Making requests

Now you've added your spreadsheet to Sheety, your API is ready to go!

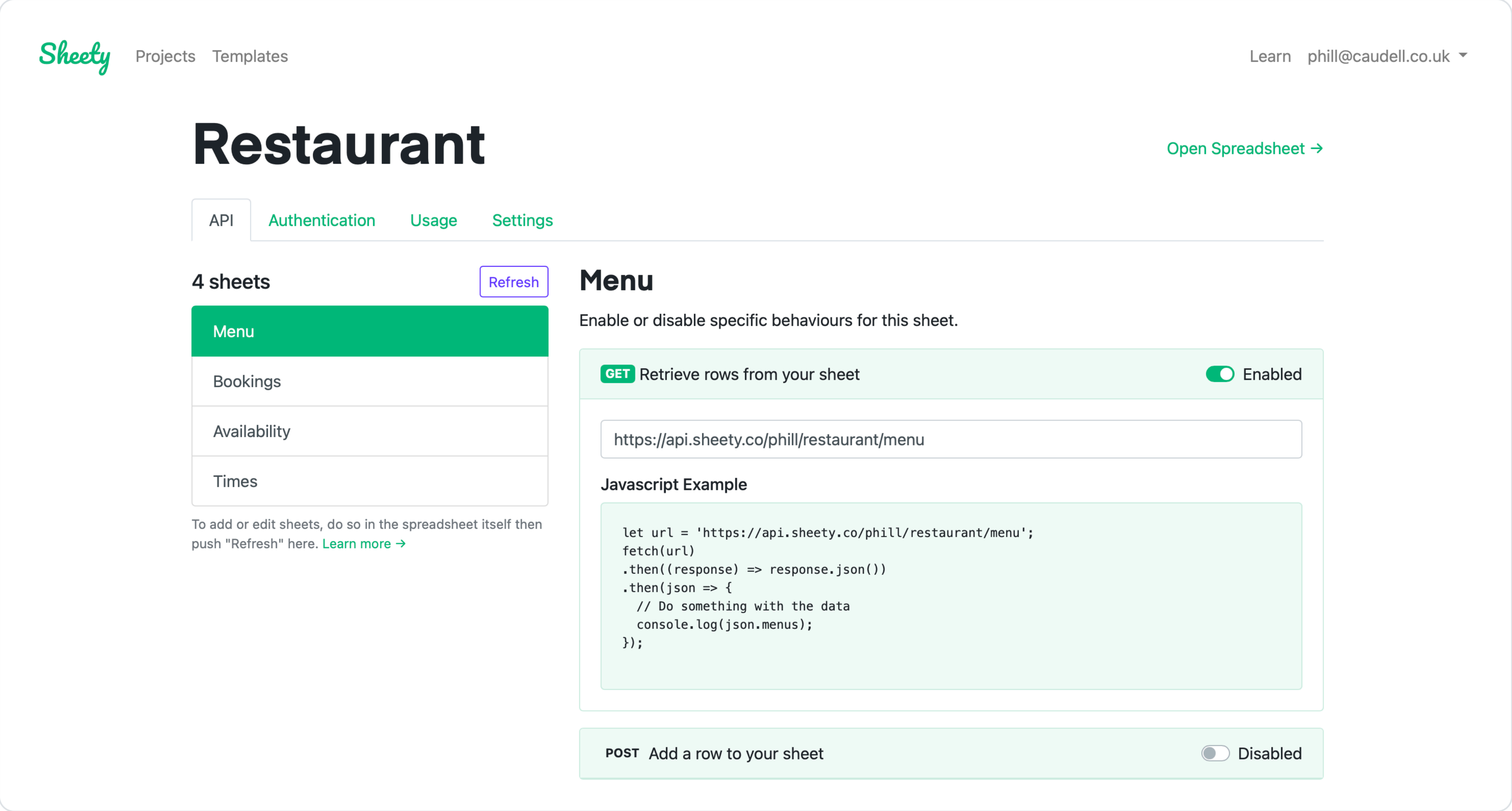
Sheety turns your spreadsheet into something called a Restful JSON API. It means you can access your spreadsheets data in a standard way using simple URLs and HTTP requests.

URL Structure

Sheety API URLs are made up three core components: username, project name, and sheet name.

- https://api.sheety.co/`username`/`projectName`/`sheetName`

💡 **Sheety Tip:** You can find the exact URLs for your sheet in the Dashboard.



Restaurant

4 sheets

To add or edit sheets, do so in the spreadsheet itself then push "Refresh" here. Learn more →

Menu

Enable or disable specific behaviours for this sheet.

GET Retrieve rows from your sheet — Enabled

https://api.sheety.co/phill/restaurant/menu

Javascript Example

```
let url = 'https://api.sheety.co/phill/restaurant/menu';
fetch(url)
.then((response) => response.json())
.then(json => {
  // Do something with the data
  console.log(json.menus);
});
```

POST Add a row to your sheet — Disabled

Getting rows

To retrieve rows from your sheet, perform a `GET` request to the endpoint. This will return all records in your sheet.

- `https://api.sheety.co/phill/themeParks/parks`

If you want to return a specific record, append the object ID (which is just the row number) to the endpoint URL.

- `https://api.sheety.co/phill/themeParks/parks/2`

Filtering rows

Sheety supports basic row filtering by using the `filter` query string.

Simply add `?filter[property]=value` to the end of the URL (where property is the name of the property to filter on, and value is the value you'd like to match).

For example to show all the family friend theme parks, the filter query string would look like:

- `https://api.sheety.co/phill/themeParks/parks?filter[familyFriendly]=true`

You can stack multiple filters by joining them with a `&`, like so:

`?filter[familyFriendly]=true&filter[country]=UK`

Add a row

To create a new row in your sheet, perform a `POST` request to the endpoint, with your row contents as a JSON payload in the request body.

Sheety expects your record to be nested in a singular root property named after your sheet. For example if your endpoint is named emails, nest your record in a property called email.

> 💡 **Sheety Tip:** Sheety camelCases all JSON keys, so a header named "First Name" will become "firstName".

POST: `https://api.sheety.co/phill/myWebsite/emails`

```
{
  "email": {
        "name": "Syed K",
        "email": "syed@gmail.com"
  }
}
```

> 💡 **Sheety Tip:** Don't forget to set the "Content-Type" header to "application/json" in your requests. Examples are included in the Dashboard.

Edit a row

Editing a row works by doing a `PUT` request to the row endpoint, with the changes you want to make as a JSON payload in the request body.

PUT: `https://api.sheety.co/phill/myWebsite/emails/2`

```
{
  "email": {
        "name": "Syed C",
        "email": "syed@gmail.com"
  }
}
```

Delete a row

To delete a row, perform a `DELETE` request to the row endpoint.

DELETE: `https://api.sheety.co/phill/myWebsite/emails/2`

And that’s it!

Hopefully you’ve now been able to retrieve and edit data in your sheet. That’s the basics of Sheety. Next up look at more advance options like API authentication.

API Authentication →

Docs

Prepare your spreadsheet

Add your spreadsheet

Making requests

Authentication

FAQs

Billing

Troubleshooting