

Manufacturing Downtime Analysis

Project Overview

This project focuses on analyzing production performance data from a biscuit manufacturing factory. The dataset captures key operational metrics from various machines and production lines, including: total units made, good units, stoppage types, and OEE (Overall Equipment Effectiveness) codes.

The goal is to understand how well the factory is performing, find reasons for downtime, and measure the quality of production.

The ultimate goal is to support data-driven decision-making for improving factory productivity, minimizing downtime, and optimizing production planning.

Objectives

- Check production performance
- Analyze Downtime and Stoppage Types
- Examine Production Trends Over Time
- Measure Equipment Effectiveness (OEE)
- Identify Improvement Opportunities
- Support Operational Decision-Making

▼ Data Gathering

```
# We import the libraries we will use
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib.ticker as mtick
from matplotlib.patches import FancyBboxPatch
import seaborn as sns
```

```
# We import the Data Frame
manufacturing = pd.read_excel('/content/Fact.xlsx')
```

▼ Data Assessment

```
manufacturing.head()
```

	Key_Machine_Product	Machine	Product	Total Biscuits Made	Total Good Biscuits Made	Start Date	End Date	Start Time	End Time	Stoppage Type	Total Units Made	Good Made Units	OEE Code
0	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	02:08:46	02:28:33	Major	341823.0	141.0	NO
1	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	02:32:18	02:42:34	Major	341768.0	453.0	NO

```
# We show a sample of Data Frame
manufacturing.sample(5)
```

Total

```
# We show the dimensions of the Data Farame and know Nulls
manufacturing.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8164 entries, 0 to 8163
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Machine          8159 non-null    object 
 1   Product          8159 non-null    object 
 2   Total Biscuits Made 8159 non-null    float64
 3   Total Good Biscuits Made 8159 non-null    float64
 4   Start Date       8159 non-null    datetime64[ns]
 5   End Date         8159 non-null    datetime64[ns]
 6   Start Time       8159 non-null    object 
 7   End Time         8159 non-null    object 
 8   Stoppage Type   8159 non-null    object 
 9   Total Units Made 8159 non-null    float64
 10  Good Made Units 8159 non-null    float64
 11  OEE Code         8149 non-null    object 
dtypes: datetime64[ns](2), float64(4), object(7)
memory usage: 829.3+ KB
```

```
# We show some descriptive statistics of the numerical columns
manufacturing.describe()
```

	Total Biscuits Made	Total Good Biscuits Made	Start Date	End Date	Total Units Made	Good Made Units
count	8159.000000	8159.000000	8159	8159	8.159000e+03	8159.000000
mean	1521.973404	479.400784	2021-07-20 06:54:03.044490752	2021-07-20 07:10:59.639661824	1.675374e+05	56481.249050
min	0.000000	0.000000	2021-07-01 00:00:00	2021-07-01 00:00:00	0.000000e+00	-4662.000000
25%	0.000000	0.000000	2021-07-10 00:00:00	2021-07-10 00:00:00	3.520000e+02	219.000000
50%	0.000000	0.000000	2021-07-27 00:00:00	2021-07-27 00:00:00	7.838000e+04	438.000000
75%	0.000000	0.000000	2021-07-29 00:00:00	2021-07-29 00:00:00	2.708730e+05	86514.500000
max	961128.000000	391738.000000	2021-08-01 00:00:00	2021-08-01 00:00:00	1.068855e+06	480646.000000

```
# We show some descriptive statistics of the objective columns
manufacturing.describe(include=[object])
```

	Key_Machine_Product	Machine	Product	Start Time	End Time	Stoppage Type	OEE Code
count	8159	8159	8159	8159	8159	8159	8149
unique	72	10	19	7199	7198	3	7
top	Biscuit Filling Machine _ Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	00:00:00	00:00:00	Minor	NO
freq	1249	4013	1401	98	102	5325	4072

```
# We check duplicated values
manufacturing.duplicated().sum()
```

```
np.int64(28)
```

```
# We show nulls in OEE Code colun
manufacturing[manufacturing['OEE Code'].isnull()]
```

Key_Machine_Product	Machine	Product	Total Biscuits Made	Total Good Biscuits Made	Start Date	End Date	Start Time	End Time	Stoppage Type	Total Units Made	Good Made Units	OEE Code
73	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN
114	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN
188	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN
227	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN
519	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN
1228	Biscuit Filling Machine _Chocolate cookies	Biscuit Filling Machine	Chocolate cookies	0.0	0.0	2021-07-19	2021-07-20	06:49:29	00:00:00	Major	188.0	188.0
1229	Biscuit Filling Machine _Chocolate cookies	Biscuit Filling Machine	Chocolate cookies	0.0	0.0	2021-07-20	2021-07-20	00:00:00	00:00:00	Minor	0.0	0.0
1231	Biscuit Filling Machine _Chocolate cookies	Biscuit Filling Machine	Chocolate cookies	0.0	0.0	2021-07-21	2021-07-21	00:00:00	00:00:00	Minor	0.0	0.0
1233	Biscuit Filling Machine _Chocolate cookies	Biscuit Filling Machine	Chocolate cookies	0.0	0.0	2021-07-21	2021-07-22	16:15:36	00:00:00	Major	427.0	427.0

```
print(manufacturing.value_counts('Stoppage Type'))
```

```
Stoppage Type
Minor    5325
Major     2830
minor      4
Name: count, dtype: int64
```

```
print(manufacturing.value_counts('OEE Code'))
```

```
OEE Code
NO      4072
CC      4052
Ru      9
@       8
PM      6
Noo     1
Noo     1
Name: count, dtype: int64
```

```
# We check the invalid values in Good Made Units column
manufacturing[manufacturing['Good Made Units']<0]
```

Key_Machine_Product	Machine	Product	Total Biscuits Made	Total Good Biscuits Made	Start Date	End Date	Start Time	End Time	Stoppage Type	Total Units Made	Good Made Units	OEE Code

▼ Quality Issues

- "OEE Code" column has (10) nulls.
- "Start Date" column has (1) null value.
- All colmuns have (5) null Records
- "OEE Code" column has (2) record is inconsistent (His value is 'NOo' or 'NOO' instead of 'NO')
- "Good Made Units" column has (1) negative value
- "Stoppage Type" column has (4) record is inconsistent (His value is 'minor' instead of 'Minor')
- There are (28) duplicate rows.

▼ Data Cleaning

```
#Taking a copy to do the adjustments  
manufacturing_clean = manufacturing.copy()
```

▼ Handling Issues

1. Handle the "**null**" Values in both "**OEE Code**" & "**Start Date**" columns.
2. Drop null records.
3. Validate data consistency (Replace '**NOo**' or '**NOO**' with '**NO**') in "OEE Code" column.
4. Replace (Negative values with Positive) in "Good Made Units" column.
5. Validate data consistency (Replace '**minor**' with '**Minor**') in "Stoppage Type" column.
6. Remove duplicate rows.

▼ Check Dataset Dimensions

```
manufacturing_clean.shape  
(8164, 13)
```

▼ Handle Missing Values in "OEE Code" Column

```
# Replace null values in "OEE Code" column  
manufacturing_clean['OEE Code'].fillna('Unclassified', inplace=True)  
  
Show hidden output
```

▼ Test

```
manufacturing_clean['OEE Code'].isnull().sum()  
np.int64(0)
```

▼ Identify Records with Missing Start Dates

```
# showing null values in the "start date" column
```

```
manufacturing_clean[manufacturing_clean['Start Date'].isnull()]
```

Key_Machine_Product	Machine	Product	Total Biscuits Made	Total Good Biscuits Made	Start Date		End Date		Start Time		End Time		Stoppage Type	Total Units Made	Good Made Units	OEE Code
					Start Date	End Date	Start Time	End Time	Start Time	End Time	Stoppage Type	Stoppage Type				
73	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Unclassified
114	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Unclassified
188	NaN	NaN	NaN	NaN	NaN	NaT	NaT	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Unclassified

Fill Missing Start Dates

```
# Replace null values in "Start Date" column
```

```
manufacturing_clean['Start Date'].fillna(method='ffill', inplace=True)
```

/tmp/ipython-input-4189722586.py:3: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chaine
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are sett

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df

```
manufacturing_clean['Start Date'].fillna(method='ffill', inplace=True)
```

/tmp/ipython-input-4189722586.py:3: FutureWarning: Series.fillna with 'method' is deprecated and will raise in a future version.
manufacturing_clean['Start Date'].fillna(method='ffill', inplace=True)

Test

```
manufacturing_clean['Start Date'].isnull().sum()
```

```
np.int64(0)
```

Remove All Records with Missing Values

```
# Dropping null records
```

```
manufacturing_clean.dropna(inplace=True)
```

Test

```
manufacturing_clean.isnull().sum()
```

```

0
Key_Machine_Product 0
Machine 0
Product 0
Total Biscuits Made 0
Total Good Biscuits Made 0
Start Date 0
End Date 0
Start Time 0
End Time 0
Stoppage Type 0
Total Units Made 0
Good Made Units 0
OEE Code 0

```

dtype: int64

▼ Standardize 'OEE Code' Format

```

# Consistent data (Replace "NOo" , "NOO" with "NO")

manufacturing_clean['OEE Code'] = manufacturing_clean['OEE Code'].str.replace('NOo', 'NO')
manufacturing_clean['OEE Code'] = manufacturing_clean['OEE Code'].str.replace('NOO', 'NO')

```

```

# Consistent data (Replace 0 with "Unclassified")

manufacturing_clean['OEE Code'] = manufacturing_clean['OEE Code'].str.replace("0", 'Unclassified')

```

▼ Test

```
manufacturing_clean['OEE Code'].value_counts()
```

OEE Code	count
NO	4074
CC	4052
Unclassified	18
Ru	9
PM	6

dtype: int64

▼ Data Correction: Convert Negative Values to Positive

```

# Replace (Negative values with Positive) in "Good Made Units" column

manufacturing_clean['Good Made Units'] = manufacturing_clean['Good Made Units'].abs()

```

▼ Test

```
manufacturing_clean['Good Made Units'].min()
```

0.0

▼ Standardize "Stoppage Type" Format

```
# Consistent data (Replace "minor" with "Minor")
manufacturing_clean['Stoppage Type'] = manufacturing_clean['Stoppage Type'].str.replace('minor', 'Minor')
```

▼ Test

```
manufacturing_clean['Stoppage Type'].value_counts()

count
Stoppage Type
Minor      5329
Major      2830
dtype: int64
```

▼ Remove Duplicate Rows

```
manufacturing_clean.drop_duplicates(inplace=True)
```

▼ Test

```
manufacturing_clean.duplicated().sum()
np.int64(0)
```

▼ Display Cleaned Dataset

	manufacturing_clean													
	Key_Machine_Product	Machine	Product	Total Biscuits Made	Total Good Biscuits Made	Start Date	End Date	Start Time	End Time	Stoppage Type	Total Units Made	Good Made Units	Other	
0	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	02:08:46	02:28:33	Major	341823.0	141.0	1	
1	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	02:32:18	02:42:34	Major	341768.0	453.0	1	
2	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	02:45:57	02:51:00	Major	4414.0	4414.0	1	
3	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	02:55:14	03:02:07	Major	9554.0	9554.0	1	
4	Biscuit Filling Machine _Bourbon Creams	Biscuit Filling Machine	Bourbon Creams	0.0	0.0	2021-07-16	2021-07-16	03:07:03	03:22:28	Major	20559.0	20559.0	1	

⌄ Display Dataset Information and Data Types

```
manufacturing_clean.info()

<class 'pandas.core.frame.DataFrame'>
Index: 8135 entries, 0 to 8161
Data columns (total 13 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Key_Machine_Product    8135 non-null   object  
 1   Machine              8135 non-null   object  
 2   Product               8135 non-null   object  
 3   Total Biscuits Made  8135 non-null   float64 
 4   Total Good Biscuits Made  8135 non-null   float64 
 5   Start Date            8135 non-null   datetime64[ns]
 6   End Date              8135 non-null   datetime64[ns]
 7   Start Time             8135 non-null   object  
 8   End Time               8135 non-null   object  
 9   Stoppage Type          8135 non-null   object  
 10  Total Units Made      8135 non-null   float64 
 11  Good Made Units       8135 non-null   float64 
 12  OEE Code               8135 non-null   object  
dtypes: datetime64[ns](2), float64(4), object(7)
memory usage: 889.8+ KB
```

⌄ Export Cleaned Dataset to Excel File

```
# Extracting The Dataset

manufacturing_clean.to_excel('manufacturing_clean.xlsx', index=False)
```

⌄ Download Cleaned Dataset to Local Machine