Working with JavaScript Modules

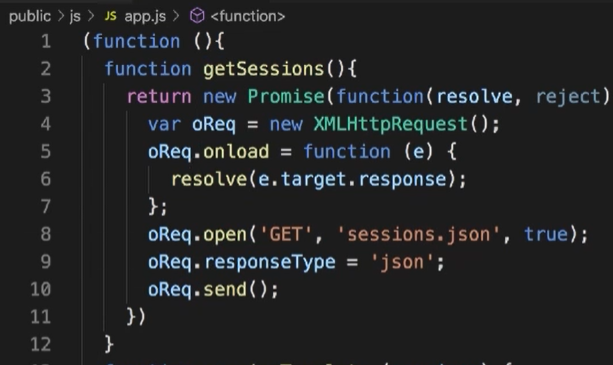
Zawartość

[What is a Module 2](#_Toc68520683)

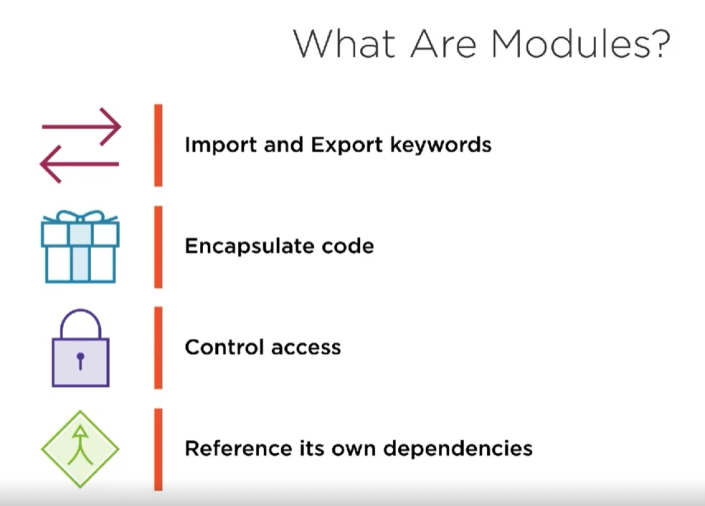
[Create Modules 3](#_Toc68520684)

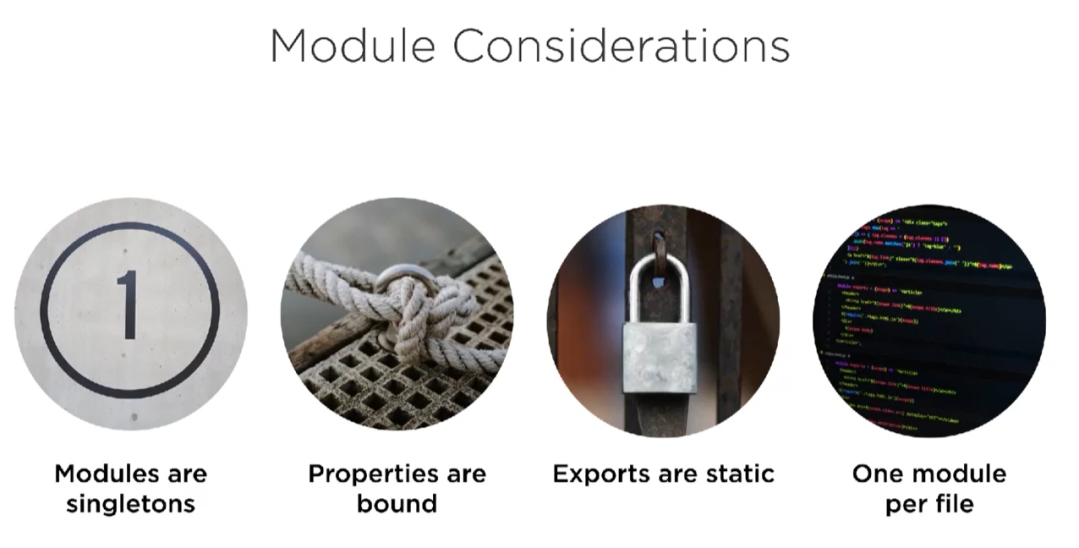
[Import Modules 6](#_Toc68520685)

# What is a Module



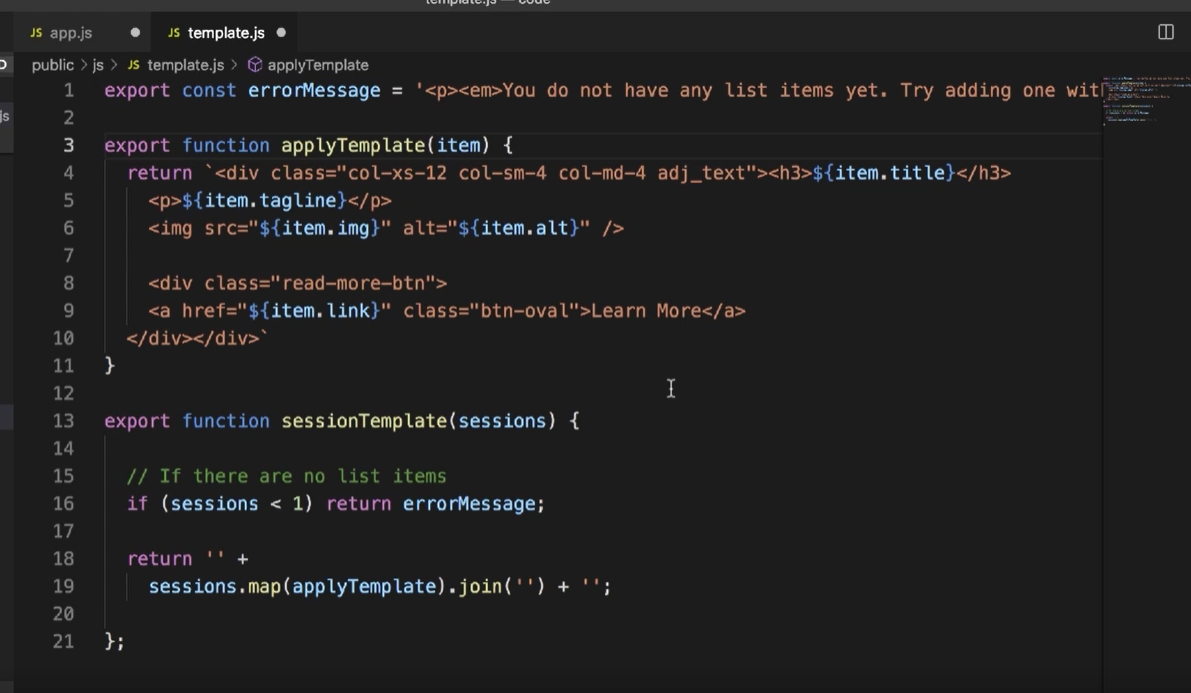
Always use iife. There is no acces to data from function because there is a scope. It is a great way to encapsulated all variables.

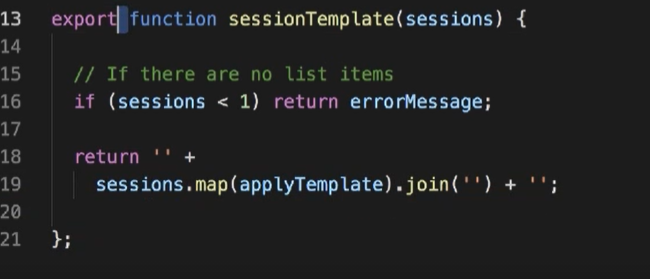


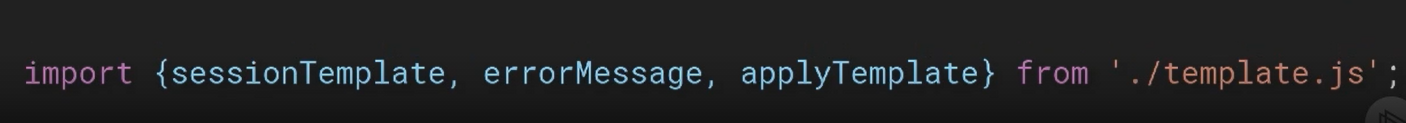
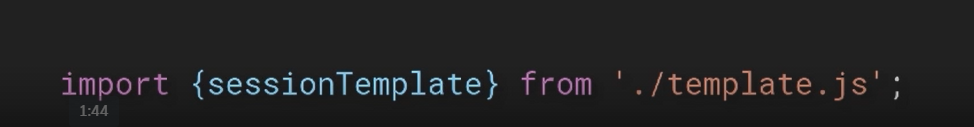


Each model has the same instance for each export. Properties from models are bound it means when dev changes some prop there are change in everywhere.

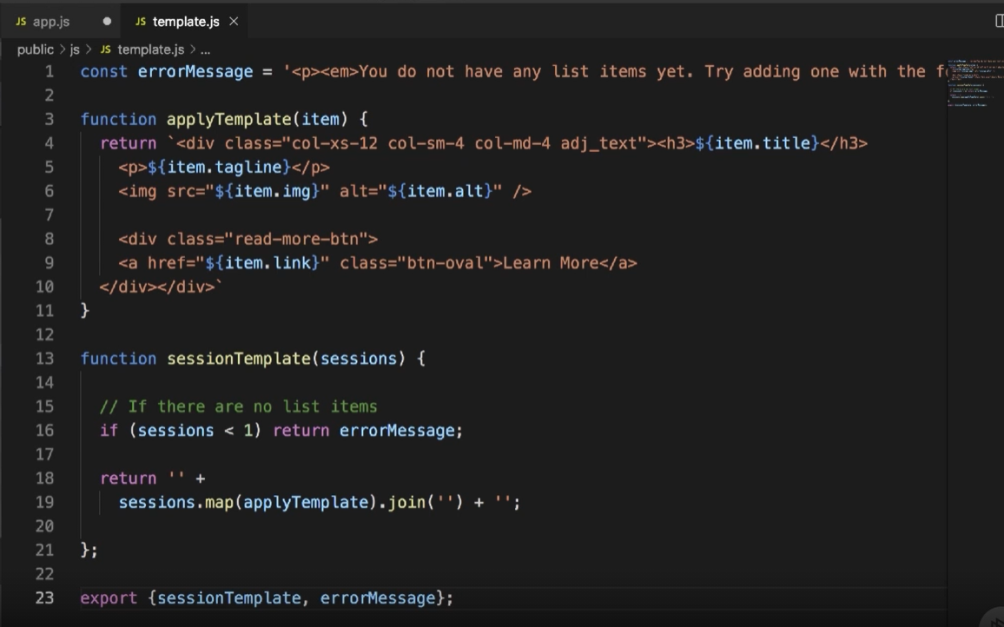
# Create Modules



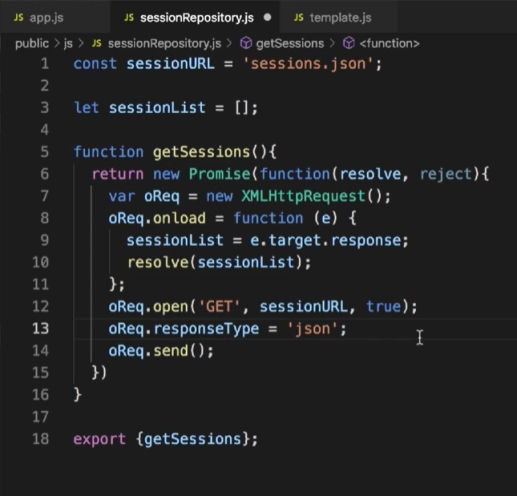




We can import all things in single line



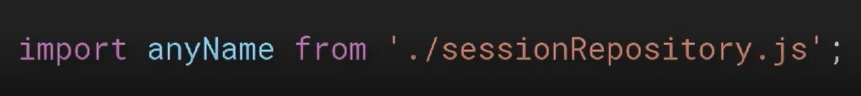
Instead of use export in each line. Keyword can be use in last file in module and return needs methods and props. All this methods are return as list.

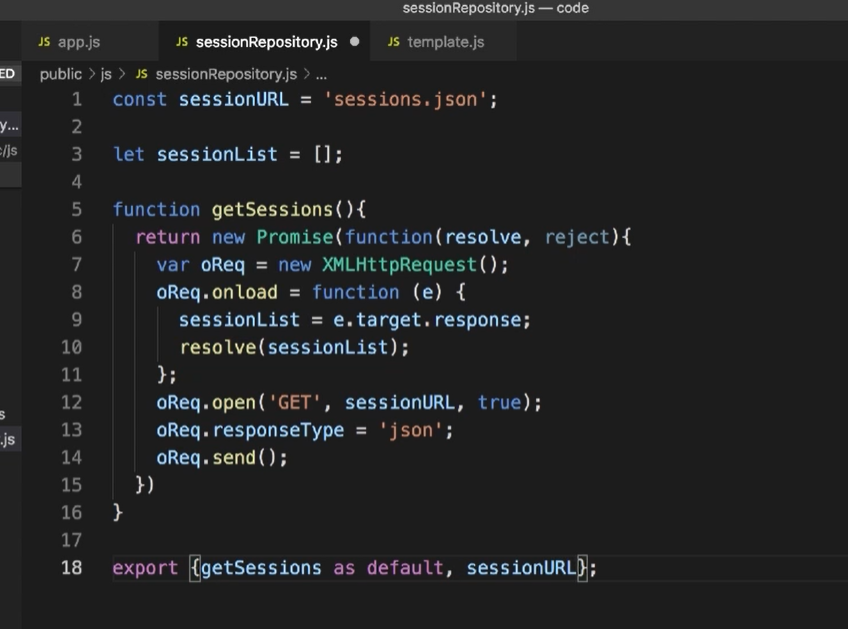
 Another example of use a export.

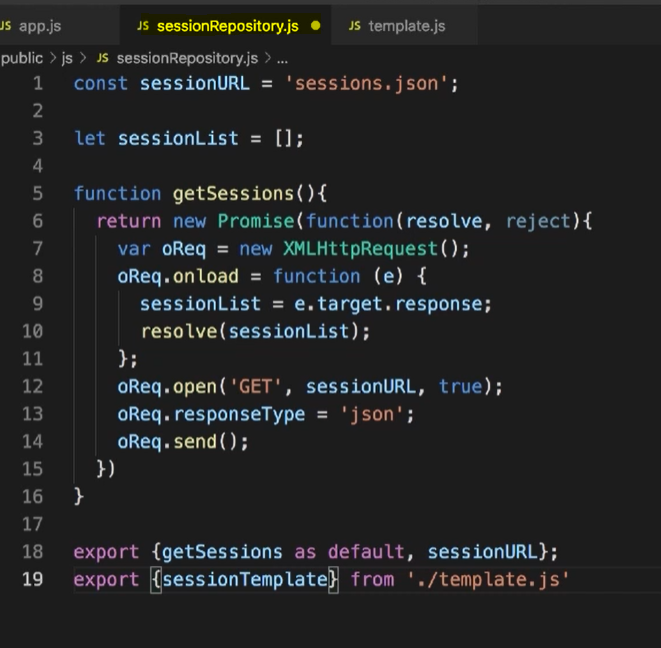


Methods and props can be rename in export.



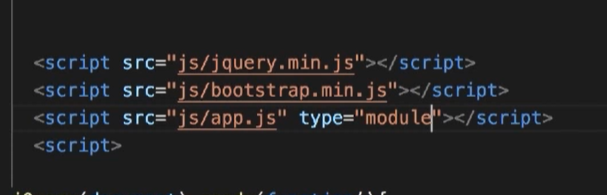


 By default can be import all things from module. With this keyword there is no need to use curly braces.

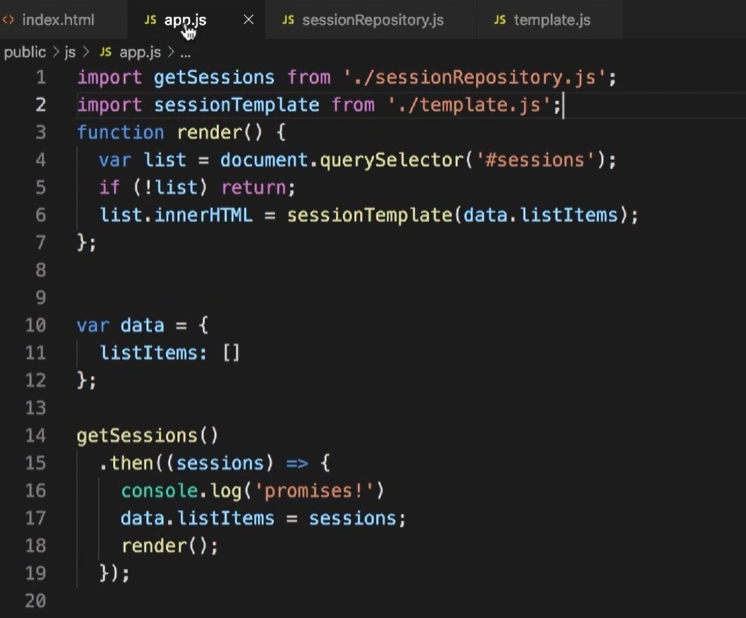


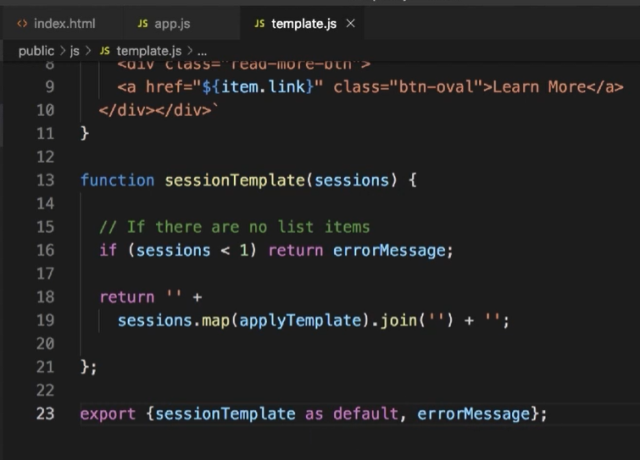
Exports can be aggregate in one js file. It works like in example above. Template is exported from sessionRepo. To do this operation required is export x from pathToFile. This feature is name re-export

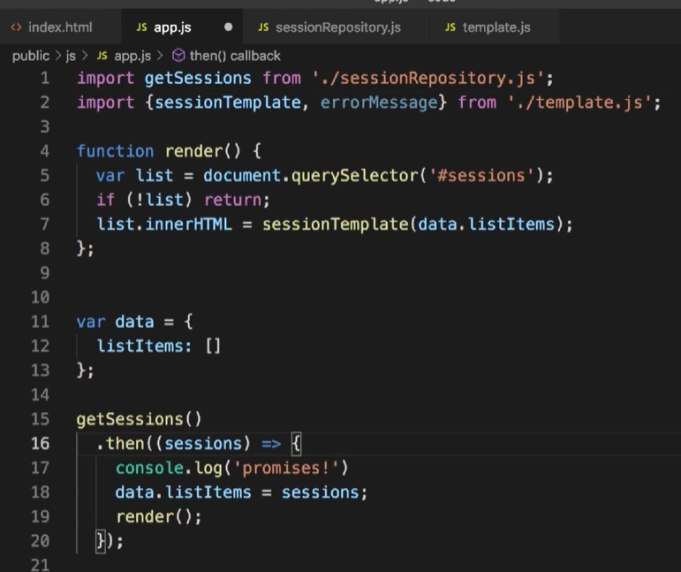
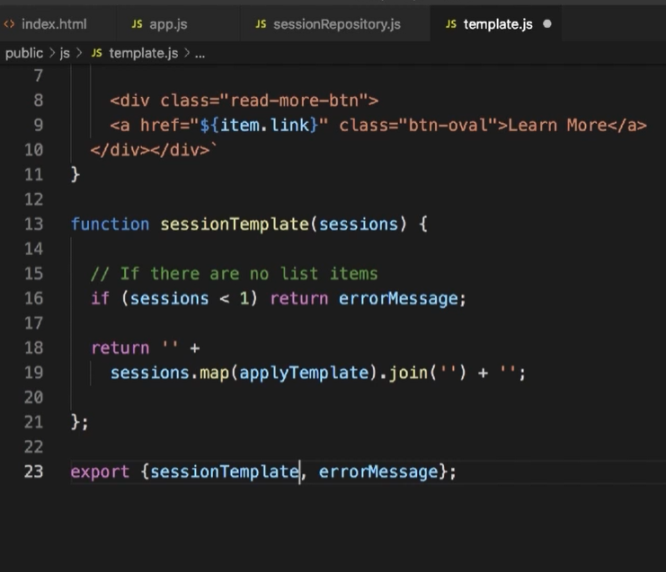
# Import Modules



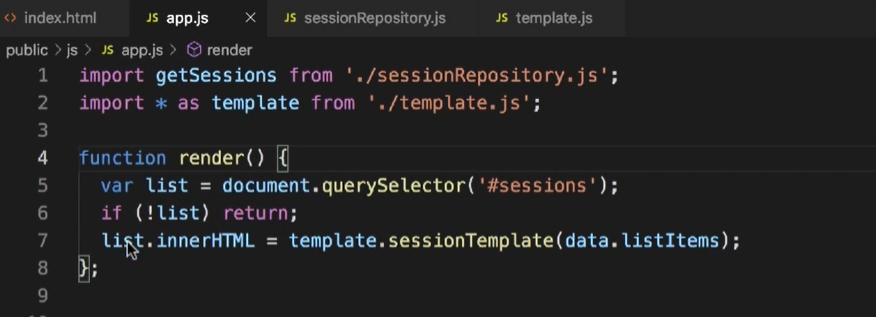
Each js file which use imports must be mark module.

everything is import

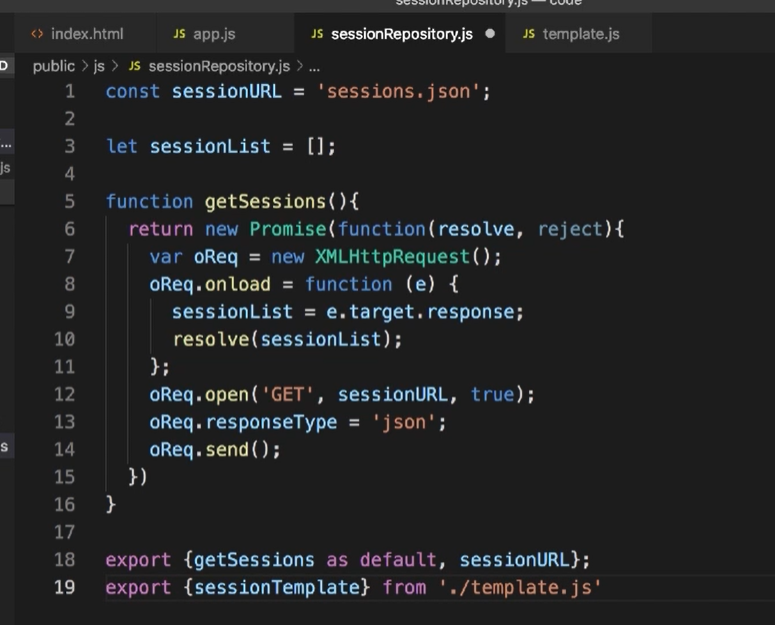
 small change in template. Now export is default.

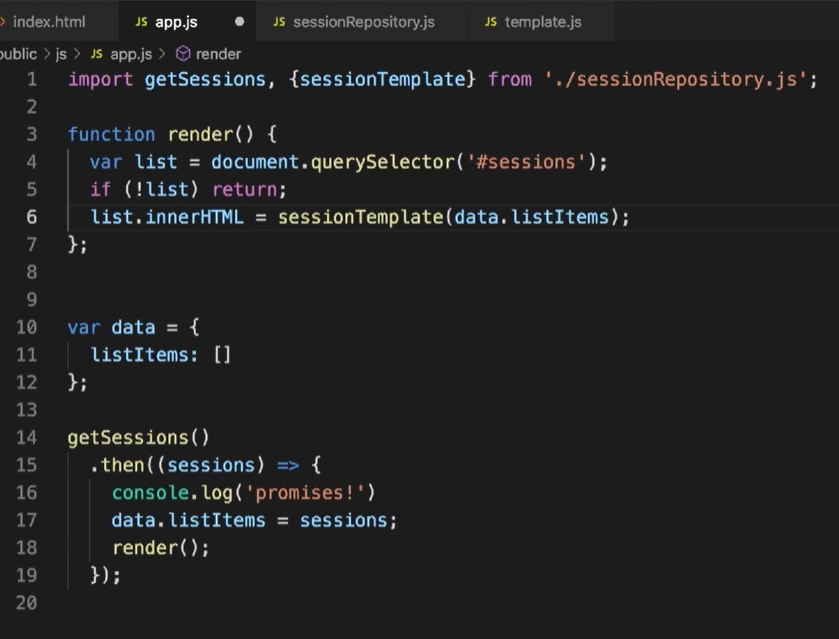
 

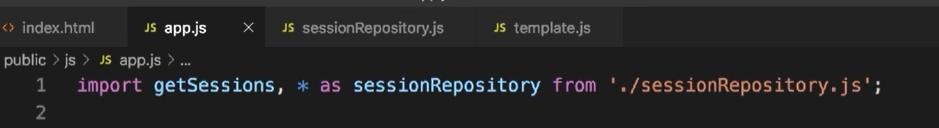
import by name is more readable. Don’t use a default import. If import is by name export must be too.



Another example is import \* as name. This import retrieve all module and can be used like an object look on line 7. Template invoke sessionTemplate method.

 Use import and re-export.

 import is simple get as many as there are imports each must be separated by coma.



Of course we can use hybrid code to get all related code.