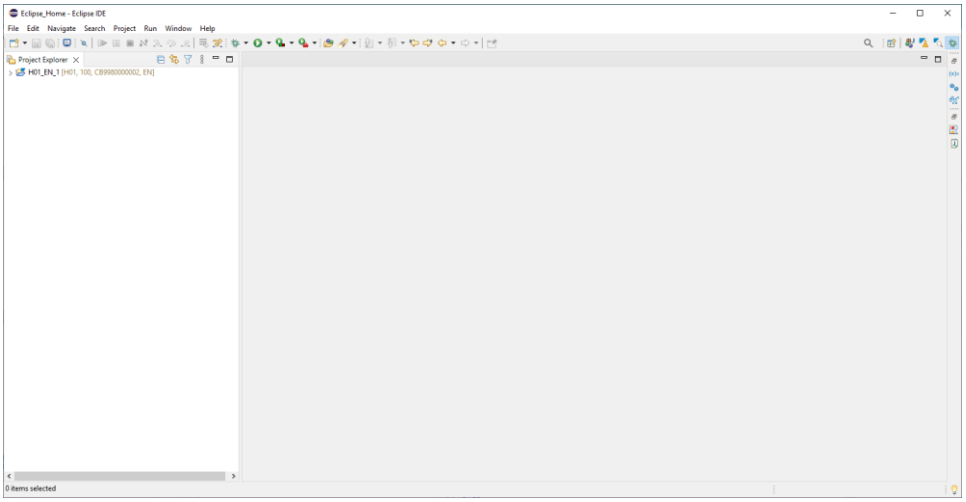
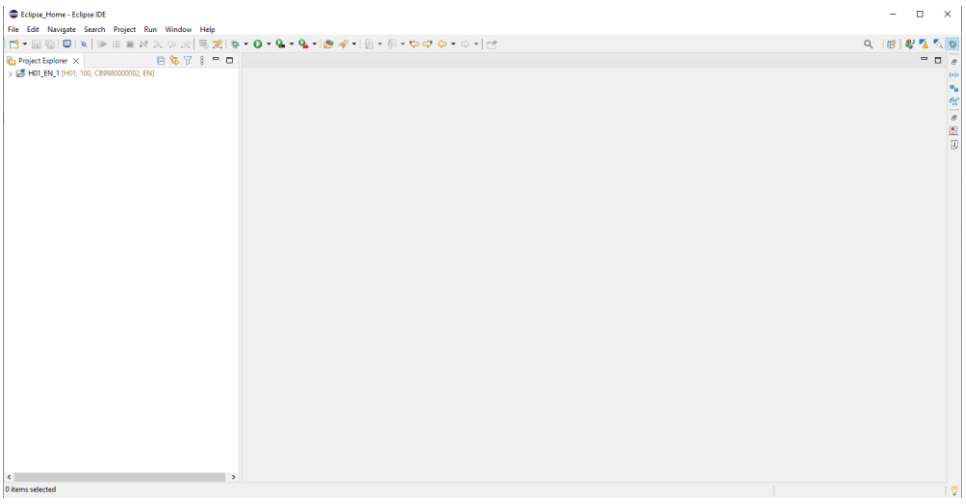
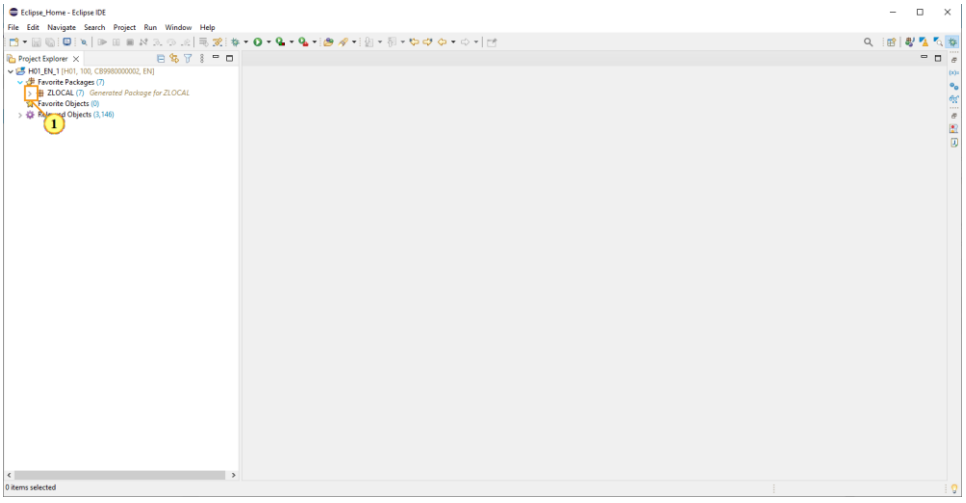
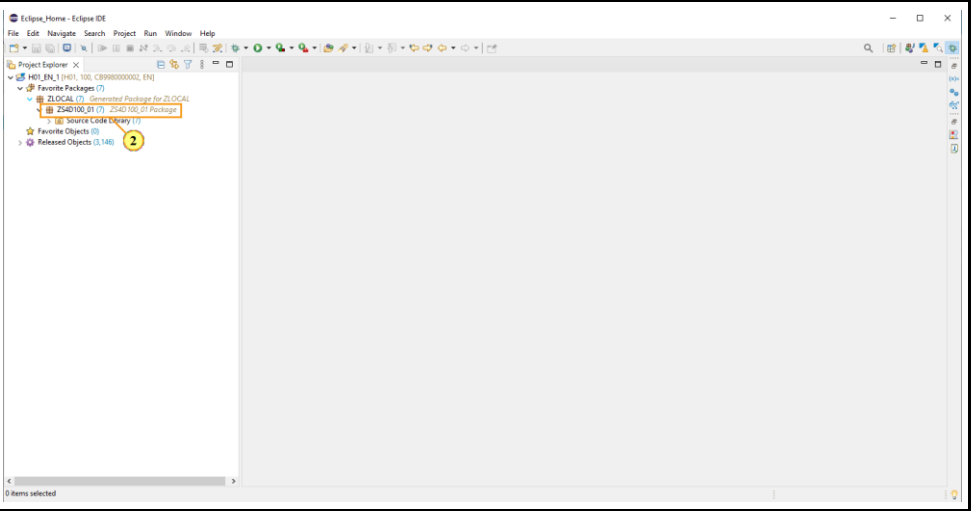
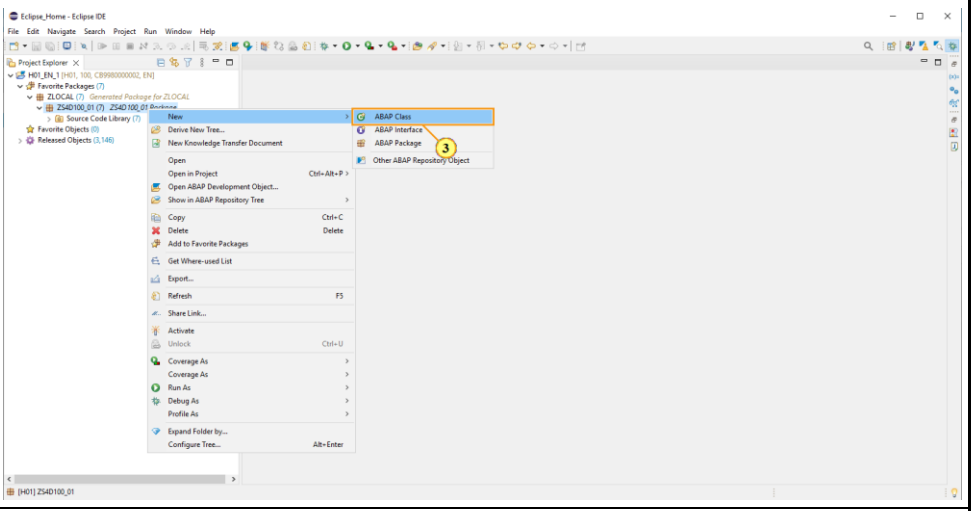
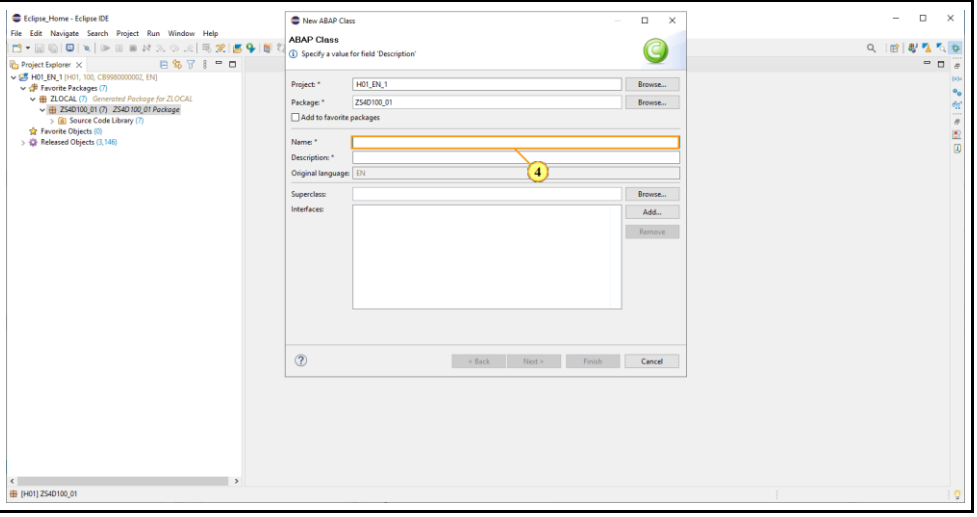
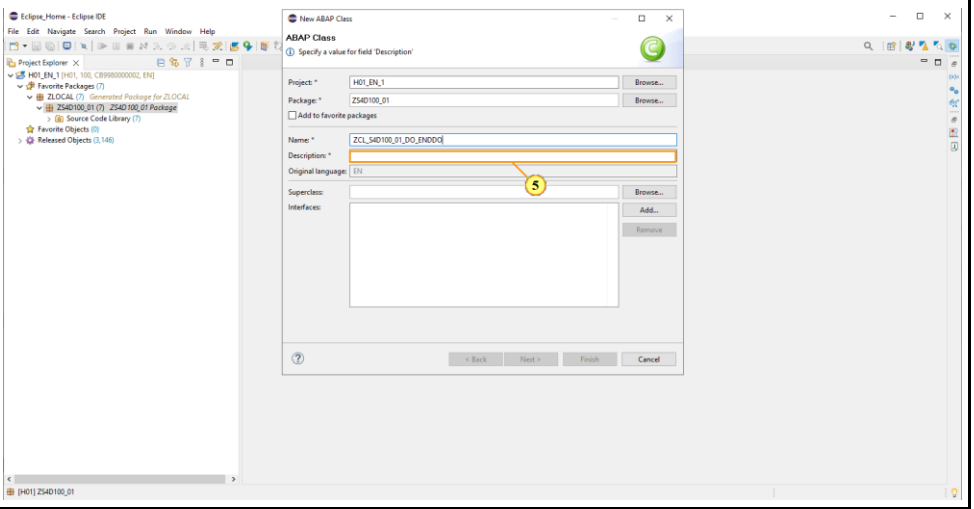
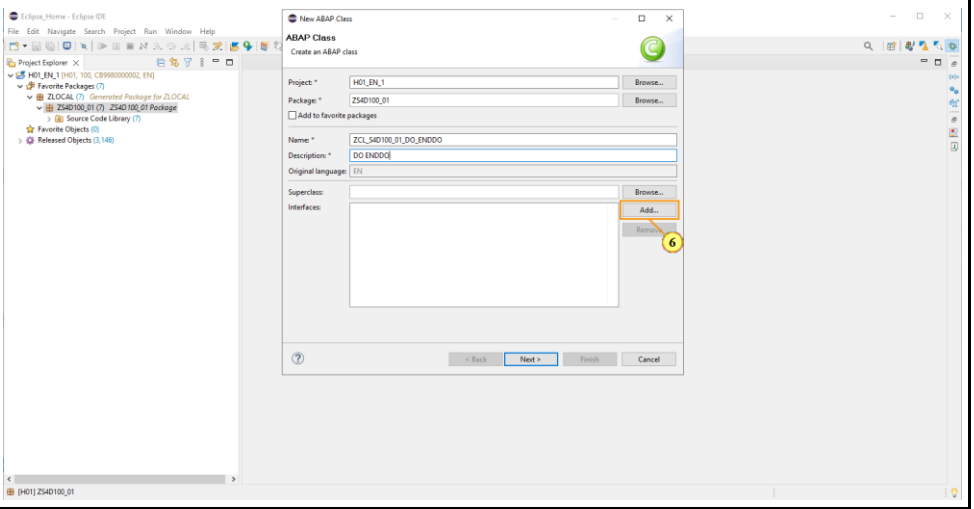
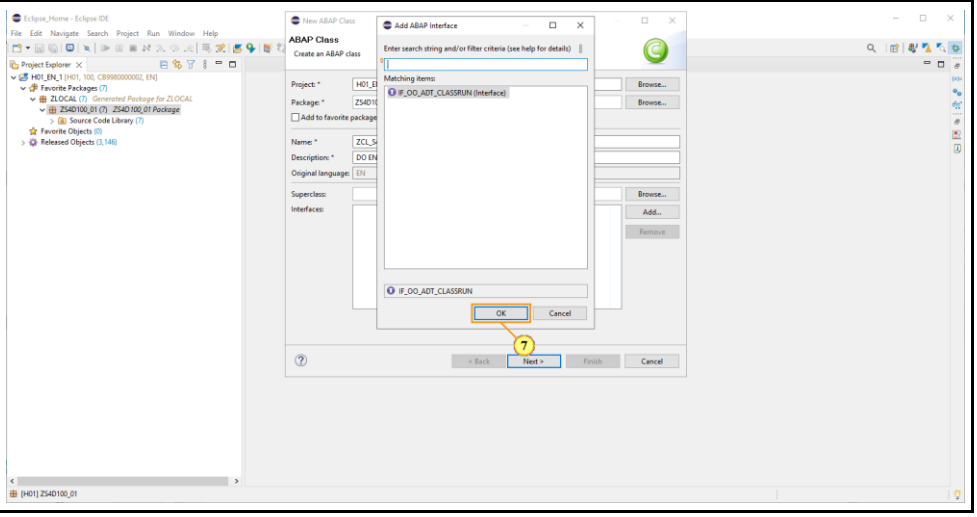
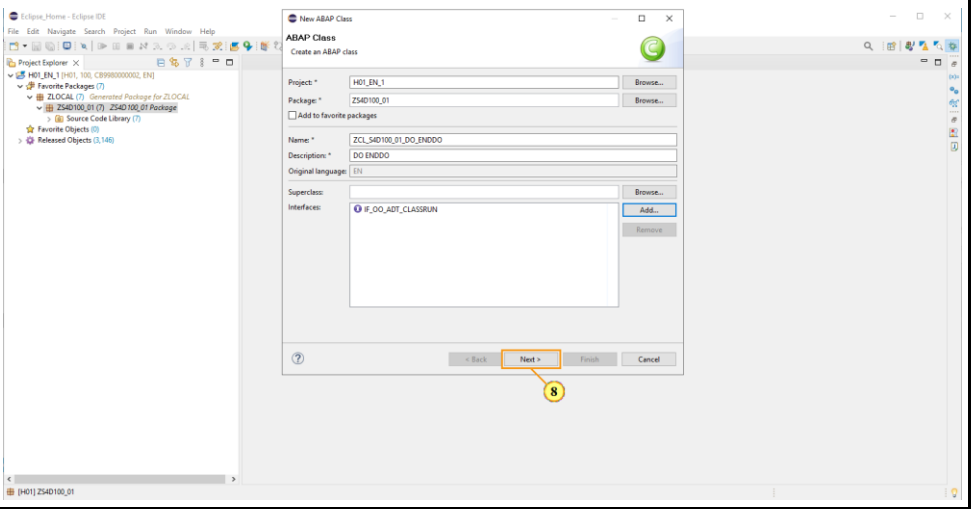
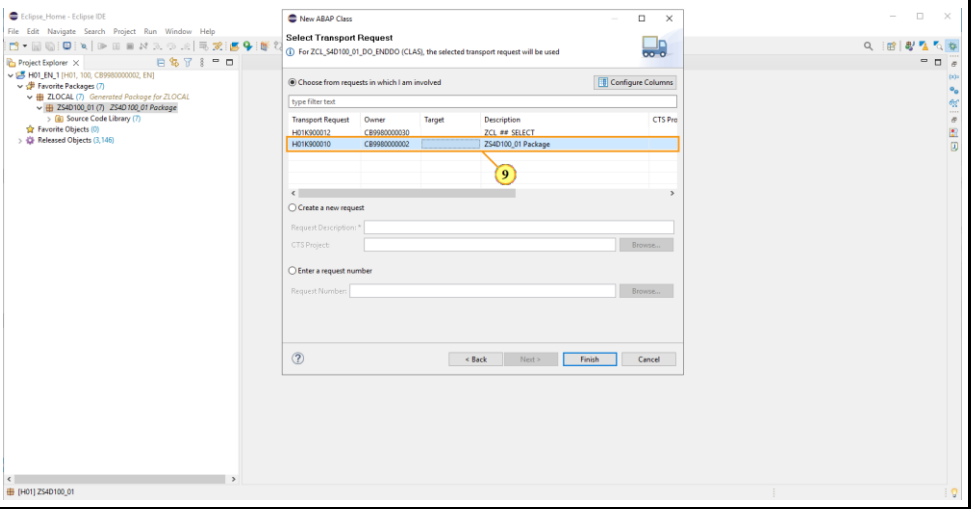
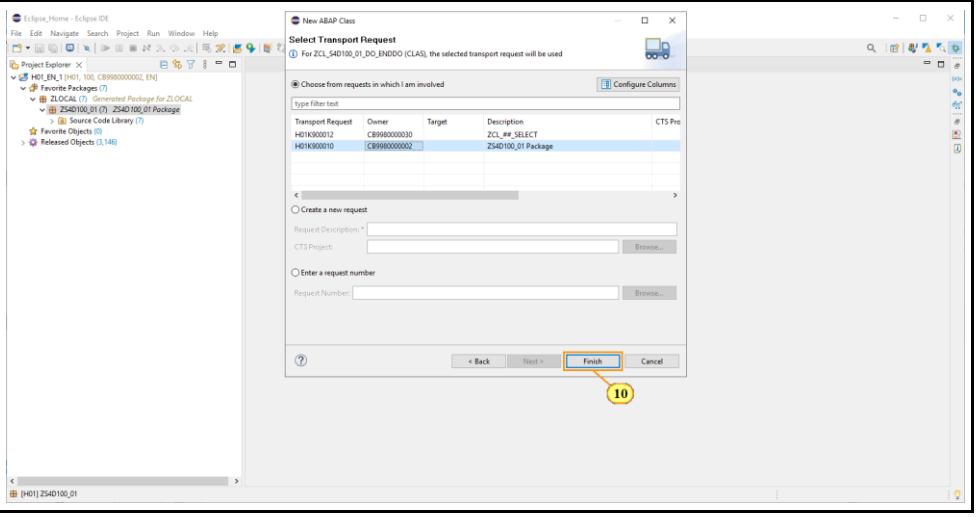
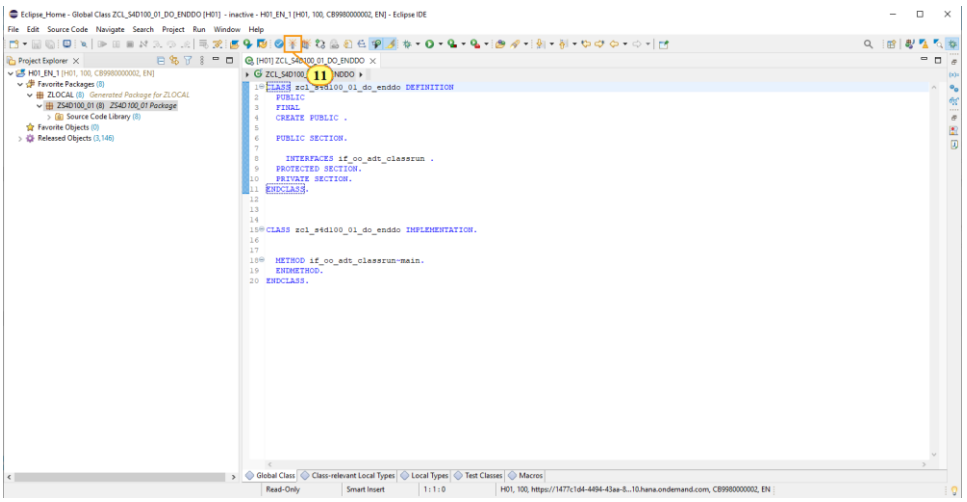
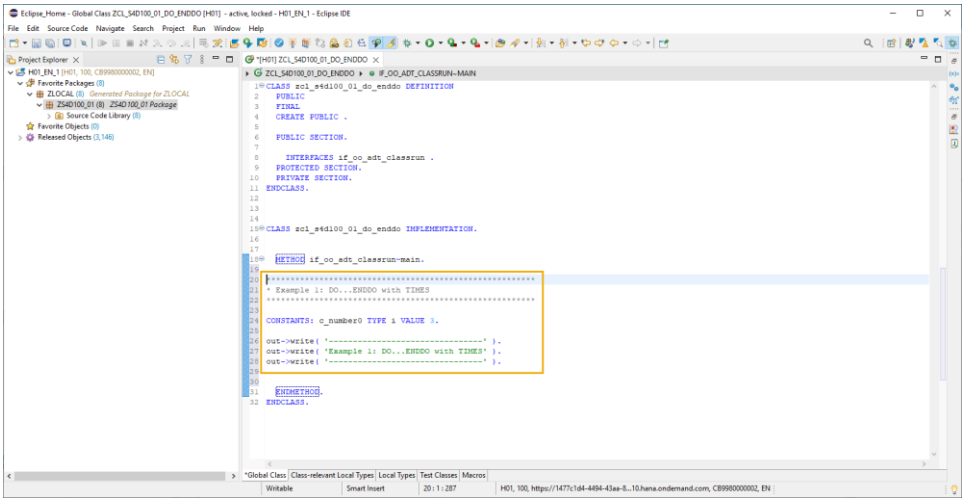
Implement a DO – ENDDO Loop

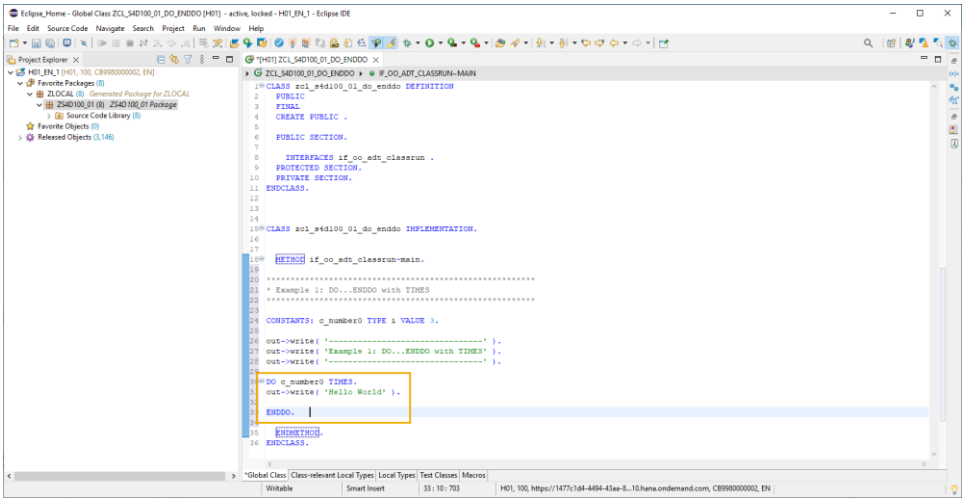
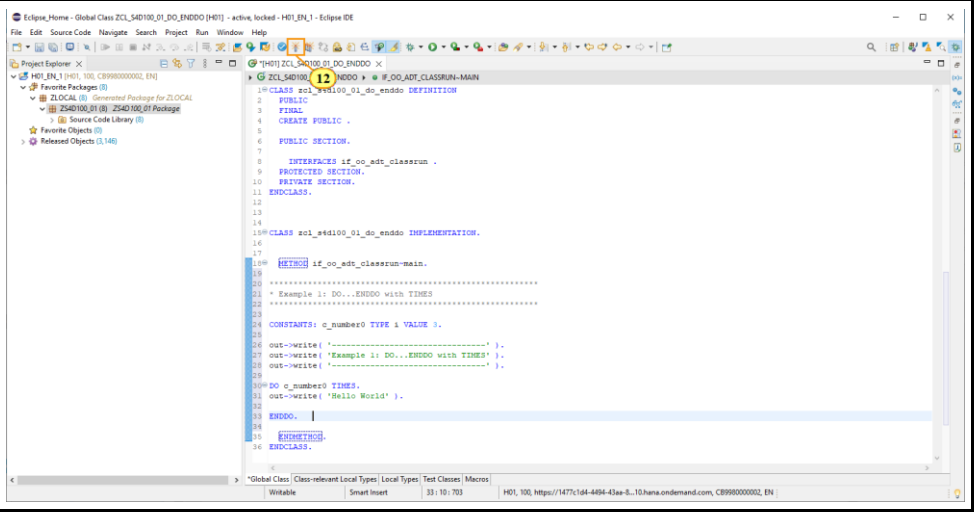
Explanation	Screenshot
ⓘ Using the DO . . ENDDO keywords users may wish to create a Loop with a specified number of Loops. To learn more about how to implement a DO – ENDDO loop, follow this interactive tutorial.	
ⓘ In the following steps, you will create a new ABAP class with the name ZCL_S4D100_01_DO_ENDDO and add it to your package and tie it to your transport request.	
1. Choose *H01_EN_1 > Favorite Packages > ZLOCAL*.	

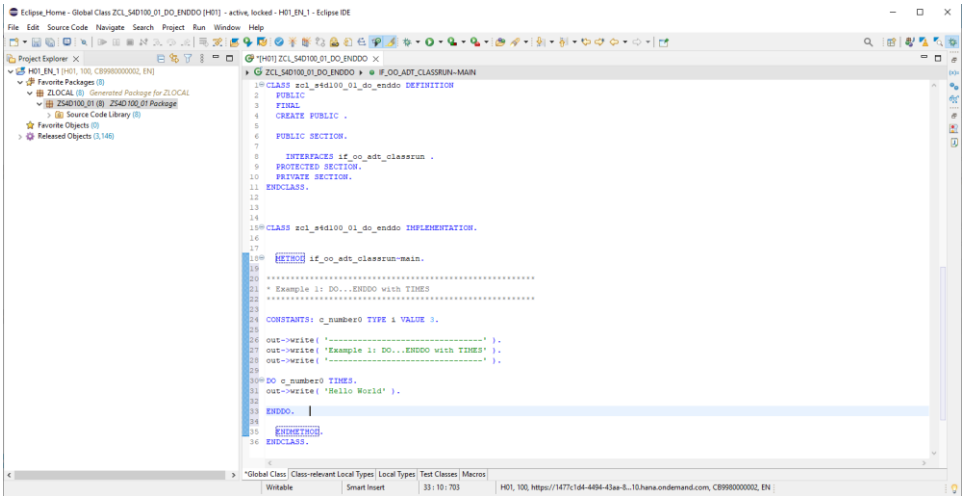
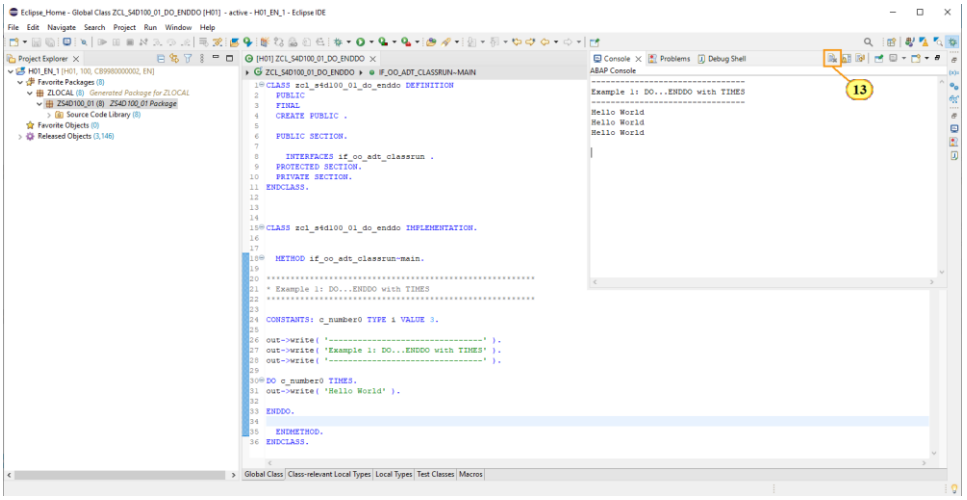
Explanation	Screenshot
2. Choose the *ZS4D100_01 (7) ZSD 100_01 Package* with the right-click.	
3. Choose *New* > *ABAP Class*.	
4. In the *Name* field, enter **ZCL_S4D100_01_DO_ENDDO**.	

Explanation	Screenshot
5. In the *Description* field, enter **DO ENDDO**.	
6. Choose *Add.*	
7. Choose *OK*.	

Explanation	Screenshot
8. Choose *Next.*	
9. Choose *H01K900010.*	
10. Choose *Finish*.	

Explanation	Screenshot
11. Choose *Activate*.	
ⓘ Enter the following code in to the code editor: ********** ********** ******** * Example 1: DO...ENDDO with TIMES ********** ********** ******** CONSTANTS: c_number0 TYPE i VALUE 3. out->write('---------- ----------'). out->write('Example 1: DO...ENDDO with TIMES'). out->write('---------- ----------'). For this tutorial, this step has been performed for you.	

Explanation	Screenshot
ⓘ Enter the following code in to the code editor: DO c_number0 TIMES. out->write('Hello World'). ENDDO. For this tutorial, this step has been performed for you.	
12. Choose *Activate*.	

Explanation	Screenshot
Please press **F9**.	
13. Choose *Clear Console*.	
ⓘ You have successfully implemented a DO-ENDDO Loop. This concludes the interactive tutorial.