SDET Take Home Instructions

Implement a program that pretends to be a SimpliSafe base station. The program has the following requirements:

1. It should be configured via basic command line options, to set things like:
   a. The system environment: Production, QA, Staging.
   b. The system serial # to send as.
   c. The "password" for the system to use when authorizing the API call
2. It should support passing in a "script" file that provides automatic driving of sensor data.
   a. The script format is defined below.
3. **Bonus:** If a script file is not provided, it should enter a command-line read-eval-print loop (REPL) mode that allows for "manual" sending of sensor events. The command language is up to you, but the program should "tell" the user how to structure input.

You should time box the implementation to approximately 2 hours total effort, but you will have up to 24 hours to return your solution to us. Please implement it in a mainstream programming language — we are comfortable reviewing Python, Ruby, C#, Java, JavaScript, TypeScript, and PHP solutions.

Script Format

The automated driving script is a YAML file with a series of actions defined, as well as "delays" between events. The overall file looks like this:

```
- action: report
  sensor_type: <sensor type>
  sensor_serial: <a string of 12 hex digits>
  event: <sensor event type>
  data: <optional data for the event>
- action: delay
  time: 00:30:00
```

A sample file will be provided for you to use. Please use a 3rd party library to parse YAML—writing a parser is not part of the exercise! 🙂

Sensors

You should implement support for the following sensors and events:

- `entry` sensor, which should have `open` and `close` events.
- `motion` sensor, which should have a `motion_detected` event.
- `glassbreak` sensor, which should have a `detected` event.
- `freeze` sensor, which should have a `trigger` event and `data` that indicates whether the high/low limit was reached, and the current temperature.
- `water` sensor, which should have a `trigger` event.
- `panic` sensor, which should have a `trigger` event.
- `key_fob` sensor, which should have `off`, `home`, `away`, and `trigger` events.
- `smoke` sensor, which should have a `trigger` event with data indicating which condition was detected (`smoke`, `co`, or `heat`), as well as a corresponding `cleared` event.

The API

For each action run (either via the script or command line), send sensor data to the following *fake* URL: https://bsapi.{env}.simplisafe.com/v1/basestation/{serial}/event. The API call should send any relevant sensor information for the event, formatted as JSON. For authorization, send the system "password" as a `Bearer` token.

We recommend you use a tool like https://requestb.in for testing — the fake URL above does not resolve, and will not be useful.