JSON data	Mapping Data	Explanation
This is the JSON message from MQTT Broker The message contains the sensors and their value	''' Fronius to MQTT Gateway, release 2.0.0 '''	
The message example is from a Fronius Inverter	KEY_TOPIC = 'topic' KEY_PAYLOAD = 'payload' KEY_RETAIN = 'retain' KEY_QOS = 'qos' retain = True qos = 0	These are constants in the Python program and must not be changed.
	# Send discovery topics def send_discovery_topics():	This is part of the python program, do not change id.

	device_payload = {
	'identifiers': [f"{'Fronius Data Manager'}"],
	'manufacturer': "Fronius, programmed by Per Rose",
	'model':'IG Plus 60 V-2',
	'name': 'Solar_Power',
	'sw_version': '2.0.0'
	}
		This is the description of the MQTT main sensor
		
		text in quotes can be changed to your MQTT main sensors information....
		
		Do not change anything outside the quotes.

	entity_payloads = {
	'sensorname': {
	'name': 'Fronius',
	'unit_of_meas': ""
	},
		Here starts the mapping of trhe sensors
		
		This is sensor name, you can change it from Fronius to anything you like concerning your own sensor.

{
"timeStamp": {
"timeStamp": "2020-09-28 17:22:43"
},
	'timeStamp': {
	'name': 'Fronius_Date',
	'unit_of_meas': "",
	'icon': 'mdi:calendar'
	},
		This the mapping of the first data element in the MQTT message.
		Be sure to replace Fronius_ with a common name for all the sensors in the message.
		Unit of Measurement - None for name
		Icon can be changed to the Icon of your likings from MDI icons.

"FAC": { "FAC": "50" },	'FAC': { 'name': 'Fronius_FAC', 'unit_of_meas': "Hz", 'icon': 'mdi:metronome-tick', },	So now it is just plain mapping of the sensor names into the Python program. Remember the Sensor_ in front of the mapped sensors
"IAC": { "IAC": "1.47" },	'IAC': { 'name': 'Fronius_IAC', 'unit_of_meas': "A", 'icon':'mdi:alpha-a-circle-outline' },	Unit of measurement (unit_of_meas), can be anything like C for temperature, Hz for Hertz, V for Volt, etc. etc. etc. You find the icons you want by searching in https://materialdesignicons.com/ for the icon you find suitable for the sensor
"IDC": { "IDC": "0.69" },	'IDC': { 'name': 'Fronius_IDC', 'unit_of_meas': "A", 'icon':'mdi:alpha-a-circle-outline' },	
"PAC": { "PAC": "264" },	'PAC': { 'name': 'Fronius_PAC', 'unit_of_meas': "V", 'icon':'mdi:alpha-v-circle-outline' },	
"TOTAL_ENERGY": { "TOTAL_ENERGY": "45173.00" },	'TOTAL_ENERGY': { 'name': 'Fronius_TOTAL_ENERGY', 'unit_of_meas': "kWh", 'icon':'mdi:solar-power' },	
"UAC": { "UAC": "236" },	'UAC': { 'name': 'Fronius_UAC', 'unit_of_meas': "V", 'icon':'mdi:alpha-v-circle-outline' },	
"UDC": { "UDC": "343" },	'UDC': { 'name': 'Fronius_UDC', 'unit_of_meas': "V", 'icon':'mdi:alpha-v-circle-outline' },	
"Day_Energy": { "Day_Energy": "4.99" },	'Day_Energy': { 'name': 'Fronius_Day_Energy', 'unit_of_meas': "kW", 'icon':'mdi:solar-power' },	
"YEAR_ENERGY": { "YEAR_ENERGY": "5324.00" },	'YEAR_ENERGY': { 'name': 'Fronius_YEAR_ENERGY', 'unit_of_meas': "kW", 'icon':'mdi:solar-power' },	
"Code": { "Code": "0" },	'code': { 'name': 'Fronius_code', 'unit_of_meas': "", 'icon':'mdi:code-parentheses-box' },	
"Reason": { "Reason": "" },	'reason': { 'name': 'Fronius_reason', 'unit_of_meas': "", 'icon':'mdi:message-alert-outline' },	
"UserMessage": { "UserMessage": "" },	'UserMessage': { 'name': 'Fronius_UserMessage', 'unit_of_meas': "", 'icon':'mdi:message-alert-outline' },	
"MONTH_ENERGY": { "MONTH_ENERGY": "5328.99" },	'MONTH_ENERGY': { 'name': 'Fronius_MONTH_ENERGY', 'unit_of_meas': "kWh", 'icon':'mdi:solar-power' },	
"Jan": { "Jan": "87.00" },	'Jan': { 'name': 'Fronius_Jan', 'unit_of_meas': "kW", 'icon':'mdi:solar-power' },	
"Feb": { "Feb": "192.00" },	'Feb': { 'name': 'Fronius_Feb', 'unit_of_meas': "kW", 'icon':'mdi:solar-power' },	

```
"Mar": {
  "Mar": "540.00"
},
"Apr": {
  "Apr": "808.00"
},
"Maj": {
  "Maj": "903.00"
},
"Jun": {
  "Jun": "846.00"
},
"Jul": {
  "Jul": "-3377.00"
},
"Aug": {
  "Aug": "0.00"
},
"Sep": {
  "Sep": "452.00"
},
"Okt": {
  "Okt": "301.00"
},
"Nov": {
  "Nov": "91.00"
},
"Dec": {
  "Dec": "67.00"
},
"lastTwelve": {
  "lastTwelve": "910.00"
},
```

```
 'Mar': {
   'name': 'Fronius_Mar',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Apr': {
   'name': 'Fronius_Apr',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Maj': {
   'name': 'Fronius_Maj',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Jun': {
   'name': 'Fronius_Jun',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Jul': {
   'name': 'Fronius_Jul',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Aug': {
   'name': 'Fronius_Aug',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Sep': {
   'name': 'Fronius_Sep',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Okt': {
   'name': 'Fronius_Okt',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Nov': {
   'name': 'Fronius_Nov',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'Dec': {
   'name': 'Fronius_Dec',
   'unit_of_meas': "kW",
   'icon':'mdi:solar-power'
 },
 'lastTwelve': {
   'name': 'Fronius_lastTwelve',
   'unit_of_meas': "kW",
}
```

And so it continues untill there are no more mqtt sensors

```
for entity, entity_payload in entity_payloads.items():
    entity_payload['val_tpl'] = f"{{{{ value_json.now.{entity} }}}}"
    entity_payload['uniq_id'] = f"{'Fronius_'}{entity}"
    entity_payload['stat_t'] =  f"{'Fronius'}/{'online'}"
    entity_payload['dev'] = device_payload
    sensor_type = ("sensor")
    entity_topic = f"{'homeassistant'}/{sensor_type}/{'Fronius_'}/{entity}/config"
```

This is a programming routine to update the sensors, the first part identify each sensor, build the necessary information for Home Assistant MQTT Autodiscover, and hand it over the the Home Assistant MQTT Publish.

NOT TO BE CHANGED

```
    """Publish data to MQTT broker."""
    service_data = {
    KEY_TOPIC: entity_topic,
    KEY_PAYLOAD: str(entity_payload).replace("'", '"').replace("^", "'"),
    KEY_RETAIN: retain,
    KEY_QOS: qos,
    }

    hass.services.call("mqtt", "publish", service_data)  # noqa: F821
```

This is the actual forwarding of the Autodiscover information to Home Assistant

NOT TO BE CHANGED

```
send_discovery_topics()
```

This is the program start --- **NOT TO BE CHANGED**