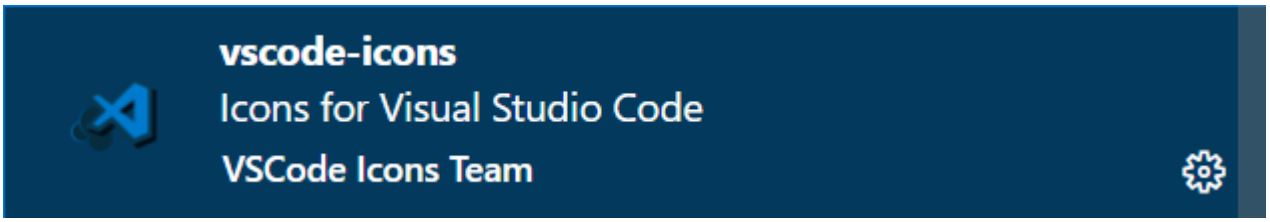
420-KBG : Atelier-1 (Préparé par Nicolas Chourot)

Objectifs

- S'approprier des modules de base de Node.js
- Créer un serveur http embryonnaire
- Utiliser GitHub.com pour conserver une copie de sécurité
- Utiliser Glitch.com pour héberger votre service

---

1) Créez un répertoire *atelier-1*
2) Lancez VS Code et installez le greffon suivant :



3) Ouvrez le répertoire *atelier-1* (menu File/Open folder)
4) Créez le fichier *server.js* et entrez le code suivant

```
import http from 'http';
const server = http.createServer((req, res) => {
  console.log(req.url);
});
const PORT = process.env.PORT || 5000;
server.listen(PORT, () => console.log(`Server running on port ${PORT}`));
```

5) Ouvrez le panneau du terminal (menu Terminal/New Terminal)
6) Inscrivez la commande suivante : npm init
7) Répondez aux questions du formulaire de l'assistant de création du fichier *package.json* :

```
PS D:\CEGEP\Session Automne 2024\KBG\testModule> npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help init` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (atelier-1)
version: (1.0.0)
description:
entry point: (server.js)
test command:
git repository:
keywords:
author: Votre nom
license: (ISC)
About to write to D:\CEGEP\Session Automne 2024\KBG\testModule\package.json:

{
  "name": "atelier-1",
  "version": "1.0.0",
  "description": "",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node server.js"
  },
  "author": "Votre nom",
  "license": "ISC"
}


Is this OK? (yes)
```

8) Par défaut le type des modules en Node.js est commonjs. Changez ce type en ajoutant "type":"module" dans le fichier package.json afin de spécifier l'utilisation de modules ES6. Ajoutez aussi la directive qui indique la version de Node.js à utiliser.

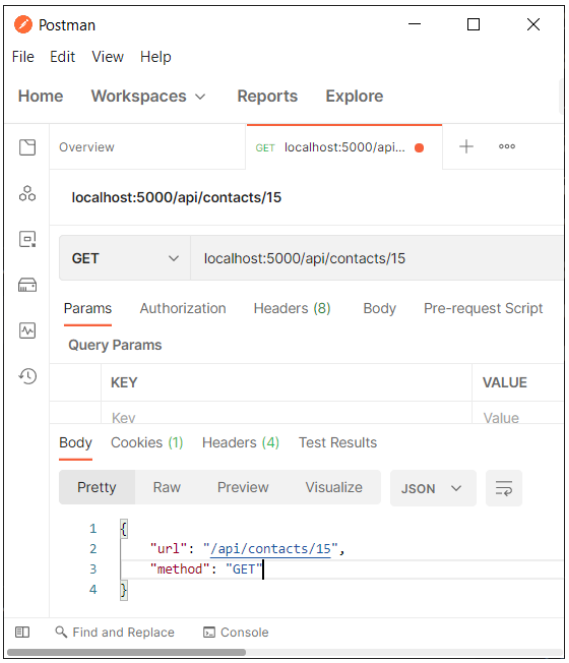
```
{
  "name": "atelier-1",
  "version": "1.0.0",
  "description": "",
  "main": "server.js",
  "type":"module",
  "engines": {
    "node": "16.14.2"
  },
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "node server.js"
  },
  "author": "Votre nom",
  "license": "ISC"
}
```

9) Ajoutez un fichier de configuration du débogueur : menu Run/Add Configuration… et sélectionnez *Node.js*. Cela ajoutera le fichier *lauch.json* qui servira lors de la prochaine exécution du code se trouvant dans *server.js*
10) Lancez server.js avec le raccourci clavier **F5** et regarder dans la console *DEBUG CONSOLE*. Il devrait être inscrit : *Server running on port 5000*
11) Ouvrez une instance de Google Chrome, et entrez l'url suivant : *localhost:5000/api/contacts/15*
12) Observez l'impact dans *DEBUG CONSOLE* de *VS Code*
13) Arrêtez le server avec le raccourci clavier **SHIFT-F5**
14) Ajoutez le code suivant après l'instruction *console.log(…)* :

```
    let reqInfo = {url:req.url, method:req.method};
    res.writeHead(200, {"Content-Type": "application/json"});
    res.end(JSON.stringify(reqInfo));
```

15) Testez à nouveau (ajoutez si vous voulez le greffon JSON Formatter à votre Google Chrome) https://chrome.google.com/webstore/detail/json-formatter/bcjindcccaagfpapjjmafapmmgkkhgoa
16) Lancez l'application POSTMAN (téléchargez et installez si pas déjà fait)
17) Ajoutez une requête GET et réglez l'URL à *localhost:5000/api/contacts/15*

18) Lancez la requête et observez le résultat.



19) Ajoutez le module *query-string* avec la commande de console : *npm install query-string*
(Notez les changements dans package.json)

20) Ajoutez dans server.js la référence à ce module :

```
import queryString from "query-string";
```

21) Remplacez le corps du callback (req, res) => {…} par le suivant :

```
console.log(req.url);
let reqInfo = { url: req.url, method: req.method, contentType: req.headers['content-type'] };

res.writeHead(200, { "Content-Type": "application/json" });
if (req.method == 'GET') {
    res.end(JSON.stringify(reqInfo));
} else {
    if (req.method == 'POST') {
        let body = [];
        req.on('data', chunk => {
            body += chunk;
        }).on('end', () => {
            try {
                if (req.headers['content-type'] === "application/json")
                    reqInfo.body = JSON.parse(body);
                else
                if (req.headers['content-type'] === "application/x-www-form-urlencoded")
                    reqInfo.body = queryString.parse(body.toString());
                else
                    reqInfo.body = body.toString();
                res.end(JSON.stringify(reqInfo));
            } catch (error) {
                console.log(error);
            }
        });
    }
}
```

Note : Vous pouvez formatter le code en faisant bouton de droite et appeler « Format Document » ou appliquer l'équivalence clavier **SHIFT+ALT+F**

22) Ajoutez une nouvelle requête dans *POSTMAN* :

Méthode : POST,
URL : localhost:5000/api/contacts,
CONTENT-TYPE : application/json
BODY : {"FirstName": "Kyle", "LastName": "Ross", "Email": "Kyle.Ross@clg.qc.ca" }

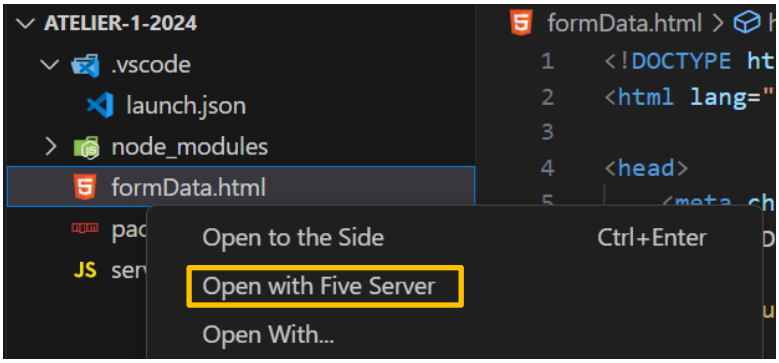
23) Lancez la requête et observez le résultat

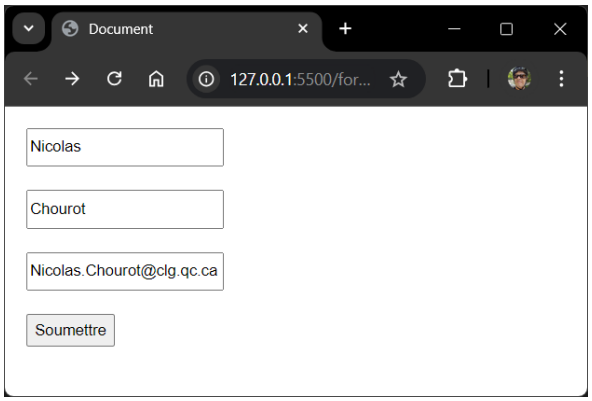
24) Ajoutez un fichier *formData.html* et y collez le contenu suivant :

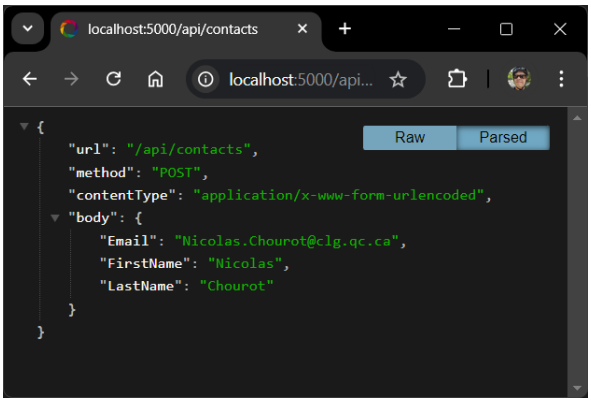
```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <title>Document</title>
    <style>
        input {
            margin: 10px;
            height: 28px;
        }
    </style>
</head>

<body>
    <form action="http://localhost:5000/api/contacts" method="POST">
        <input type="text" placeholder="Firstname" name="FirstName" required> <br>
        <input type="text" placeholder="Lastname" name="LastName" required><br>
        <input type="email" placeholder="Email" name="Email" required><br>
        <input type="submit" value="Soumettre">
    </form>
</body>

</html>
```

25) Installez les greffons suivants dans VS Code :

26) Lancez la page *formData.html* par l'entremise de ce dernier greffon.

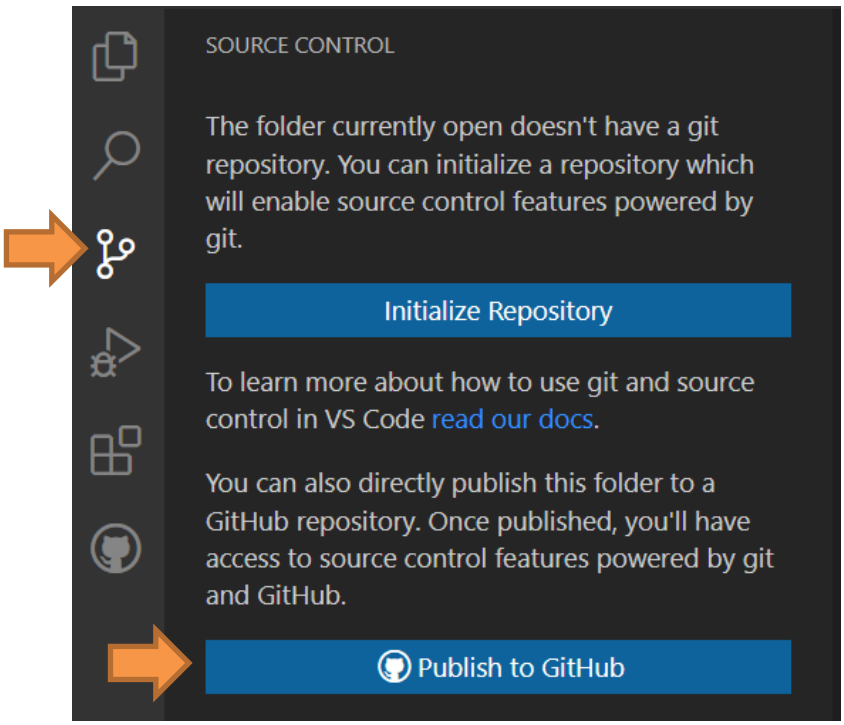




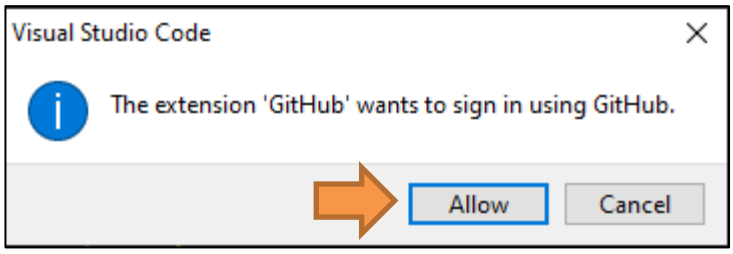


27) Ajoutez dans le projet le fichier .gitignore disponible dans les ressources du cours (Colnet)

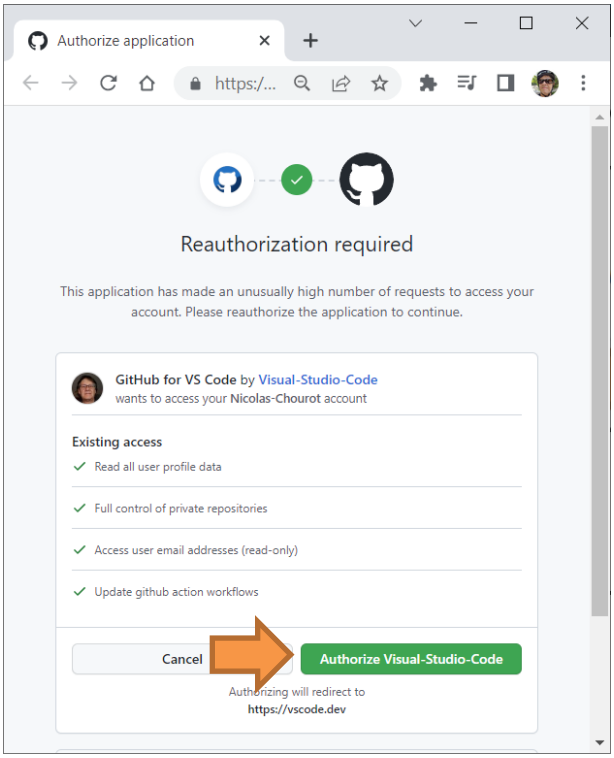
28) Publiez le projet dans GitHub :



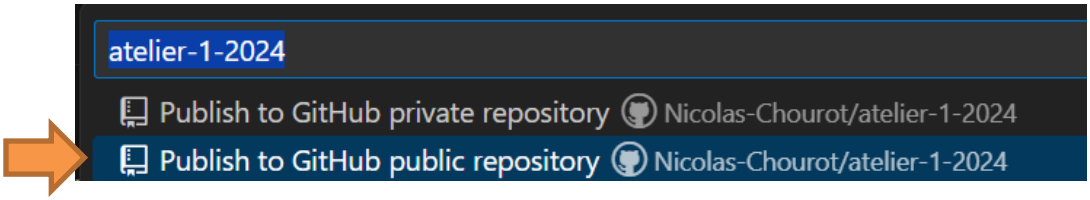
29) Acceptez de se connecter via l'extension GitHub :



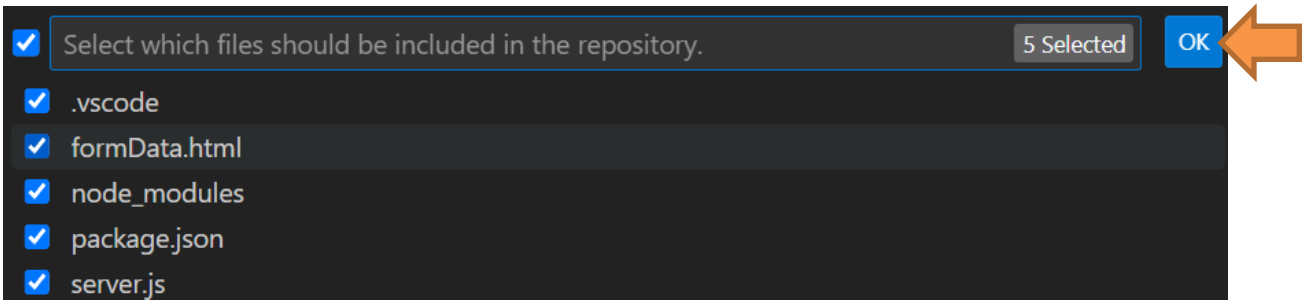
30) Une instance de Google Chrome apparaîtra pour vous demander d'accepter la connexion :



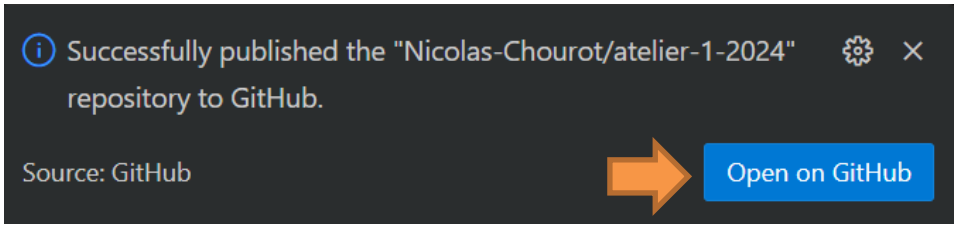
31) Sélectionnez « publier dans un répertoire public » :



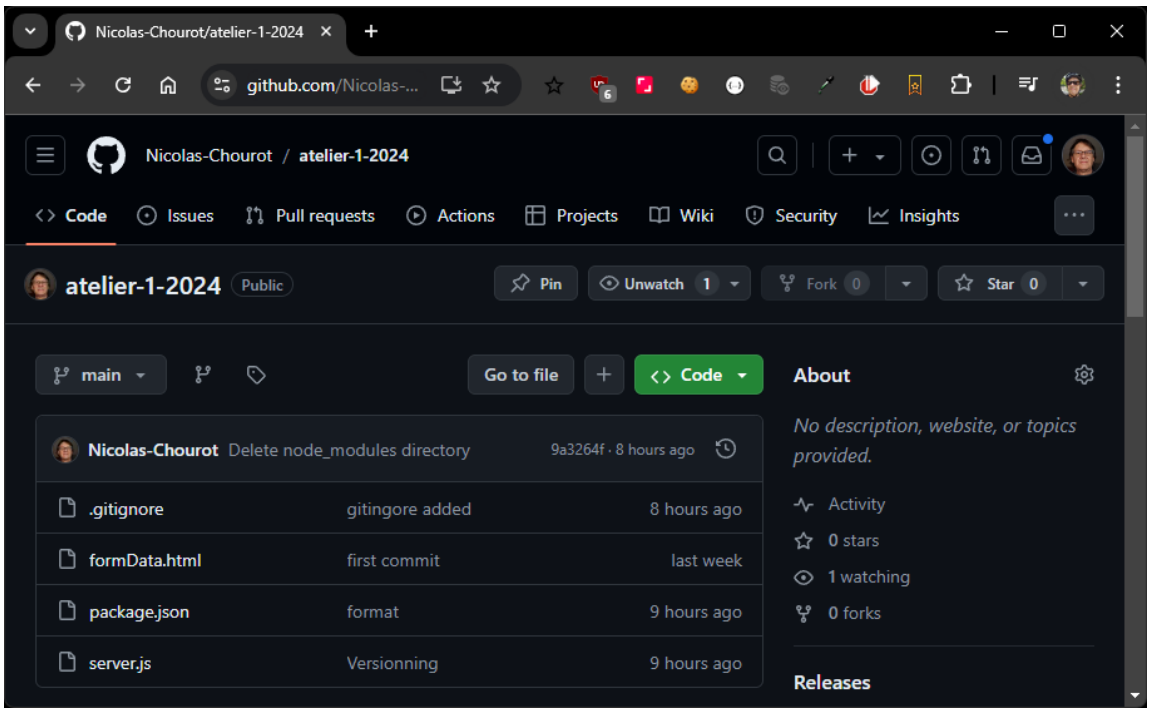
32) Acceptez de publier tous les fichiers et répertoires :



33) Message de confirmation de l'opération, allez dans le site de Github pour contempler votre répertoire :



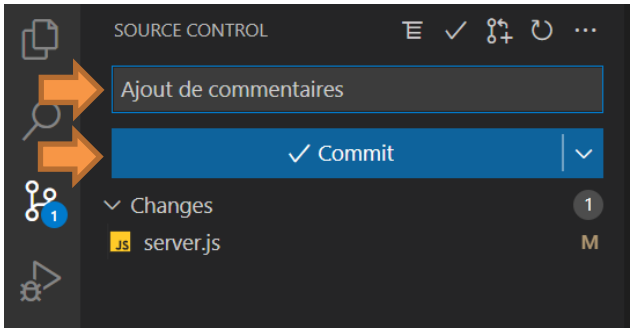
34) Votre répertoire dans GitHub :



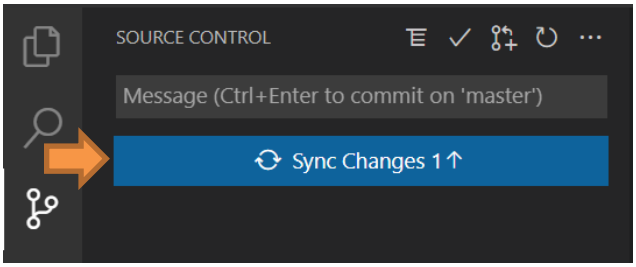
35) Ajouter le commentaire au début du fichier server.js

```
// Mon premier server Http
```

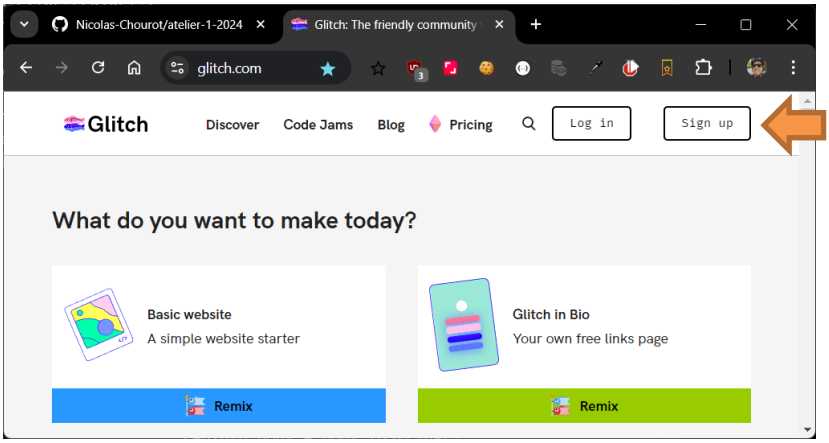
36) Commettre les changements dans Git : Ajouter le commentaire « ajout de commentaire », et commettre :



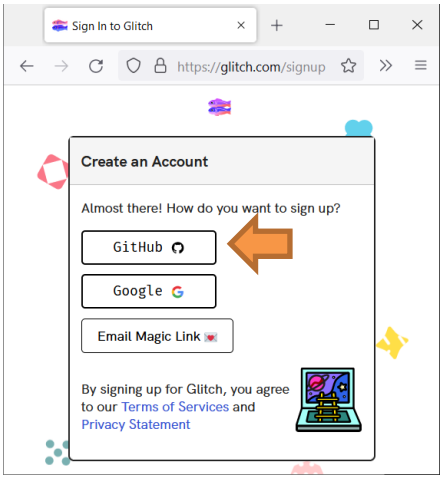
37) Synchroniser les changements dans GitHub :



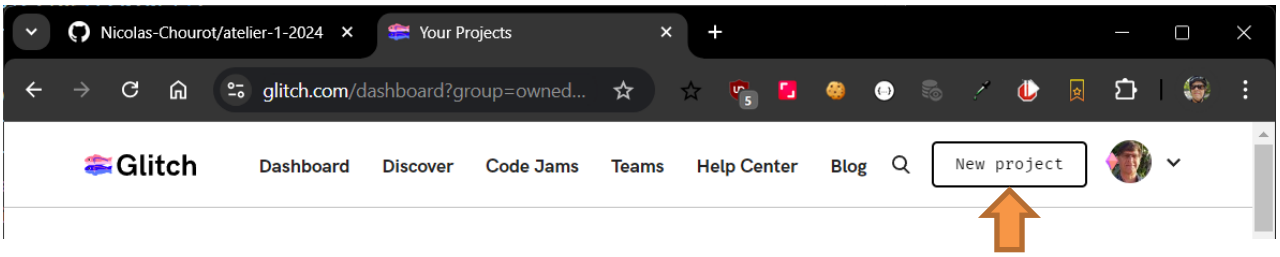
38) Créer un compte Glitch : https://glitch.com



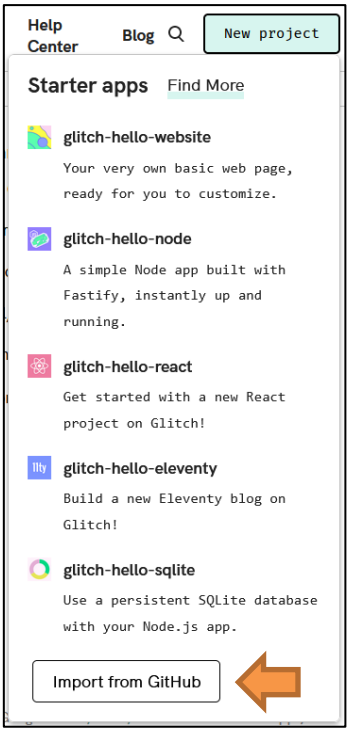
39) Sélectionnez votre compte GitHub :



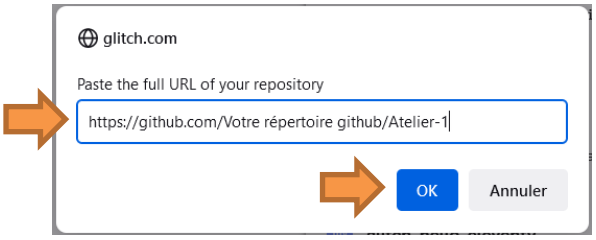
40) Une fois votre compte créer, ajoutez un nouveau projet :



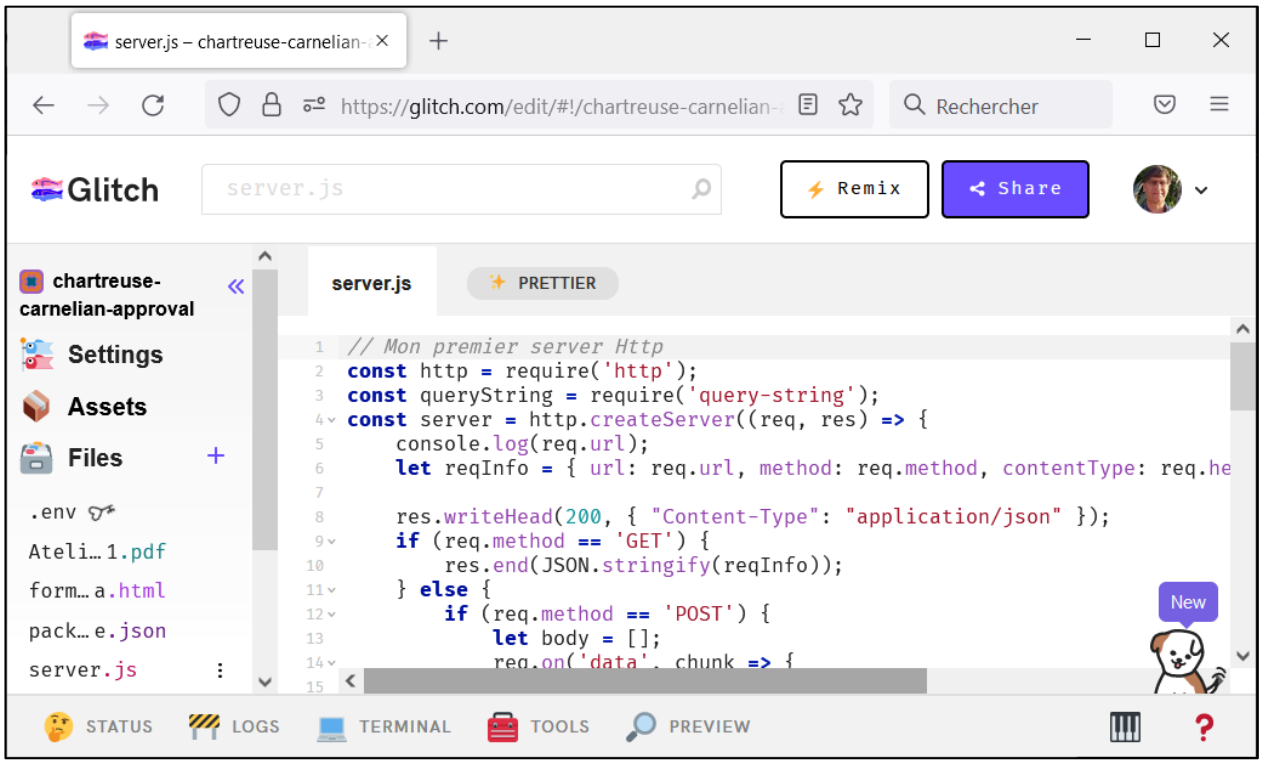
41) Importez à partir d'un répertoire GitHub :



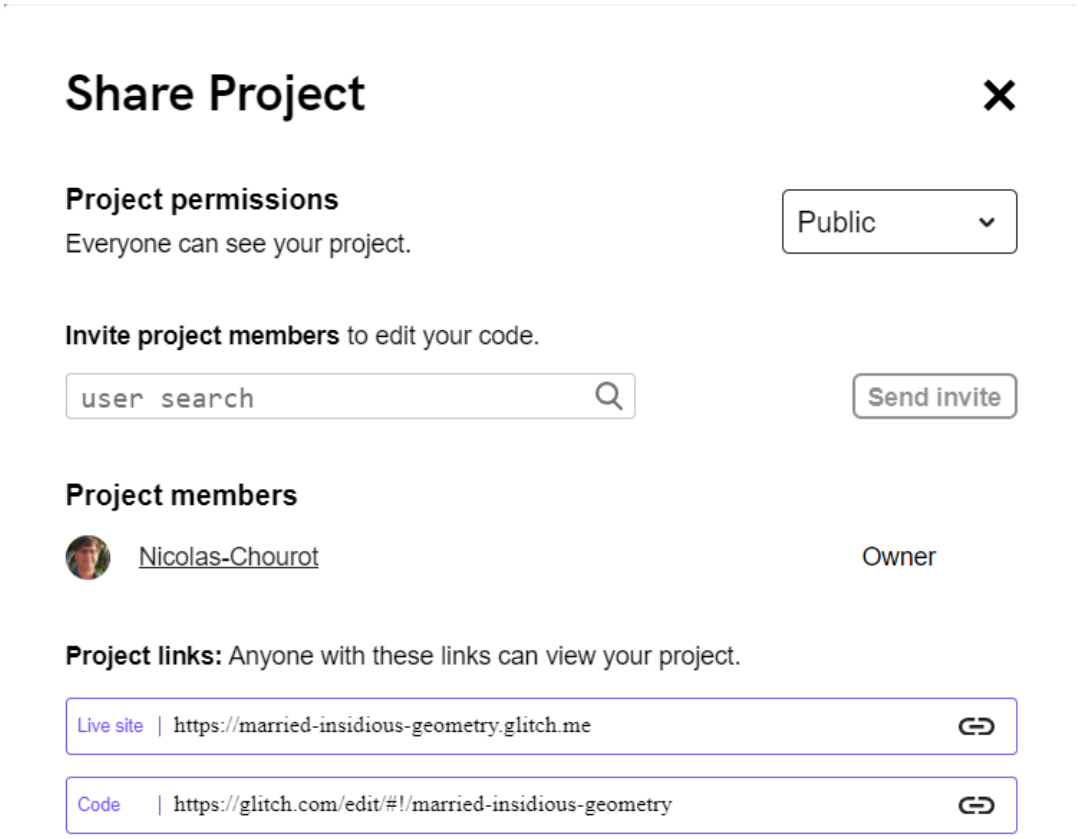
42) Copier l'url du répertoire GitHub et coller le dans le dialogue :



43) Interface de votre projet sur Glitch :



44) Connaître l’url de votre site :



45) Faire les modifications suivantes :

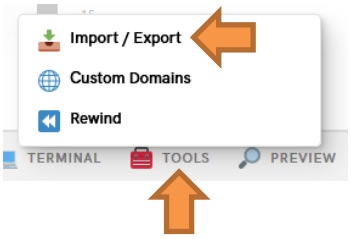
Dans *formData.html* :

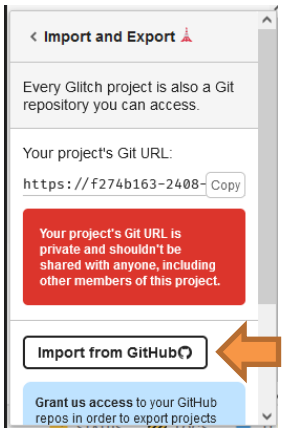
```
<form action="https://xxx-yyy-zzz.glitch.me" method="POST">
```

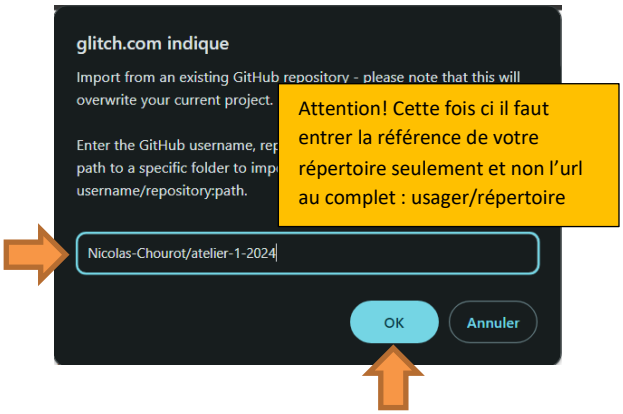
Note : utilisez l'url de vote site Glitch

46) Commettre les changements dans GitHub

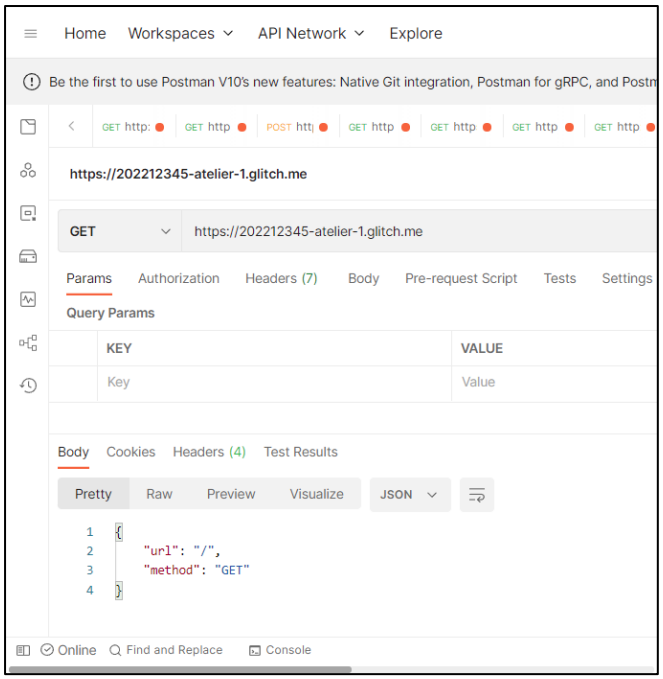
47) Mettre à jour votre site Glitch avec la version GitHub :

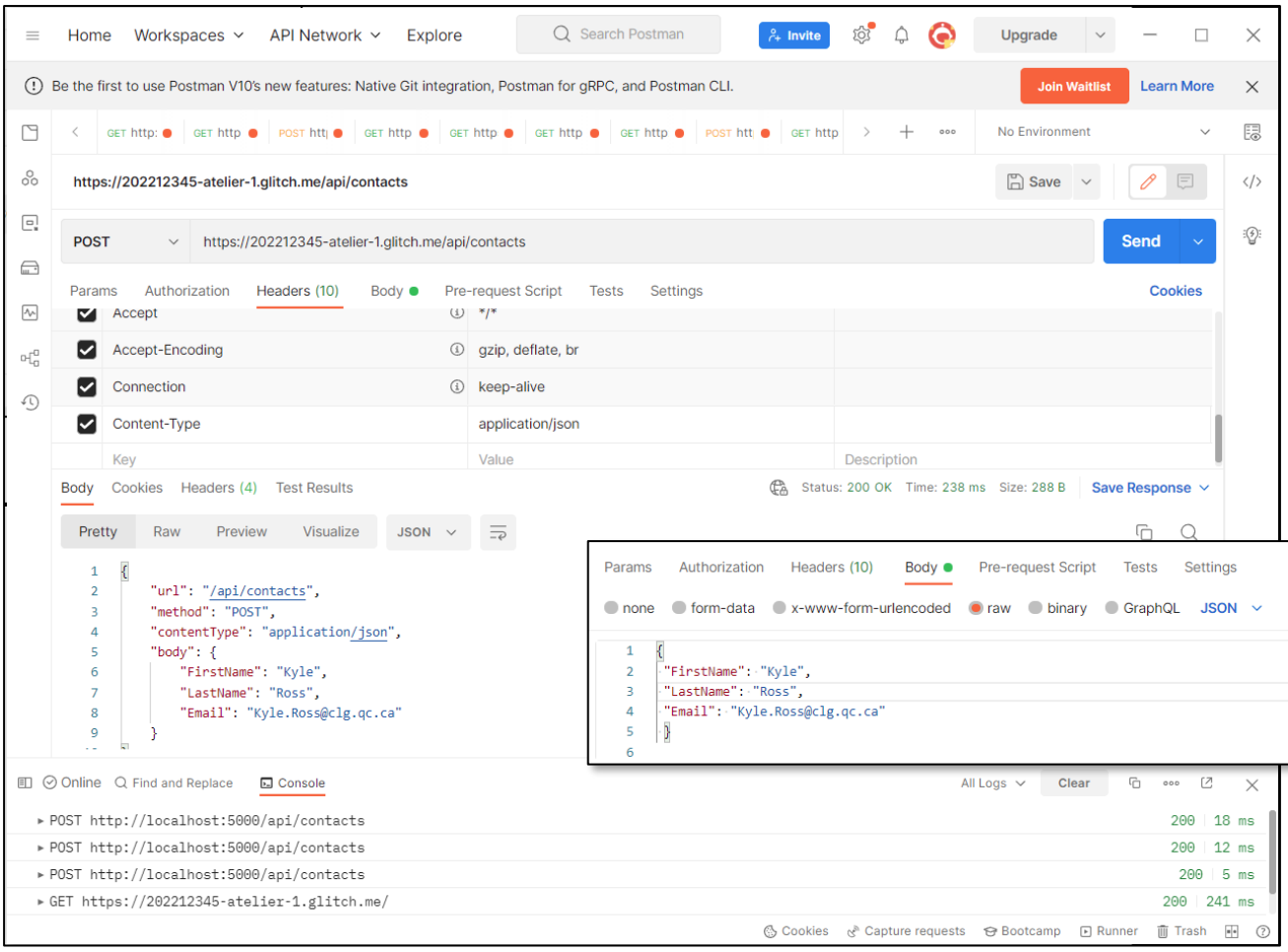






48) Vérifier le bon fonctionnement de votre site Glitch avec Postman et la version locale de *formdata.html* (utilisez l'url de votre site Glitch plutôt que http://localhost:50000...)





Fin de l'atelier