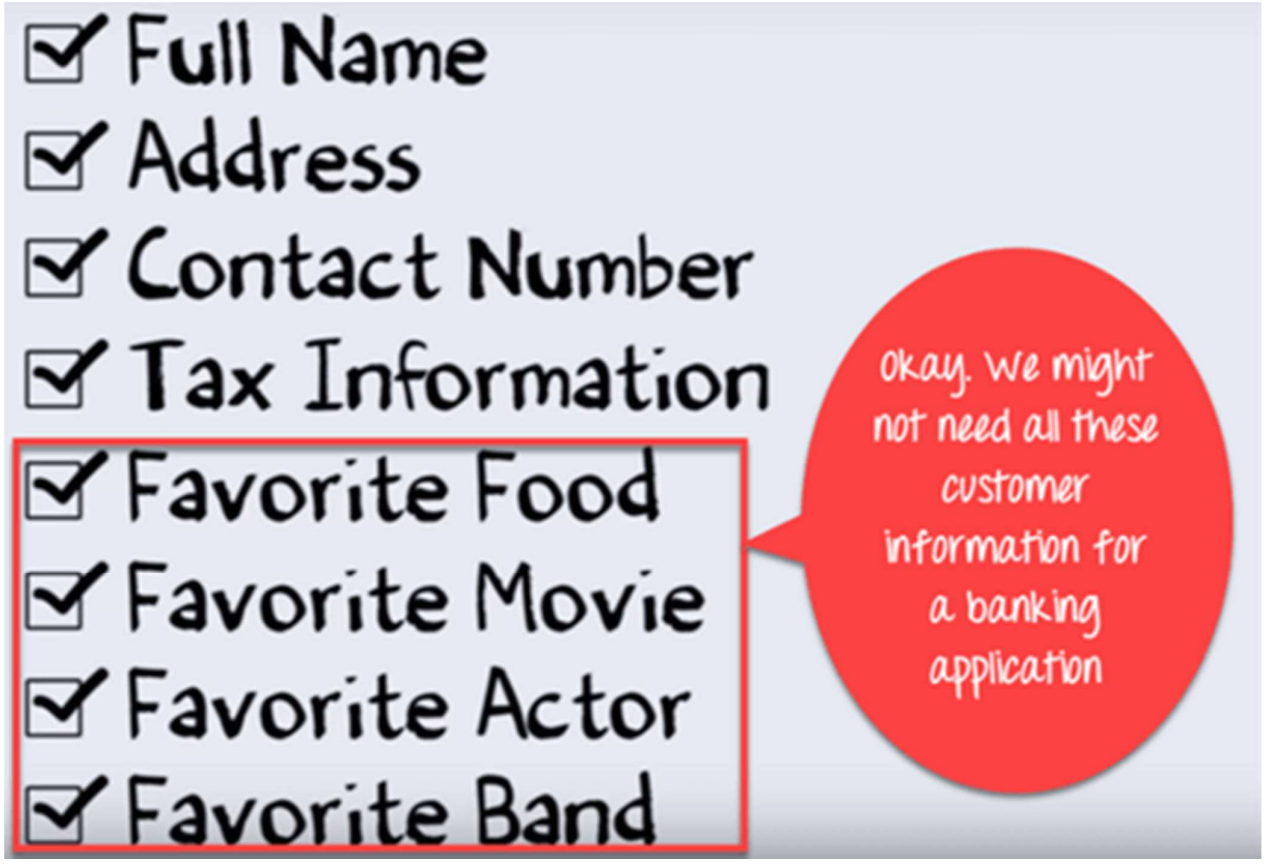
ABSTRACTION

ABSTRACTION is the concept of object-oriented programming that "shows" only essential attributes and "hides" unnecessary information.

It helps in reducing programming complexity and efforts.

An example

Let us see a banking application.



Need to select only the useful information for the banking application from that pool. Data like name, address, tax information, etc. make sense for a banking application.

An advantage can be the same information once extracted can be used for a wide range of applications. For instance, you can use the same data for hospital application, job portal application, a Government database, etc. with little or no modification. Hence, it becomes your Master Data.

Difference between Abstraction and Encapsulation

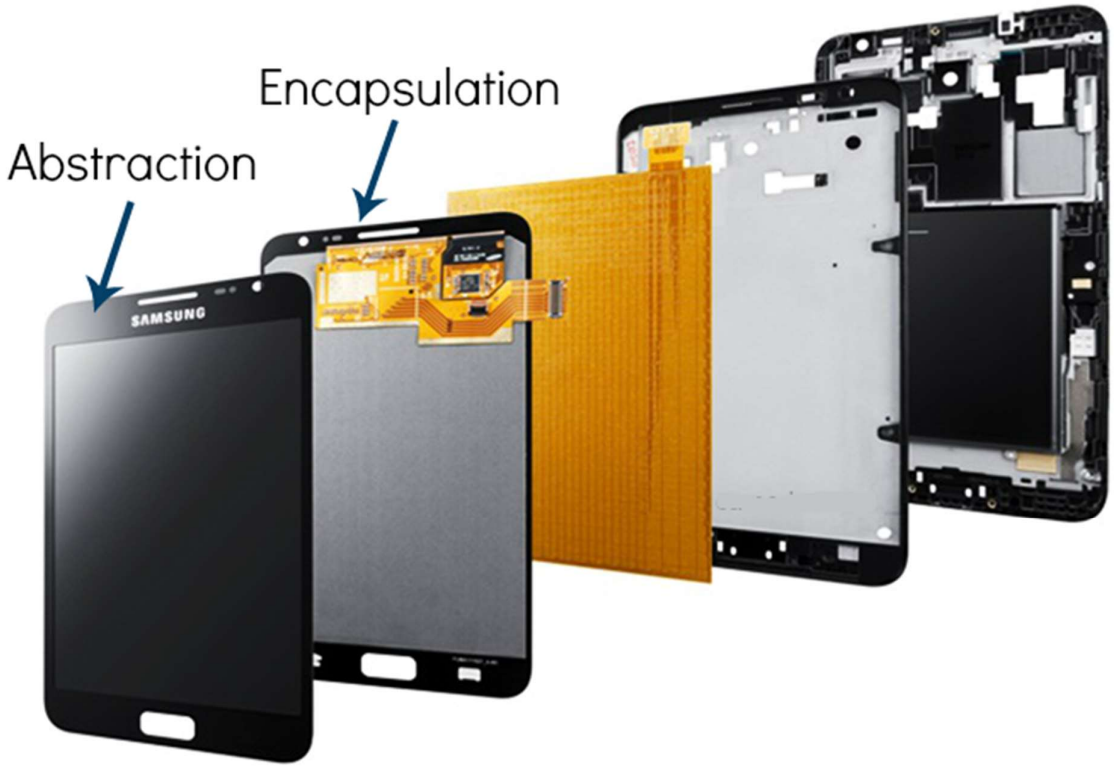
Abstraction	Encapsulation
Solves issues at the design level	Solves in implantation level.
It is about data hiding unwanted details while showing most essential information.	Binds the code and data into a single unit.
Allows Focusing on what the information object must contain.	Hides the internal details or mechanics of how an object does something for security reasons.


Using Abstraction
Without Abstraction

MAIN ADVANTAGE

Abstraction helps to reduce the complexity of the design and implementation process of software.

Example to show Encapsulation and Abstraction



Real World Example of Abstraction

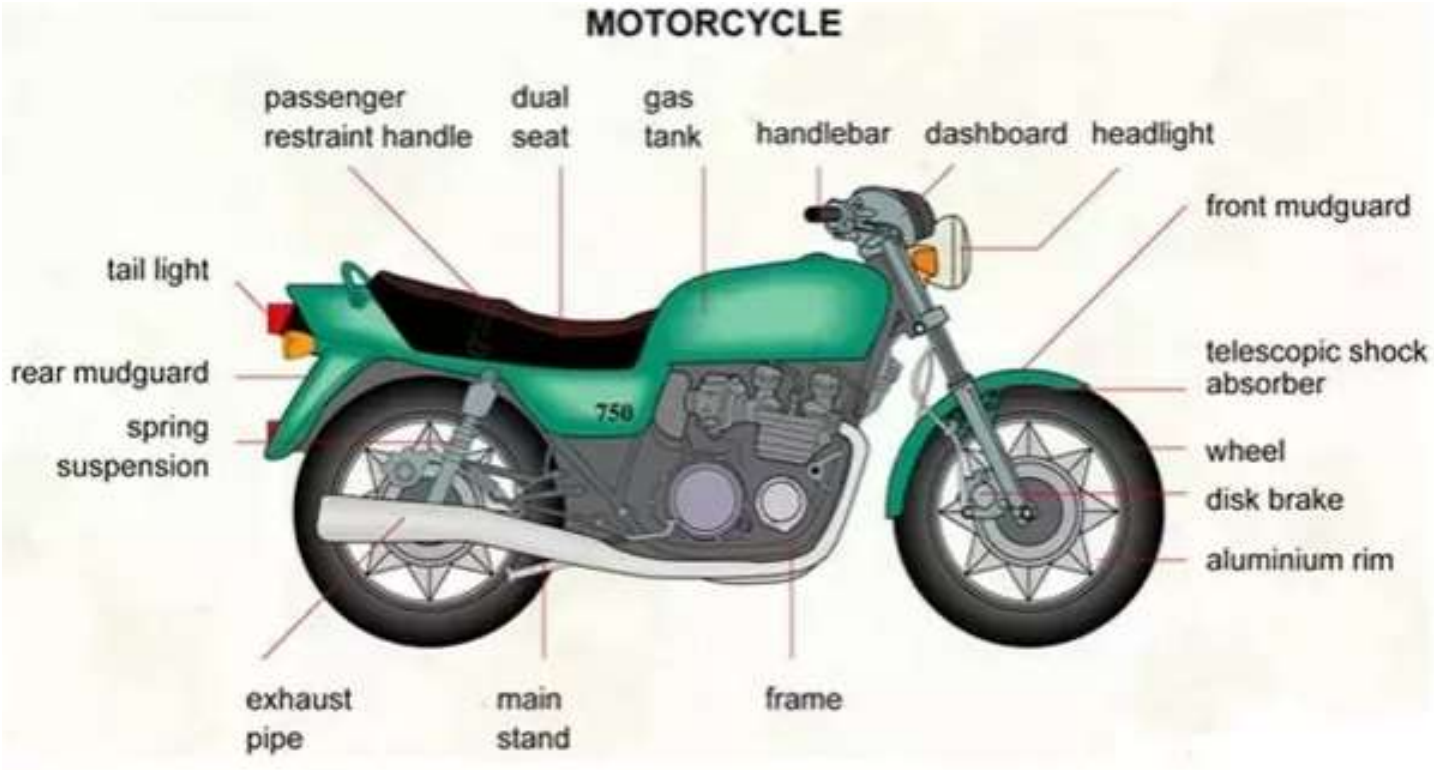
Consider a Car, which abstracts the internal details and exposes to the driver only those details that are relevant to the interaction of the driver with the Car.

Real Life Example of Abstraction

Consider an ATM Machine; All are performing operations on the ATM machine like cash withdrawal, money transfer, retrieve mini-statement...etc. but we can't know internal details about ATM.

Summary

Abstraction means hiding the complexity and showing the easier form of the system.



NEXT LECTURE: POLYMORPHISM