

Christmas Cheer Laser, part 10

Here is the API. The commands, except for gas, are easy GET statements given in the web page. The request we were given is a GET request, so we can cut and paste most of the API calls.

```
(Invoke-WebRequest -Uri http://localhost:1225/).RawContent
```

```
-----
Christmas Cheer Laser Project Web API
-----

Turn the laser on/off:
GET http://localhost:1225/api/on
GET http://localhost:1225/api/off

Check the current Mega-Jollies of laser output
GET http://localhost:1225/api/output

Change the lense refraction value (1.0 - 2.0):
GET http://localhost:1225/api/refraction?val=1.0

Change laser temperature in degrees Celsius:
GET http://localhost:1225/api/temperature?val=-10

Change the mirror angle value (0 - 359):
GET http://localhost:1225/api/angle?val=45.1

Change gaseous elements mixture:
POST http://localhost:1225/api/gas
POST BODY EXAMPLE (gas mixture percentages):
O=5&H=5&He=5&N=5&Ne=20&Ar=10&Xe=10&F=20&Kr=10&Rn=10
-----
```

The values we've collected, at considerable effort, are:

`angle?val=65.5`

`temperature?val=-33.5`

```
$correct_gases_postbody = @{'n O=6`n H=7`n He=3`n N=4`n Ne=22`n
Ar=11`n Xe=10`n F=20`n Kr=8`n Rn=9`n}
```

`refraction?val=1.867`

Since the gaseous elements use a POST request, there needs to be two extra parameters in the Invoke-Web request. We need `-Method POST` to say it is a POST request, and `-Body` to hold the data we want to POST.

```
-Body [contents of body]
```

It is easiest to put the body contents into a variable and then put `-Body $Body` into the request along with `-Method POST`. We are given two example formats for the body data. These values are in JSON format but corrected so that they will work with Invoke-Web.

```
$Body = @{'O=6;H=7;He=3;N=4;Ne=22;Ar=11;Xe=10;F=20;Kr=8;Rn=9}
```

The gases could also be written this way, still in JSON format. It is a PowerShell hash table, roughly equivalent to a Python dictionary.

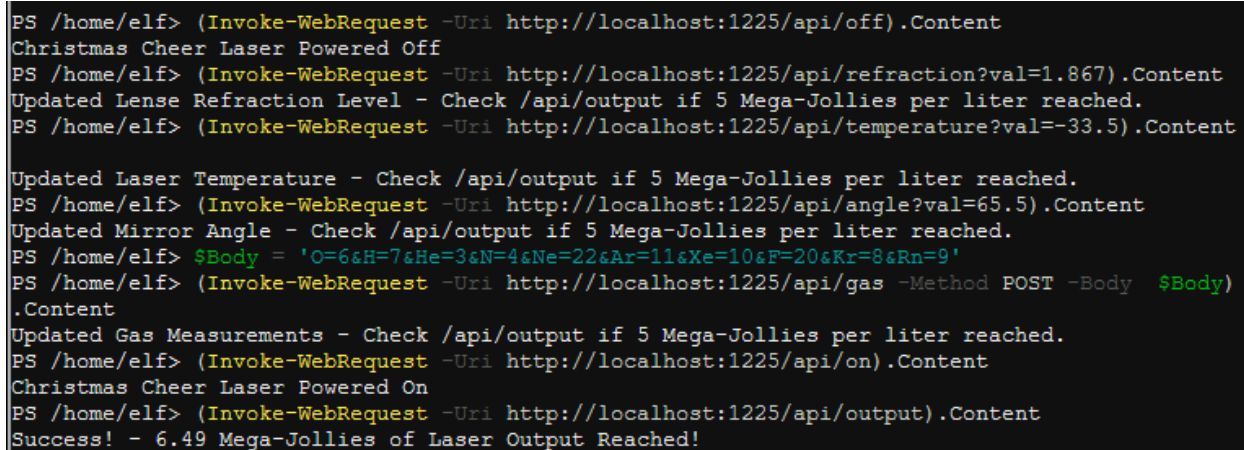
```
$Body = @{
    O=6
    H=7
    He=3
    N=4
    Ne=22
    Ar=11
    Xe=10
    F=20
    Kr=8
    Rn=9
}
```

The example on the laser status page is in the text/application format.

`$Body = 'O=6&H=7&He=3&N=4&Ne=22&Ar=11&Xe=10&F=20&Kr=8&Rn=9'` (correct values and quotes inserted)

Either of the three versions of `$Body` should work. I'll use the last. I've also replaced `RawContent` with `Content` so the screen shot will be smaller.

```
(Invoke-WebRequest -Uri http://localhost:1225/api/off).Content
(Invoke-WebRequest -Uri
http://localhost:1225/api/refraction?val=1.867).Content
(Invoke-WebRequest -Uri http://localhost:1225/api/temperature?val=-
33.5).Content
(Invoke-WebRequest -Uri http://localhost:1225/api/angle?val=65.5).Content
$Body = 'O=6&H=7&He=3&N=4&Ne=22&Ar=11&Xe=10&F=20&Kr=8&Rn=9'
(Invoke-WebRequest -Uri http://localhost:1225/api/gas -Method POST -Body
$Body).Content
(Invoke-WebRequest -Uri http://localhost:1225/api/on).Content
(Invoke-WebRequest -Uri http://localhost:1225/api/output).Content
```



```
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/off).Content
Christmas Cheer Laser Powered Off
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/refraction?val=1.867).Content
Updated Lense Refraction Level - Check /api/output if 5 Mega-Jollies per liter reached.
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/temperature?val=-33.5).Content
Updated Laser Temperature - Check /api/output if 5 Mega-Jollies per liter reached.
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/angle?val=65.5).Content
Updated Mirror Angle - Check /api/output if 5 Mega-Jollies per liter reached.
PS /home/elf> $Body = 'O=6&H=7&He=3&N=4&Ne=22&Ar=11&Xe=10&F=20&Kr=8&Rn=9'
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/gas -Method POST -Body $Body)
.Content
Updated Gas Measurements - Check /api/output if 5 Mega-Jollies per liter reached.
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/on).Content
Christmas Cheer Laser Powered On
PS /home/elf> (Invoke-WebRequest -Uri http://localhost:1225/api/output).Content
Success! - 6.49 Mega-Jollies of Laser Output Reached!
```

6.49 Mega-Jollies of Laser Output Reached! WooHoo! Note: The value you get is random, anything over 5 is good. Done at last.