

Challenge #1

```
class HttpError extends Error {
  constructor(response) {
    super(`${response.status} for ${response.url}`);
    this.name = 'HttpError';
    this.response = response;
  }
}

function loadJson(url) {
  return fetch(url)
    .then(response => {
      if (response.status == 200) {
        return response.json();
      } else {
        throw new HttpError(response);
      }
    });
}

function demoGithubUser() {
  let name = prompt("Enter a name?", "vanimar");
  return loadJson(`https://api.github.com/users/${name}`).then(user => {
    alert(`Full name: ${user.name}.`);
    return user;
  })
  .catch(err => {
    if (err instanceof HttpError && err.response.status == 404) {
      alert("No such user, please reenter.");
      return demoGithubUser();
    } else {
      throw err;
    }
  });
}

demoGithubUser();
```