In [1]:

```
# install the SBTi Python module
%pip install sbti-finance-tool

# Install other Python modules required to use this notebook
%pip install matplotlib==3.2.2
%pip install openpyxl==3.0.9
```

```
Requirement already satisfied: sbti-finance-tool in c:\users\ck101\anaconda3\lib\sit
e-packages (1.0.3)
Requirement already satisfied: pandas==1.3.4 in c:\users\ck101\anaconda3\lib\site-pa
ckages (from sbti-finance-tool) (1.3.4)
Requirement already satisfied: openpyxl==3.0.9 in c:\users\ck101\anaconda3\lib\site-
packages (from sbti-finance-tool) (3.0.9)
Requirement already satisfied: requests==2.26.0 in c:\users\ck101\anaconda3\lib\site
-packages (from sbti-finance-tool) (2.26.0)
Requirement already satisfied: six==1.15.0 in c:\users\ck101\anaconda3\lib\site-pack
ages (from sbti-finance-tool) (1.15.0)
Requirement already satisfied: xlrd==1.2.0 in c:\users\ck101\anaconda3\lib\site-pack
ages (from sbti-finance-tool) (1.2.0)
Requirement already satisfied: pydantic==1.8.2 in c:\users\ck101\anaconda3\lib\site-
packages (from sbti-finance-tool) (1.8.2)
Requirement already satisfied: pytz>=2017.3 in c:\users\ck101\anaconda3\lib\site-pac
kages (from pandas==1.3.4->sbti-finance-tool) (2020.1)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\ck101\anaconda3\li
b\site-packages (from pandas==1.3.4->sbti-finance-tool) (2.8.1)
Requirement already satisfied: numpy>=1.17.3; platform_machine != "aarch64" and plat
form_machine != "arm64" and python_version < "3.10" in c:\users\ck101\anaconda3\lib
\site-packages (from pandas==1.3.4->sbti-finance-tool) (1.19.2)
Requirement already satisfied: et-xmlfile in c:\users\ck101\anaconda3\lib\site-packa
ges (from openpyxl==3.0.9->sbti-finance-tool) (1.0.1)
Requirement already satisfied: charset-normalizer~=2.0.0; python_version >= "3" in
c:\users\ck101\anaconda3\lib\site-packages (from requests==2.26.0->sbti-finance-too
l) (2.0.12)
Requirement already satisfied: idna<4,>=2.5; python_version >= "3" in c:\users\ck101
\anaconda3\lib\site-packages (from requests==2.26.0->sbti-finance-tool) (2.10)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\ck101\anaconda3\lib\si
te-packages (from requests==2.26.0->sbti-finance-tool) (2020.6.20)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\ck101\anaconda3\lib
\site-packages (from requests==2.26.0->sbti-finance-tool) (1.25.11)
Requirement already satisfied: typing-extensions>=3.7.4.3 in c:\users\ck101\anaconda
3\lib\site-packages (from pydantic==1.8.2->sbti-finance-tool) (3.7.4.3)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: matplotlib==3.2.2 in c:\users\ck101\anaconda3\lib\sit
e-packages (3.2.2)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\ck101\anaconda3\lib\sit
e-packages (from matplotlib==3.2.2) (1.3.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in c:\users
\ck101\anaconda3\lib\site-packages (from matplotlib==3.2.2) (2.4.7)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\ck101\anaconda3\lib
\site-packages (from matplotlib==3.2.2) (2.8.1)
Requirement already satisfied: numpy>=1.11 in c:\users\ck101\anaconda3\lib\site-pack
ages (from matplotlib==3.2.2) (1.19.2)
Requirement already satisfied: cycler>=0.10 in c:\users\ck101\anaconda3\lib\site-pac
kages (from matplotlib==3.2.2) (0.10.0)
Requirement already satisfied: six>=1.5 in c:\users\ck101\anaconda3\lib\site-package
s (from python-dateutil>=2.1->matplotlib==3.2.2) (1.15.0)
Note: you may need to restart the kernel to use updated packages.
Requirement already satisfied: openpyxl==3.0.9 in c:\users\ck101\anaconda3\lib\site-
packages (3.0.9)
Requirement already satisfied: et-xmlfile in c:\users\ck101\anaconda3\lib\site-packa
ges (from openpyxl==3.0.9) (1.0.1)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:

```
import SBTi
from SBTi.data.excel import ExcelProvider
from SBTi.portfolio_aggregation import PortfolioAggregationMethod
from SBTi.portfolio_coverage_tvp import PortfolioCoverageTVP
```

```
from SBTi.temperature_score import TemperatureScore, Scenario, ScenarioType, Engagem
from SBTi.interfaces import ETimeFrames, EScope
import pandas as pd
%matplotlib inline
```

In [ ]:
```
# download resources for Google Colab environment
import urllib.request
import os

if not os.path.isdir("data"):
    os.mkdir("data")
if not os.path.isfile("data/data_provider_test.xlsx"):
    urllib.request.urlretrieve("https://github.com/ScienceBasedTargets/SBTi-finance-
if not os.path.isfile("data/example_portfolio_test.csv"):
    urllib.request.urlretrieve("https://github.com/ScienceBasedTargets/SBTi-finance-
if not os.path.isfile("utils.py"):
    urllib.request.urlretrieve("https://github.com/ScienceBasedTargets/SBTi-finance-
try:  # Import statement when run in Google Colabs
    from utils import collect_company_contributions, plot_grouped_statistics, anonym
        plot_grouped_heatmap, print_scenario_gain, print_grouped_scores, get_contrib
except:  # Import statement when run locally
    from examples.utils import collect_company_contributions, plot_grouped_statistic
        plot_grouped_heatmap, print_scenario_gain, print_grouped_scores, get_contrib
```

In [3]:
```
import logging
root_logger = logging.getLogger()
root_logger.setLevel("INFO")
```

In [4]:
```
provider = ExcelProvider(path="data/data_provider_test.xlsx")
```

In [5]:
```
portfolio = pd.read_csv("data/example_portfolio_test.csv", encoding="iso-8859-1")
```

In [6]:
```
portfolio.head(5)
```

Out[6]:

	company_name	company_id	company_isin	weights	investment_value	engagement_target
0	Comm	Comm56152	Comm56152	NaN	1.236666e+10	NaN
1	AUU	AUU56605	AUU56605	NaN	8.428221e+09	NaN
2	WBC	WBC56627	WBC56627	NaN	7.701580e+09	NaN
3	RRR	RRR56917	RRR56917	NaN	1.842410e+08	NaN
4	MACC	MACC56982	MACC56982	NaN	1.734706e+09	NaN

In [7]:
```
# convert the data of the portfolio
companies = SBTi.utils.dataframe_to_portfolio(portfolio)
```

In [8]:
```
# calculate the company temperature scores
time_frames = [ETimeFrames.MID]
scopes = [EScope.S1S2, EScope.S1S2S3]
fallback_score = 3.2

temperature_score = TemperatureScore(
    time_frames=time_frames,
    scopes=scopes,
    fallback_score=fallback_score
)
amended_portfolio = temperature_score.calculate(data_providers=[provider], portfolio
amended_portfolio.head()
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-8-fb1ce5eda478> in <module>
      9     fallback_score=fallback_score
     10 )
---> 11 amended_portfolio = temperature_score.calculate(data_providers=[provider], p
ortfolio=companies)
     12 amended_portfolio.head()

~\anaconda3\lib\site-packages\SBTi\temperature_score.py in calculate(self, data, dat
a_providers, portfolio)
    488         if data is None:
    489             if data_providers is not None and portfolio is not None:
--> 490                 data = utils.get_data(data_providers, portfolio)
    491             else:
    492                 raise ValueError(

~\anaconda3\lib\site-packages\SBTi\utils.py in get_data(data_providers, portfolio)
    203
    204     # Supplement the company data with the SBTi target status
--> 205     company_data = SBTi().get_sbti_targets(company_data, _make_isin_map(df_p
ortfolio))
    206
    207     # Prepare the data

~\anaconda3\lib\site-packages\SBTi\utils.py in _make_isin_map(df_portfolio)
    157     return {
    158         company_id: company[ColumnsConfig.COMPANY_ISIN]
--> 159         for company_id, company in df_portfolio[
    160             [ColumnsConfig.COMPANY_ID, ColumnsConfig.COMPANY_ISIN]
    161         ]

~\anaconda3\lib\site-packages\pandas\core\frame.py in to_dict(self, orient, into)
   1826         elif orient == "index":
   1827             if not self.index.is_unique:
-> 1828                 raise ValueError("DataFrame index must be unique for orient
='index'.")
   1829             return into_c(
   1830                 (t[0], dict(zip(self.columns, t[1:])))

ValueError: DataFrame index must be unique for orient='index'.
```

In [9]:

```
# calculate a portfolio level temperature score
temperature_score.aggregation_method = PortfolioAggregationMethod.ECOTS
aggregated_portfolio = temperature_score.aggregate_scores(amended_portfolio)
print_aggregations(aggregated_portfolio)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-9-7528f4461435> in <module>
      1 # calculate a portfolio level temperature score
      2 temperature_score.aggregation_method = PortfolioAggregationMethod.ECOTS
----> 3 aggregated_portfolio = temperature_score.aggregate_scores(amended_portfolio)
      4 print_aggregations(aggregated_portfolio)

NameError: name 'amended_portfolio' is not defined
```

In [ ]: