prepare_data.py

```python
# Micro Electro Mechanical Systems
# ##############################################################################

# This python script prepares the data for further processing.

# Authors:
# Christopher Mahn
# Parastoo Oghlansheikhha
# Lukas-René Schulz
# Silas Teske
# Joshua Wolf
# Hajar Zare

# ##############################################################################

# Import of Libraries
# ------------------------------------------------------------------------------

# import math as m
# import string as st
# import random as r
# import re
import os


# ------------------------------------------------------------------------------
# Debugging-Settings

verbose = False  # Shows more debugging information


# Functions
# ------------------------------------------------------------------------------

def convert_data(input_filename, output_filename):
    if(verbose):
        print(f'[Info] Converting {input_filename}')
    """
    This function takes the data from the file and converts it in a standard
    format for later processing.

    Args:
        input_filename (str): name of the file that will be converted
        output_filename (str): the name of the output files the data will be
                               split and converted into. The different files
                               will be appended with "_01", "_02" and so on.
    """

    # - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    # Import of measurements

    if(verbose):
        print(f'[Info] Opening file "{input_filename}"', end="\r")
    with open(os.path.join("data", input_filename)) as file:
        if(verbose):
            print(f'[Info] Reading file "{input_filename}"', end="\r")
        data = file.readlines()
    if(verbose):
        print(f'[Info] Read file "{input_filename}" successfully')

    # - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

    # Converting strings to integers and floats

    for i, e in enumerate(data):
        try:
            data[i] = e.split(';')
            for j, e in enumerate(data[i]):
                if(verbose):
                    print(f'[Info][{i+1}/{len(data)}][{j+1}/{len(data[i])}] Converting entries', end="\
  r")
                if(j==0):
                    data[i][j] = int(e.strip())
                else:
                    data[i][j] = float(e.strip())
        except(ValueError):
            if(verbose):
                print(f'[Info] Detected file header at line {i+1}{20*" "}')
    if(verbose):
```

```
080         print("")
081 
082     # - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
083 
084     # Splitting data into multiple lists for individual measurements
085 
086     measurements = []
087 
088     for i, e in enumerate(data):
089         if(verbose):
090             print(f'[Info][{i+1}/{len(data)}] Splitting measurements', end="\r")
091         if(e[0]=="TIME"):
092             measurements.append([])
093         else:
094             measurements[-1].append(e)
095     if(verbose):
096         print("")
097 
098 
099     # - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
100 
101     # Export of converted measurements
102 
103     # Opening files for writing
104     for i, measurement in enumerate(measurements):
105         if(verbose):
106             print(f'[Info][{i+1}/{len(measurements)}] Opening file "{output_filename}_{i+1:02d}.csv"')
107         with open(os.path.join("data", f'{output_filename}_{i+1:02d}.csv'), "w") as file:
108 
109             # Writing the data
110             for j, e in enumerate(measurement):
111                 if(verbose):
112                     print(f'[Info][{i+1}/{len(measurements)}][{j+1}/{len(measurement)}] Writing file
      "{output_filename}_{i+1:02d}.csv"', end="\r")
113                 file.write(f'{e[0]}; {e[1]}; {e[2]}; {e[3]}; {e[4]}; {e[5]}; {e[6]}\n')
114             if(verbose):
115                 print("")
116 
117 
118 # Classes
119 # ----------------------------------------------------------------------------
120 
121 
122 # Beginning of the Programm
123 # ----------------------------------------------------------------------------
124 
125 if __name__ == '__main__':
126     convert_data("rotation_groupE.txt", "data_turntable")
127     convert_data("track_groupE.txt", "data_track")
```