**ABSTRACTION**

**Abstraction** in its main sense is a conceptual process by which general [rules](https://en.wikipedia.org/wiki/Rule_of_inference) and [concepts](https://en.wikipedia.org/wiki/Concept) are derived from the usage and classification of specific examples, literal ("real" or "[concrete](https://en.wikipedia.org/wiki/Abstract_and_concrete)") signifiers, [first principles](https://en.wikipedia.org/wiki/First_principle), or other methods. "An abstraction" is the product of this process — a concept that acts as a super-categorical noun for all subordinate concepts, and connects any related concepts as a *group*, *field*, or *category* Conceptual abstractions may be formed by filtering the [information](https://en.wikipedia.org/wiki/Information) content of a [concept](https://en.wikipedia.org/wiki/Concept) or an observable [phenomenon](https://en.wikipedia.org/wiki/Phenomenon), selecting only the aspects which are relevant for a particular purpose. For example, abstracting a leather soccer ball to the more general idea of a [ball](https://en.wikipedia.org/wiki/Ball) selects only the information on general ball attributes and behavior, eliminating the other characteristics of that particular ball  In a [type–token distinction](https://en.wikipedia.org/wiki/Type%E2%80%93token_distinction), a type (e.g., a 'ball') is more abstract than its tokens (e.g., 'that leather soccer ball').

Abstraction in its secondary use is a [material process](https://en.wikipedia.org/wiki/Abstraction#Material_process)  discussed in the [themes below](https://en.wikipedia.org/wiki/Abstraction#Themes).

Example

:( Without abstract method)

class Employee extends Person {

 private String empCode;

 public String getEmpCode() {

 return empCode;

 }

 public void setEmpCode(String empCode) {

 this.empCode = empCode;

 }

}

abstract class Person {

 private String name;

 public String getName() {

 return name;

 }

 public void setName(String name) {

 this.name = name;

 }

}

public class Main{

 public static void main(String args[]){

 //INSTIATING AN ABSTRACT CLASS GIVES COMPILE TIME ERROR

 //Person p = new Person() ;

 //THIS REFERENCE VARIABLE CAN ACESS ONLY THOSE METHOD WHICH ARE OVERRIDDEN

 Person person = new Employee();

 person.setName("Jatin Kansagara");

 System.out.println(person.getName());

 }

}