Q Explain below listed principles with examples among SOLID principles.

● S: Single-responsibility principle : The name itself suggest that the “class should be having one and only one responsibility”. What does it mean? Well let’s take the class A which does the following operations.

* Open a database connection
* Fetch data from database
* Write the data in an external file

The issue with this class is that it handles lot of operations. Suppose any of the following change happens in future.

* New database
* Adopt ORM to manage queries on database
* Change in the output structure

So in all the cases the above class would be changed. Which might affect the implementation of the other two operations as well. So ideally according to SRP there should be three classes each having the single responsibility.

● O: Open-closed principle : This principle suggests that “classes should be open for extension but closed for modification”. What is means is that if the class A is written by the developer AA, and if the developer BB wants some modification on that then developer BB should be easily do that by extending class A, but not by modifying class A.

The easy example would be the RecyclerView.Adapter class. Developers can easily extend this class and create their own custom adapter with custom behaviour without modifying the existing RecyclerView.Adapter class.

● I: Interface segregation principle : This principle suggests that “many client specific interfaces are better than one general interface”. This is the first principle which is applied on interface, all the above three principles applies on classes.