```python
"""test_communication.py | Robin Forestier | 16.02.2022

This file is used for communication test with the server flask.
The programme send the current time (13:46), the temperature of the camera (054) for 54°C and a bool value 1 or 0
(1 is for entry).
"""

# `time` is a module that contains a lot of functions to work with time.
# `datetime` is a module that contains a lot of functions to work with date and time.
import time
from datetime import datetime

# `requests` is a module that allows to send HTTP requests.
import requests

while True:

    # `t` is a variable that contains the current time.
    t = datetime.now()
    # `strftime` is a function of the `datetime` module. It allows to convert a date into a string.
    current_time = t.strftime("%H:%M")

    # Take the temperature of the RPi
    # Error also temp is 0
    try:
        with open('/sys/class/thermal/thermal_zone0/temp', 'r') as ftemp:
            temp = int(int(ftemp.read()) / 1000)
    except OSError:
        temp = 0

    # Creating a string with the current time and the temperature of the camera.
    data = "{}{:03d}1".format(current_time, temp)
    data_length = len(data)
    data = {'data': '$,RPWCSD,{:03d},{},0*'.format(data_length, data)}

    print(data)

    try:
        # Sending the data to the server (change the ip to your server IP).
        r = requests.post("http://172.16.32.27/camera", data=data, timeout=0.5)
        # Checking if the status code is bigger than 299.
        if r.status_code > 299:
            print("[Error] Communication error")
        else:
            # Reading the data send by the server.
            data = r.text
            # if the data is a correct trame ($,...,*)
            if data[0] == "$" and data[::-1][0] == "*":
                data = data.split(',')

                # Communication OK
                if data[1] == "RPWCOK":
                    print("ok")
                # Communication Error
                if data[1] == "RPWCER":
                    print("[ERROR] The cam had send a bad trame.")

    # This is a way to catch all the exceptions that can occur when you try to send a request to the server.
    except requests.exceptions.RequestException as e:
        print(e)

    time.sleep(10)

```