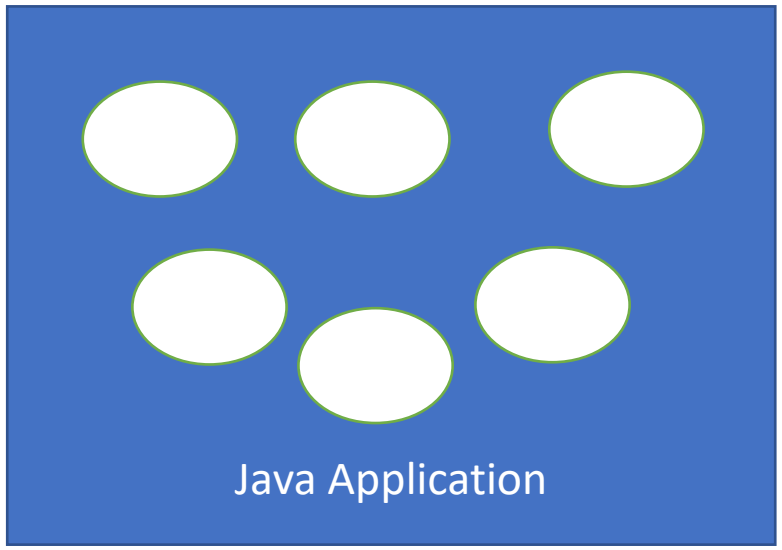
In a Java application, data is primarily stored in the form of Objects

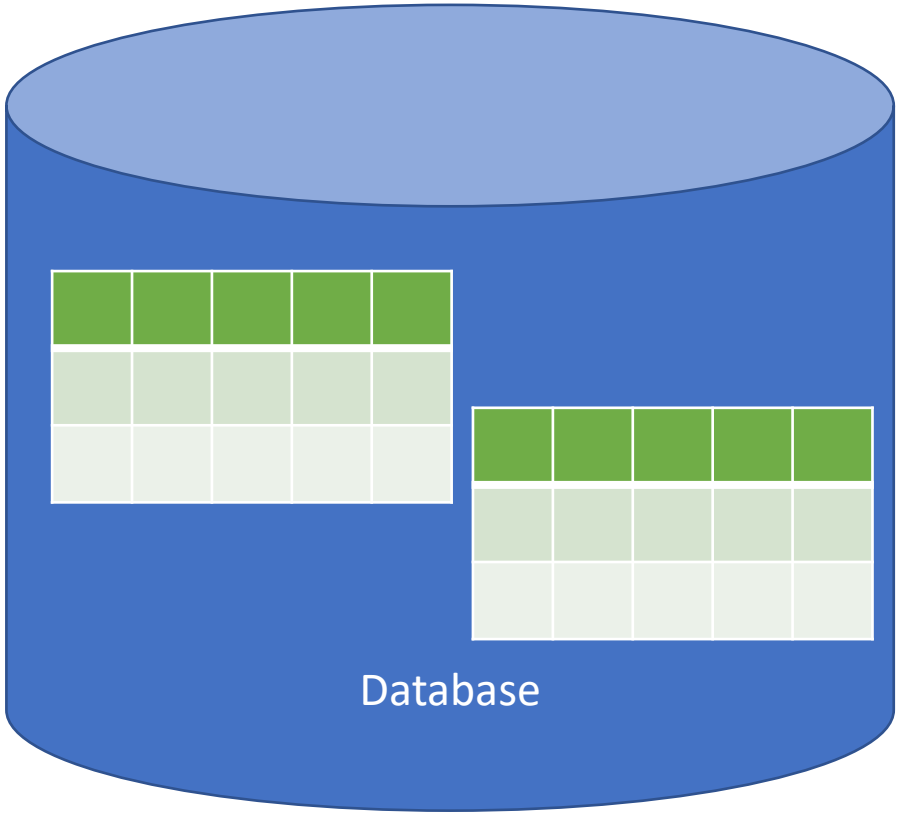
For ex:
Employee data might be stored in an Employee object
Customer data might be stored in a Customer object

Objects are basically transient in nature, which means an Object is temporary. It will remain in the memory as long as you have reference to the same, after which it will be eligible for GC



Database primarily comprises of Tables where data is stored and maintained.

Data in Tables is stored in the form of rows and columns



Cont'd…

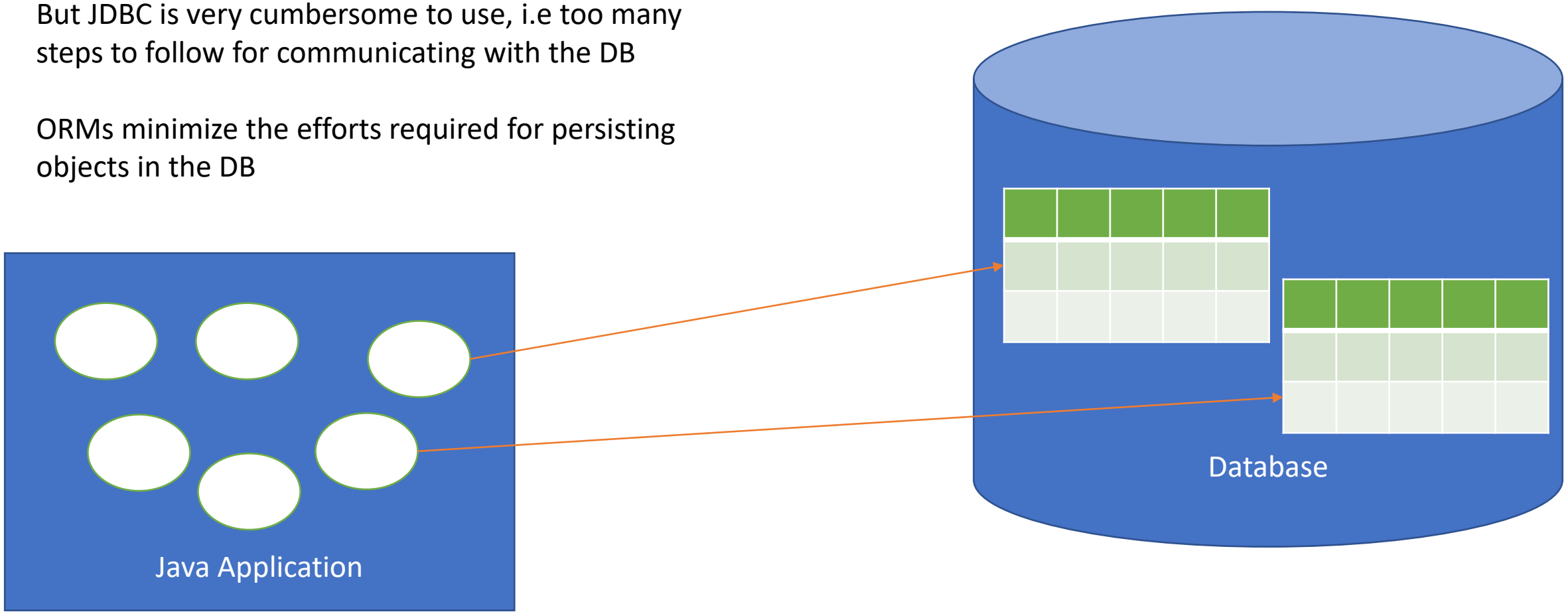
We need a way by which we can persist our object's data in the Database

We already have one option in Java called as JDBC. But JDBC is very cumbersome to use, i.e too many steps to follow for communicating with the DB

ORMs minimize the efforts required for persisting objects in the DB

Database primarily comprises of Tables where data is stored and maintained.

Data in Tables is stored in the form of rows and columns

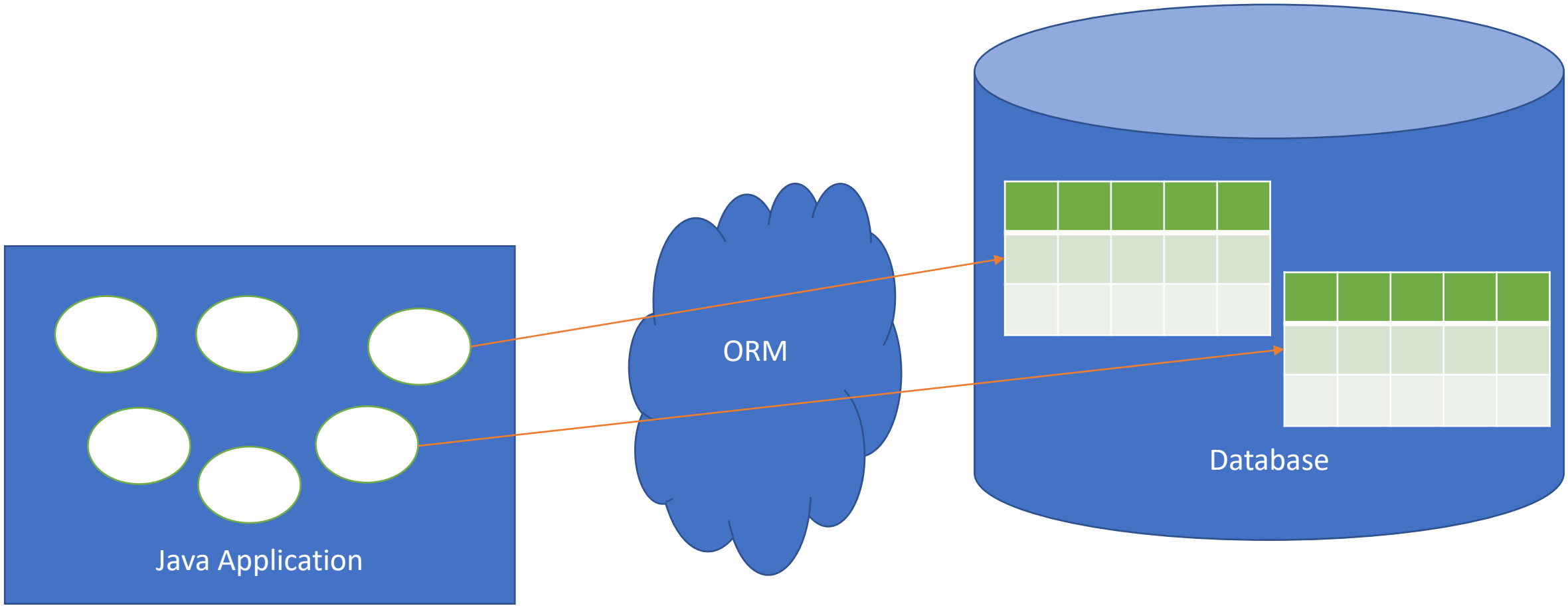


Cont’d…

ORMs minimize the efforts required for persisting objects in the DB

Database primarily comprises of Tables where data is stored and maintained.

Data in Tables is stored in the form of rows and columns



One to One Association

TBL_EMPLOYEE

ID	NAME	SALARY	DOJ	ADDR_ID	
1001	MAJRUL	10000	01-02-2022	111	

Two choices:

Should we keep ADDR_ID in Employee Table?
OR
Should we keep EMP_ID in Address Table?

TBL_ADDRESS

ID	CITY	STATE	PINCODE		
111	MUMBAI	MH	400083		
222	PUNE	MH	400083		
333	DELHI	DL	110001		