**: The Production/Slag Ratio serves as a performance metric. Monitoring this closely and aiming for higher ratios will likely yield improved operational efficiency.

3. Data Analysis

In [17]:

```
cast_house3 = pd.read_sql_query(text("""select Mass,
(Case
When Mass = 'VVEL 1 : SHIN 1' Then '50% VVEL & Shinogowa'
When Mass = 'VVEL 1 : SHIN 2' Then '33% VVEL & 66% Shinogowa'
When Mass = 'VVEL 2 : SHIN 1' Then '66% VVEL & 33% Shinogowa'
When Mass = 'VVEL 1 : MISU 1' Then '50% VVEL & MISUS'
When Mass = 'VVEL 1 : MISU 2' Then '33% VVEL & 66% MISUS'
When Mass = 'VVEL 2 : MISU 1' Then '66% VVEL & 33% MISUS'
When Mass = 'CHUS 1 : VVEL 1' Then '50% Chuson & VVEL'
When Mass = 'CHUS 1 : VVEL 2' Then '33% Chuson & 66% VVEL'
When Mass = 'CHUS 2 : VVEL 1' Then '66% Chuson & 33% VVEL'
Else Mass End) as Mass_ratio,
count(Mass) from company.cast_house
group by Mass;"""),
con=engine.connect())
cast_house3
```

Out[17]:

	Mass	Mass_ratio	count(Mass)
0	VVEL 1 : SHIN 1	50% VVEL & Shinogowa	88
1	VVEL 1 : SHIN 2	33% VVEL & 66% Shinogowa	36
2	VVEL 2 : SHIN 1	66% VVEL & 33% Shinogowa	55
3	VVEL 1 : MISU 1	50% VVEL & MISUS	17
4	CHUS 1 : VVEL 2	33% Chuson & 66% VVEL	34
5	VVEL 2 : MISU 1	66% VVEL & 33% MISUS	18
6	VVEL 1 : MISU 2	33% VVEL & 66% MISUS	9

Conclusion

The data provides insights into the usage frequency and composition ratios of different material blends in casting. VVEL and Shinogowa are the most frequently used combination, suggesting their importance in the casting process. The material cost analysis indicates that Chuson and Shinogowa are the most expensive materials, while VVEL and Vesuvius offer more economical alternatives.

Summarization

- **Blend Frequency**: The "VVEL 1 : SHIN 1" blend (50% VVEL & 50% Shinogowa) is the most common, with 88 uses. Other VVEL-Shinogowa and VVEL-MISUS blends show lower, but significant, usage.
- **Cost**: Chuson and Shinogowa are the highest-cost materials at 100,000 MT each, while VVEL and Vesuvius are more affordable at 60,000 MT each.
- **Material Variety**: Blends with VVEL as the primary component appear frequently, indicating its cost-efficiency and versatility.

Actionable Insights

1. **Leverage Cost-Effective Blends**: Increase the use of blends like "CHUS 1 : VVEL 2" and "VVEL 1 : SHIN 1," which contain cost-effective VVEL. This approach balances cost while meeting production needs.
2. **Optimize Expensive Material Usage**: Use Shinogowa and Chuson strategically, perhaps reserving these materials for blends that impact product quality significantly, given their high cost.
3. **Evaluate Low-Frequency Blends**: Blends like "VVEL 1 : MISU 2" (33% VVEL & 66% MISUS) have fewer uses and might be optimized or replaced if they contribute minimally to quality or specific requirements. Streamlining these could reduce inventory and simplify operations.
4. **Monitor Blend Consistency**: Ensuring consistent use of the "VVEL 1 : SHIN 1" blend, or other frequently used combinations, may help standardize output quality and streamline material management.e purposes. Assessing their performance could identify areas where they add unique value, potentially improving product diversity.

4. Data Analysis

In [18]:

```
cast_house4 = pd.read_sql_query(text("""select Date,TAPNO,TEMP,C,S,FeO
from  process_analysis
where FeO>=1.25
order by FeO desc;
"""),
con=engine.connect())
cast_house4
```

Out[18]:

	Date	TAPNO	TEMP	C	S	FeO
0	17-01-2024	125878	1440.0	3.97	0.050	3.17
1	13-01-2024	125848	1415.0	3.18	0.067	2.89
2	02-01-2024	125751	1439.0	3.15	0.118	2.68
3	09-01-2024	125813	1482.0	4.45	0.024	2.29
4	11-01-2024	125832	1431.0	2.97	0.123	2.13
5	15-01-2024	125866	1441.0	3.44	0.086	2.10
6	16-01-2024	125876	1396.0	2.92	0.104	2.06
7	14-01-2024	125861	1408.0	3.38	0.078	2.04
8	15-01-2024	125867	1434.0	3.25	0.075	1.94
9	26-01-2024	125953	1414.0	3.88	0.049	1.92
10	14-01-2024	125860	1450.0	3.56	0.092	1.88
11	15-01-2024	125863	1441.0	3.44	0.081	1.82
12	05-01-2024	125780	1436.0	3.63	0.071	1.75
13	14-01-2024	125856	1428.0	3.81	0.052	1.65
14	14-01-2024	125859	1483.0	3.72	0.016	1.61
15	28-01-2024	125973	1413.0	3.71	0.105	1.60
16	10-01-2024	125825	1449.0	3.39	0.065	1.49
17	15-01-2024	125862	1457.0	3.46	0.090	1.47
18	05-01-2024	125781	1440.0	3.56	0.046	1.45
19	13-01-2024	125851	1448.0	3.75	0.069	1.45
20	13-01-2024	125846	1492.0	3.89	0.047	1.44
21	04-01-2024	125769	1478.0	4.31	0.019	1.43
22	11-01-2024	125834	1434.0	3.46	0.065	1.39
23	12-01-2024	125838	1446.0	3.57	0.065	1.36
24	01-01-2024	125746	1434.0	3.35	0.043	1.34
25	05-01-2024	125778	1440.0	3.67	0.060	1.30
26	10-01-2024	125826	1400.0	3.29	0.135	1.30
27	30-01-2024	125982	1429.0	3.96	0.055	1.26
28	02-01-2024	125748	1418.0	3.21	0.134	1.25

Conclusion

This dataset records various chemical properties and temperatures for tap samples taken throughout January 2024 in a cast house. The metrics—temperature (TEMP), carbon (C), sulfur (S), and iron oxide (FeO)—provide insights into process consistency and material characteristics over time. Temperatures range widely, as does the carbon content, with notable fluctuations in sulfur and FeO levels. These variations could impact material quality and production efficiency.

Summarization

- **Temperature**: The temperatures range from 1396°C to 1492°C, with a few higher values around 1480–1490°C, suggesting targeted adjustments likely for specific processes or material characteristics.
- **Carbon (C)**: Carbon levels fluctuate between 2.92% and 4.45%, with the highest levels observed at higher temperatures.
- **Sulfur (S)**: Sulfur levels vary, with some lower values (0.016–0.019) and a few peaks (above 0.1), suggesting variable sulfur content across samples.
- **Iron Oxide (FeO)**: FeO levels decline as temperature increases, with lower values around 1.3 or less at higher temperature and carbon concentrations.

Actionable Insights

1. **Optimize Temperature Ranges**: Standardizing around a specific temperature range (e.g., 1430–1450°C) could reduce variability, enhancing predictability in carbon and FeO levels.
2. **Manage Carbon Consistency**: Carbon plays a vital role in end-product quality. Regular monitoring can help maintain C levels around the mid-range (3.4–3.8%) to improve product stability.

3. **Reduce Sulfur Variance**: High sulfur levels impact product quality. Introducing stricter controls, especially when sulfur spikes above 0.1%, could enhance quality consistency.
4. **Adjust FeO Control Parameters**: The inverse relationship between FeO and temperature suggests that controlling temperature more tightly could further minimize FeO fluctuations, reducing unwanted impurities and optimizing material properties.

4. Data Analysis

In [19]:

```
cast_house_consumable = pd.read_sql_query(text("SELECT * FROM company.cast_house_consumable;"), con=engine.connect())
cast_house_consumable
```

Out[19]:

	Date	Chuson	VVEL	Shinogowa	MISUS	Vesuvius	Drill_bit_35mm	Drill_bit_40mm	Drill_bit_45mm	Drill_Rod	...	Thermocouple_tip	Ladl
0	01-01-2024	0	1	2	0	0	0	0	9	8	...	200	
1	02-01-2024	0	1	2	0	0	0	0	9	8	...	0	
2	03-01-2024	0	2	2	0	0	0	0	9	8	...	0	
3	04-01-2024	0	1	2	0	0	0	0	9	8	...	0	
4	05-01-2024	0	1	2	0	0	0	4	8	8	...	0	
5	06-01-2024	0	1	2	0	0	0	4	8	9	...	0	
6	07-01-2024	0	2	2	0	0	0	0	8	8	...	0	
7	08-01-2024	0	1	2	0	0	0	0	8	9	...	0	
8	09-01-2024	0	1	1	1	0	0	0	9	8	...	0	
9	10-01-2024	0	2	2	0	0	0	0	9	9	...	0	
10	11-01-2024	0	1	2	0	0	0	0	6	8	...	0	
11	12-01-2024	0	2	1	0	0	0	6	6	9	...	0	
12	13-01-2024	0	2	1	0	0	0	6	6	8	...	0	
13	14-01-2024	1	1	0	0	0	0	0	8	9	...	0	
14	15-01-2024	2	1	0	0	0	0	0	8	7	...	0	
15	16-01-2024	2	2	0	0	0	0	0	9	8	...	0	
16	17-01-2024	1	2	0	1	0	0	8	9	8	...	0	
17	18-01-2024	0	2	1	0	0	0	8	0	8	...	300	

	Date	Chuson	VVEL	Shinogowa	MISUS	Vesuvius	Drill_bit_35mm	Drill_bit_40mm	Drill_bit_45mm	Drill_Rod	...	Thermocouple_tip	Ladl
18	19-01-2024	0	2	2	0	0	0	0	0	8	...	0	
19	20-01-2024	0	2	1	0	0	0	0	8	8	...	0	
20	21-01-2024	0	2	2	0	0	0	0	8	8	...	0	
21	22-01-2024	0	2	2	0	0	0	8	0	8	...	0	
22	23-01-2024	0	2	2	0	0	0	8	0	8	...	0	
23	24-01-2024	0	2	2	0	0	0	0	8	8	...	0	
24	25-01-2024	0	2	0	2	0	0	0	8	8	...	0	
25	26-01-2024	0	2	0	3	0	0	0	8	8	...	0	
26	27-01-2024	0	2	1	2	0	0	0	8	8	...	0	
27	28-01-2024	0	2	0	1	0	0	0	8	8	...	0	
28	29-01-2024	0	2	0	0	0	0	0	8	8	...	0	
29	30-01-2024	0	2	0	0	0	0	0	8	8	...	0	
30	31-01-2024	0	2	1	0	0	0	0	8	8	...	0	

31 rows × 21 columns

Conclusion

The data provides a daily breakdown of materials used in the casting process, along with respective quantities and prices. The records show varying usage patterns for primary materials (Chuson, VVEL, Shinogowa, and MISUS) and supplementary items (drill bits, rods, thermocouple tips, and lancing pipes). Chuson and Shinogowa materials, along with VVEL and some specialized tools, are critical components with significant cost implications.

Summarization

1. **Daily Material Consumption**: Shinogowa and VVEL are consistently utilized, with some use of Chuson and MISUS. Supporting items like drill bits, drill rods, thermocouple tips, and various lancing pipes also see frequent usage.
2. **Cost Insights**: Key cost drivers are Chuson and Shinogowa at 100,000 MT, MISUS at 82,000 MT, and specific tools like drill bits (from 630 to 640 per piece) and thermocouple tips (50 per unit). Sand and other lower-cost materials are used in moderate quantities.
3. **Auxiliary Supplies**: Items like lancing pipes, thermocouple tips, and ladle-covering compounds are integral to operations, suggesting the need for steady replenishment.

Actionable Insights

1. **Optimize High-Cost Material Usage**: Given the high price of Chuson and Shinogowa, consider alternate blends or adjust usage to minimize reliance on these materials where feasible without compromising quality.

2. **Monitor Inventory for High-Turnover Items**: Drill rods, drill bits, and thermocouple tips are used consistently. Ensure adequate stock levels to prevent delays, and potentially negotiate bulk purchase rates to reduce costs.
3. **Track and Evaluate Sand Usage**: Although sand is a lower-cost material, it is consumed at varying rates. Monitoring its use could identify opportunities to reduce waste.
4. **Consider Bulk Purchasing for High-Usage Consumables**: Items like lancing pipes, drill bits, and pocking rods could be purchased in larger quantities to take advantage of volume discounts, which could lead to notable savings over time.

In [ ]: