Using access tokens in the frontend

(Demo prep)

- Starting Catalog items:

Name	Description	Price
Potion	Restores some HP	5
Antidote	Cures poison	7
Hi-Potion	Restores a medium amount of HP	9

- Start with Admin, Player1 and Player2
- Player1 and Player2 have 100 gil (Admin has 0)
- Player1 has 3 Potions in Inventory bag
- Disable Edge extensions and favorites
- Apply VS Code settings (%APPDATA%\Code\User\settings.json)
- Apply PowerShell environment settings

Exploring the front-end client

Let's see now how the front-end portal is taking advantage of OAuth 2.0 access tokens to make authorized calls to the microservices that we secured in previous modules.

In Play.Frontend

1. Start signed in as admin
2. Start Catalog and Inventory microservices
3. Go to Catalog section in front-end
4. Place a breakpoint in Catalog.js → populateItems()
5. Refresh front-end
6. Copy the value of the token
7. Decode the token in jwt.ms
8. Remove breakpoint
9. Browse to Users section
10. Increase gil of Player1 to 200
11. Copy Id of Player1

12. Go to Catalog
13. Grant an Antidote to Player1
14. Back to Users
15. Click Inventory Bag of Player1
16. Show Antidote was granted
17. Sign out
18. Sign in as Player1
19. Browse to My Inventory
20. Place a breakpoint in Inventory.js → populateItems()
21. Stop Inventory microservice
22. Place a breakpoint in ItemsController → GetAsync
23. F5 Inventory
24. Refresh My Inventory
25. Show how user.sub is used in front-end
26. Show how sub is read in ItemsController
27. Logout
28. Go to Login
29. Click on Register
30. Register Player3
31. Notice the error page

In the next lesson we will make a small modification to our Identity microservice to fix the new user Registration experience.