

imu-convert.py

```
001 # Parsen und konvertieren von IMU-Daten
002 # #####
003
004 # Authors:
005 # Joshua Wolf
006 # Silas Teske
007 # Lasse Zeh
008 # Christopher Mahn
009
010 # #####
011
012 # Import of Libraries
013 # -----
014
015 # import math as m
016 # import string as st
017 # import random as r
018 import numpy as np
019 import os
020
021
022 # -----
023 # Debugging-Settings
024
025 verbose = False # Shows more debugging information
026
027
028 # Functions
029 # -----
030
031
032 # Classes
033 # -----
034
035
036 # Beginning of the Programm
037 # -----
038
039 if __name__ == '__main__':
040     print("Running imu-convert.py...")
041
042     # Import der Messwerte des Neigungssensors
043     file = open(os.path.join("data", "imu_data.txt"))
044     data = file.readlines()
045     file.close()
046
047     # Select active lines
048     data_split = []
049     for i, e in enumerate(data):
050         if(e[0] != "#"):
051             data_split.append(e.strip().split(" "))
052
053     # Select only marked lines
054     raw_markers = []
055     for i in data_split:
056         if(i[0] != "0000"):
057             raw_markers.append(i)
058
059     # Convert lines in float-lists
060     markers = []
061     for i, e in enumerate(raw_markers):
062         markers.append([])
063         for j in e:
064             if(j != ""): # Skip empty strings
065                 markers[i].append(float(j))
066
067     # Convert units
068     for i, e in enumerate(markers):
069         if(len(e) == 12):
070             markers[i][0] = int(e[0]) # marker-id to integer
071             markers[i][1] = e[1]
072             markers[i][2] = e[2] / 180 * np.pi # convert °/s to rad/s
073             markers[i][3] = e[3] / 180 * np.pi # convert °/s to rad/s
074             markers[i][4] = e[4] / 180 * np.pi # convert °/s to rad/s
075             markers[i][5] = e[5]
076             markers[i][6] = e[6]
077             markers[i][7] = e[7]
078             markers[i][8] = e[8] / 180 * np.pi # convert ° to s
079             markers[i][9] = e[9] / 180 * np.pi # convert ° to s
080             markers[i][10] = e[10] / 180 * np.pi # convert ° to s
```

```
081         markers[i][11] = e[11] - 273.15 # convert °K to °C
082
083 # Export
084 file = open(os.path.join("data", "imu_data_converted.txt"), "w")
085 for i in markers:
086     for j, e in enumerate(i):
087         if(j == 0):
088             file.writelines(f"{e}")
089         else:
090             file.writelines(f"; {e}")
091     file.writelines(f"\n")
092 file.close()
```