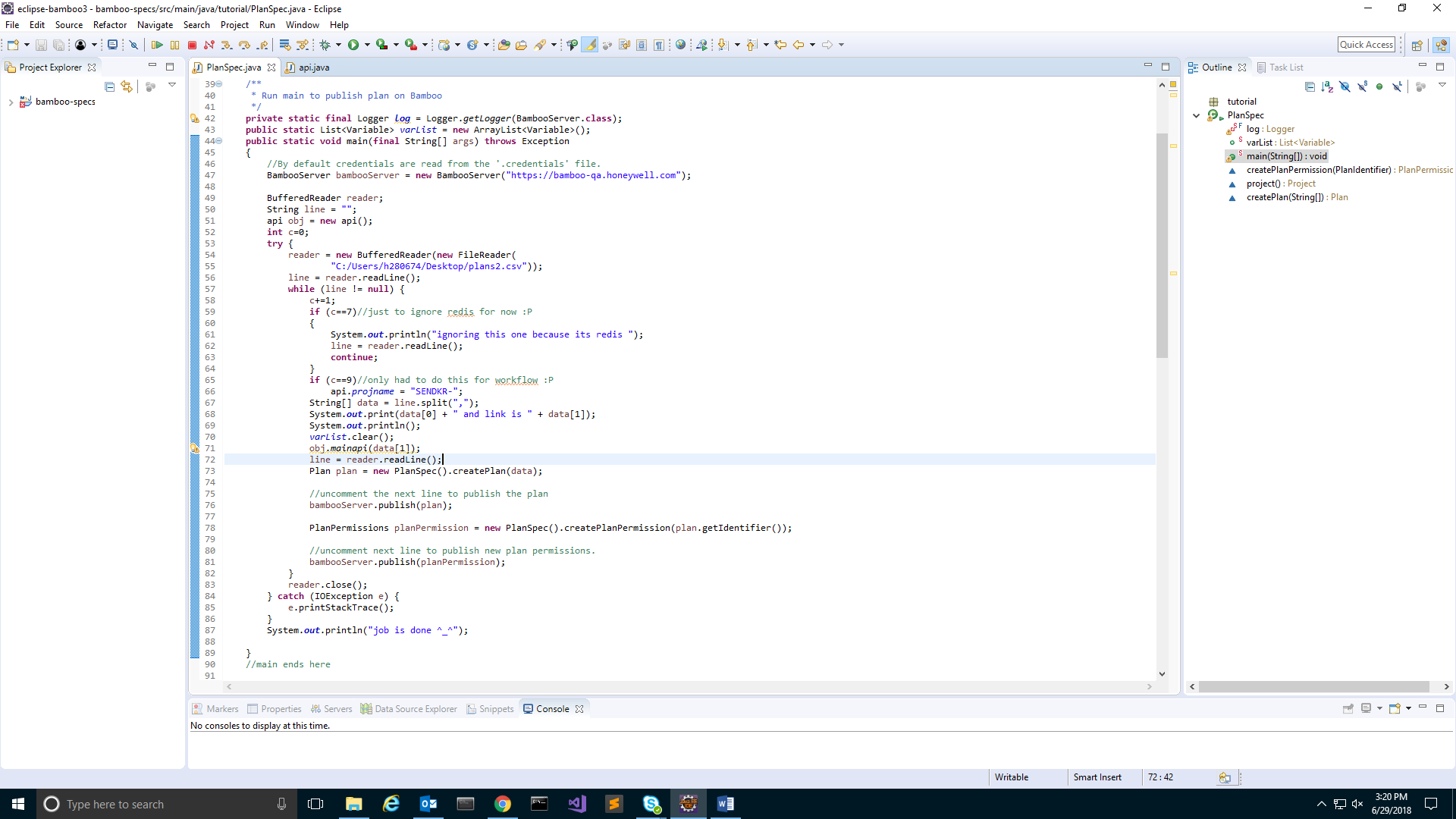
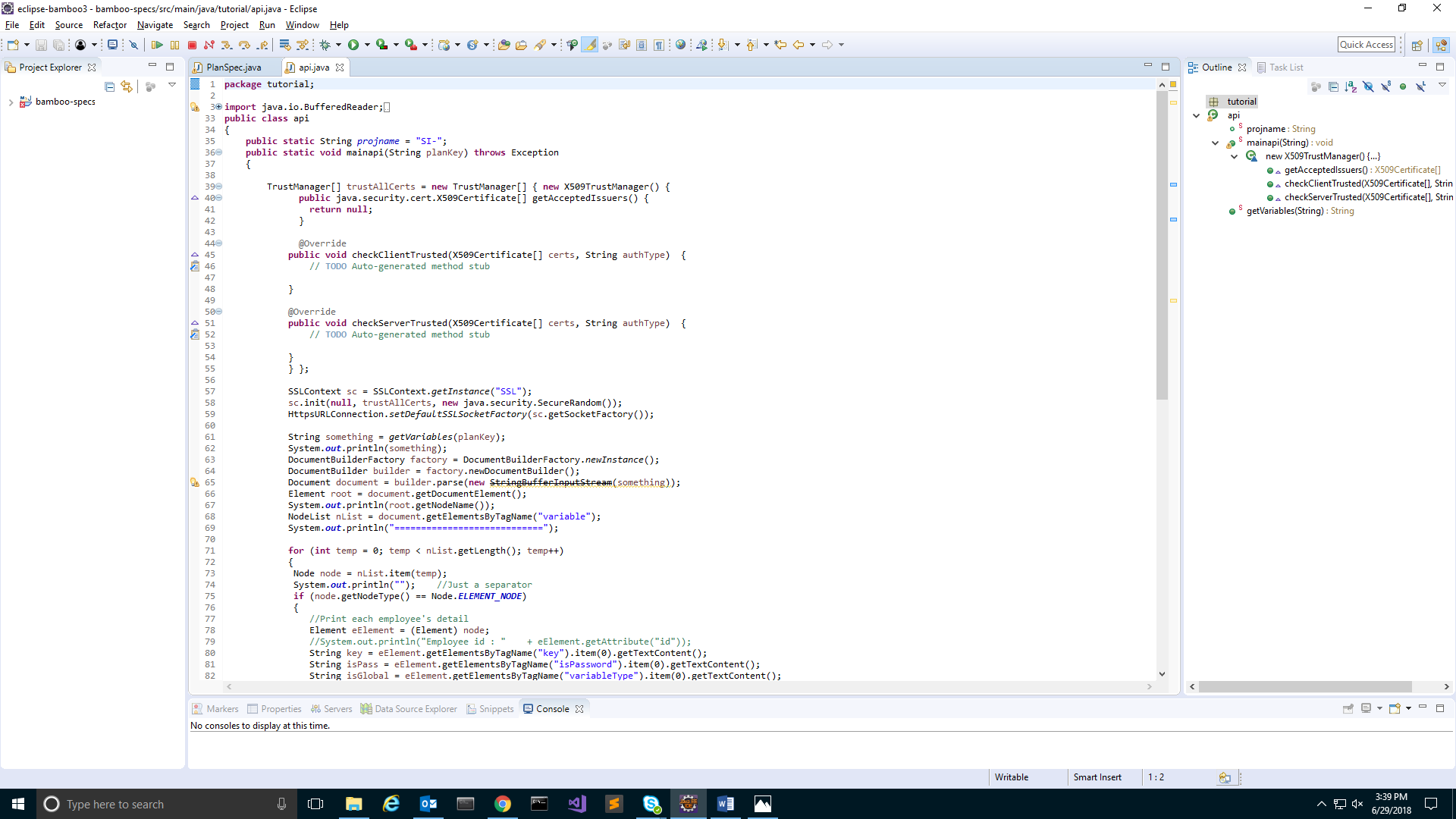
Bamboo Automation

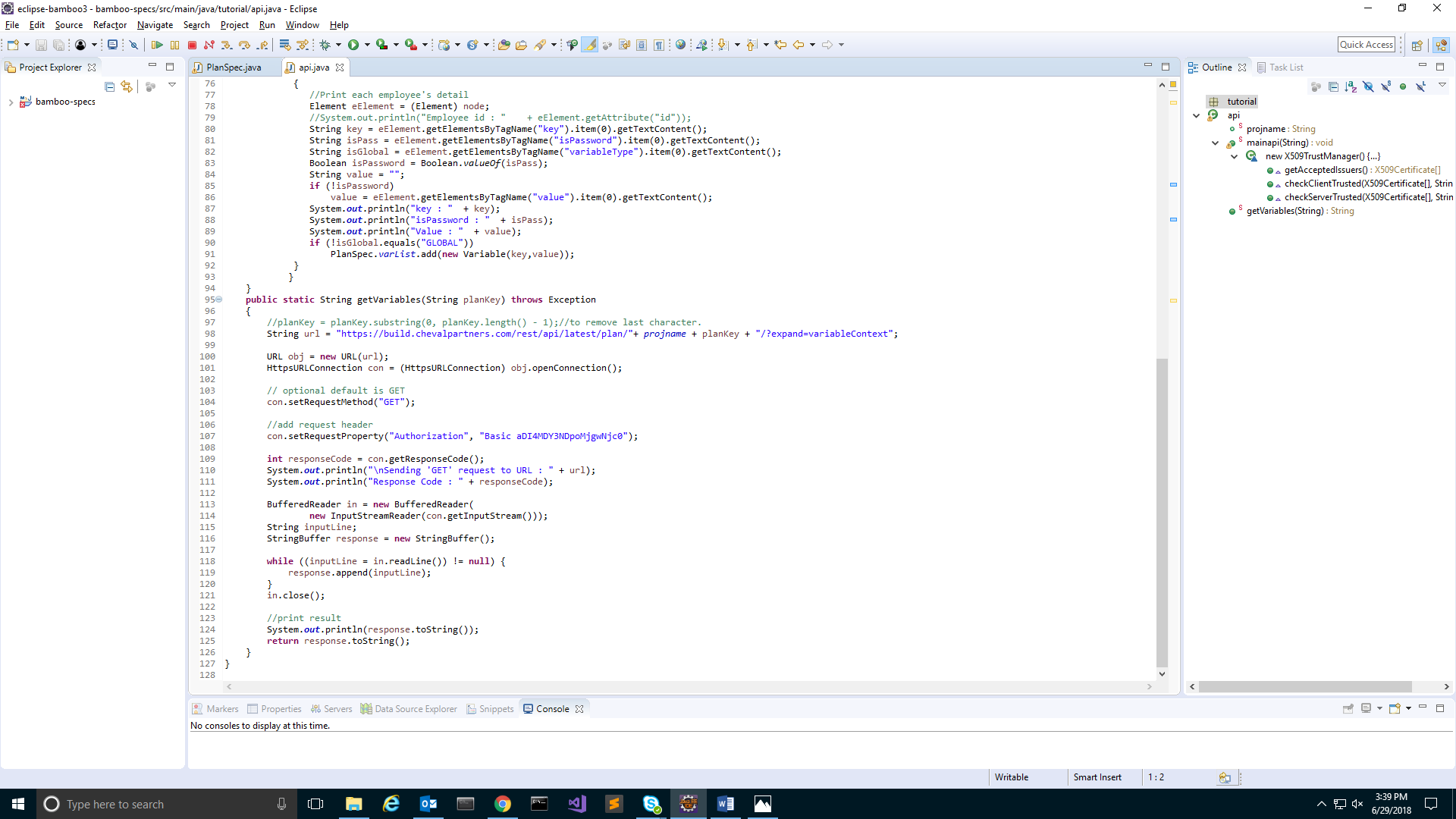
* Read about bamboo java specs reference online and get a base code. Another way to get this code is to go to a plan and select “configure plan” from the right top corner drop down. Then in the configure plan page, go to view plan as bamboo specs. This opens another page with the entire configuration as code.
* The program starts with main function. Global variable varList is initialized to store plan variables(line 43).
* The credentials for bamboo server connection is taken from the .credentials file present in the root folder of the project.
* Keep details like plan-key, repository name, repo url, etc. in a csv file.
* Read from this csv file in your java code and update fields accordingly. The reading starts from line 54. File Reader class is used to read from csv file.



* Call api to get variables(both global and plan). The api used is {bamboo-server}/rest/api/latest/plan/{project-key}-{plan-key}/?expand=variableContext. The function mainapi is called in line number 71. This class is responsible for getting the variables through api and update an array of type Variable. This array is then passed to the plan.



* An SSL context is created to bypass certificate verification as the api is https. This is shown in the screenshot above.
* Line 61 calls the function getVariables passing the parameter planKey.



* This function returns an xml as a string.
* It is then parsed using DocumentBuilder class in javax.xml.parsers.
* NodeList contains all the variables returned by api (line 68).
* Then we loop through this list and only add those variables which are not global.( line 90).
* Once the varList is updated, control returns to main function, which now creates a plan object by filling all the fields with data from the csv and passing the Variable array called varArray.
* This plan is then published.
* New plan permission is created for the plan which can be modified through code.
* Plan permission is published.