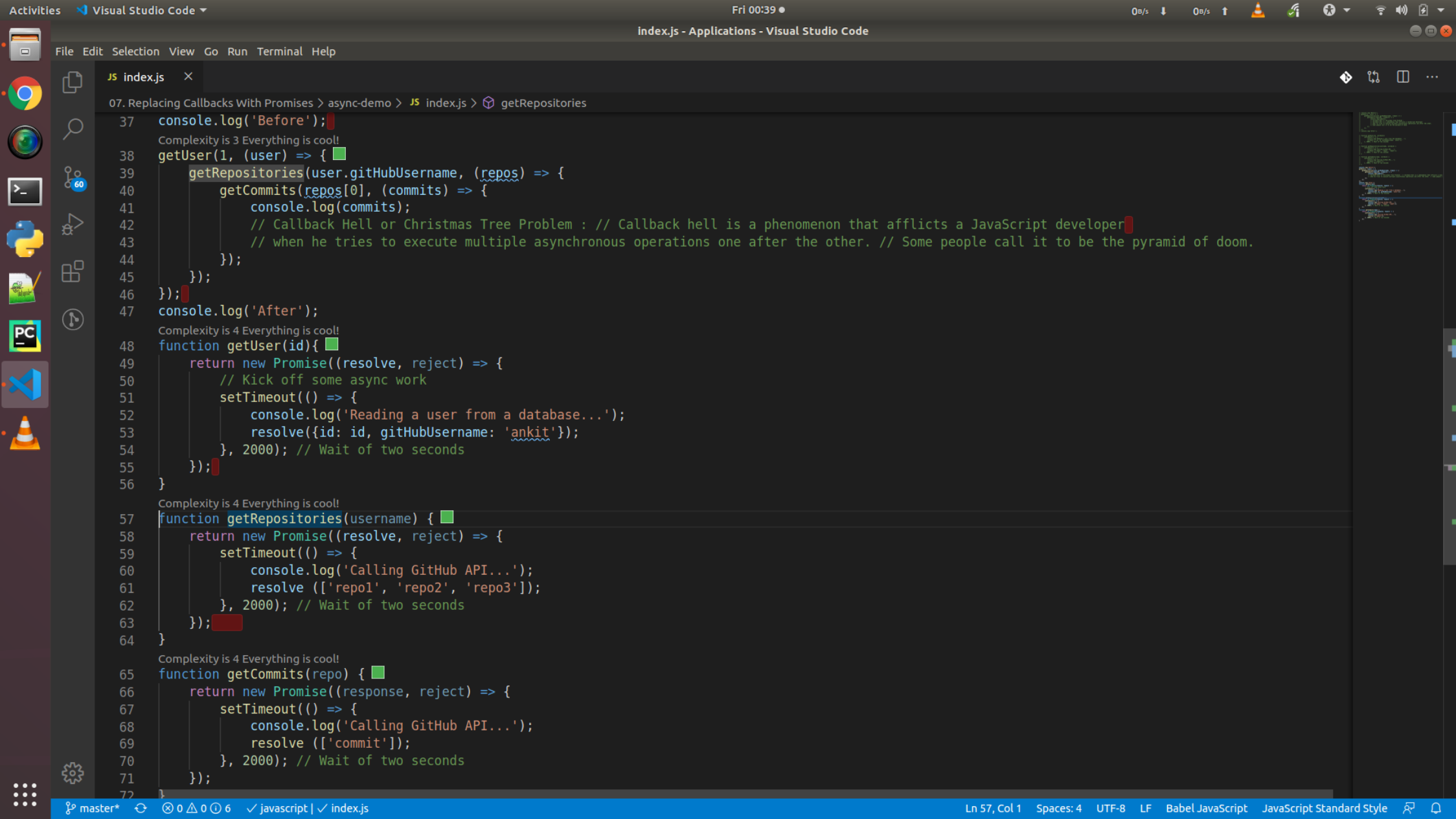


Replacing Callbacks with Promises



Mosh Hamedani

programmingwithmosh.com



```
37 console.log('Before');
38 Complexity is 3 Everything is cool!
39 getUser(1, (user) => {
40     getRepositories(user.gitHubUsername, (repos) => {
41         getCommits(repos[0], (commits) => {
42             console.log(commits);
43             // Callback Hell or Christmas Tree Problem : // Callback hell is a phenomenon that afflicts a JavaScript developer
44             // when he tries to execute multiple asynchronous operations one after the other. // Some people call it to be the pyramid of doom.
45         });
46     });
47 console.log('After');
48 Complexity is 4 Everything is cool!
49 function getUser(id) {
50     return new Promise((resolve, reject) => {
51         // Kick off some async work
52         setTimeout(() => {
53             console.log('Reading a user from a database...');
54             resolve({id: id, gitHubUsername: 'ankit'});
55         }, 2000); // Wait of two seconds
56     });
57 }
58 Complexity is 4 Everything is cool!
59 function getRepositories(username) {
60     return new Promise((resolve, reject) => {
61         setTimeout(() => {
62             console.log('Calling GitHub API...');
63             resolve(['repo1', 'repo2', 'repo3']);
64         }, 2000); // Wait of two seconds
65     });
66 }
67 Complexity is 4 Everything is cool!
68 function getCommits(repo) {
69     return new Promise((response, reject) => {
70         setTimeout(() => {
71             console.log('Calling GitHub API...');
72             resolve(['commit']);
73         }, 2000); // Wait of two seconds
74     });
75 }
```