

In [1]:

```
import pickle
import pandas as pd
```

Desserialização do arquivo do modelo

In [2]:

```
with open('RF_model.pickle', 'rb') as arquivo:
    RF_model = pickle.load(arquivo)
```

Importando os dados e aplicando o modelo

In [3]:

```
tabela = pd.read_csv('desafio_manutencao_preditiva_teste.csv')
tabela.drop(['product_id', 'type'],axis=1, inplace=True)
previsao = RF_model.predict(tabela)
```

In [4]:

```
tabela.head()
```

Out[4]:

	udi	air_temperature_k	process_temperature_k	rotational_speed_rpm	torque_nm	tool_wear_min
0	446	297.5	308.6	1793	26.7	70
1	7076	300.7	310.5	1536	47.4	192
2	1191	297.2	308.4	1460	42.1	41
3	2618	299.4	309.1	1670	35.9	68
4	5067	304.1	313.1	1550	30.9	9

In [5]:

```
tabela['predictedValues'] = previsao
tabela

df_out = pd.merge(tabela,tabela[['predictedValues']],how = 'left',left_index = True, right_index = True)
df_out
```

Out[5]:

	udi	air_temperature_k	process_temperature_k	rotational_speed_rpm	torque_nm	tool_wear_min	predictedValues_x	predictedValues_y
0	446	297.5	308.6	1793	26.7	70	1	1
1	7076	300.7	310.5	1536	47.4	192	1	1
2	1191	297.2	308.4	1460	42.1	41	1	1
3	2618	299.4	309.1	1670	35.9	68	1	1
4	5067	304.1	313.1	1550	30.9	9	1	1
...
3328	5554	302.5	311.9	1306	59.7	172	1	1
3329	6961	300.7	311.0	1413	52.0	91	1	1
3330	6914	300.8	311.2	1481	38.5	181	1	1
3331	5510	302.8	312.2	1509	36.5	52	1	1
3332	3066	300.1	309.2	1687	27.7	95	1	1

3333 rows × 8 columns

In [6]:

```
df_out['predictedValues'] = df_out['predictedValues_y']
df_out['predictedValues'] = df_out['predictedValues'].replace(1, 'Sem falha')
df_out['predictedValues'] = df_out['predictedValues'].replace(0, 'Falhou')
df_out.drop(['udi', 'air_temperature_k', 'process_temperature_k', 'rotational_speed_rpm', 'torque_nm', 'tool_wear_min', 'predictedVa
```

Acima, para um melhor entendimento, eu transformo as informações dos resultados que constam apenas em 0 e 1 para Sem falha e Falha. Além disso removo as colunas para gerar o arquivo final, conforme solicitado no desafio.

In [7]:

```
df_out.value_counts()
```

Out[7]:

```
predictedValues  
Sem falha      3333  
dtype: int64
```

In [8]:

```
df_out[df_out['predictedValues']=='Falhou'].value_counts().sum() / df_out[df_out['predictedValues']=='Sem falha'].value_counts().sum()
```

Out[8]:

```
0.0
```

Dos dados testados, apenas 1,15% das máquinas apresentam potencial de falha.

Exportando o tabela com o resultado dos testes

In [9]:

```
df_out.to_csv ('predicted.csv', index=True)
```