# GET SMART: WITH JAVA PROGRAMMING

Yaman Omar Alashqar

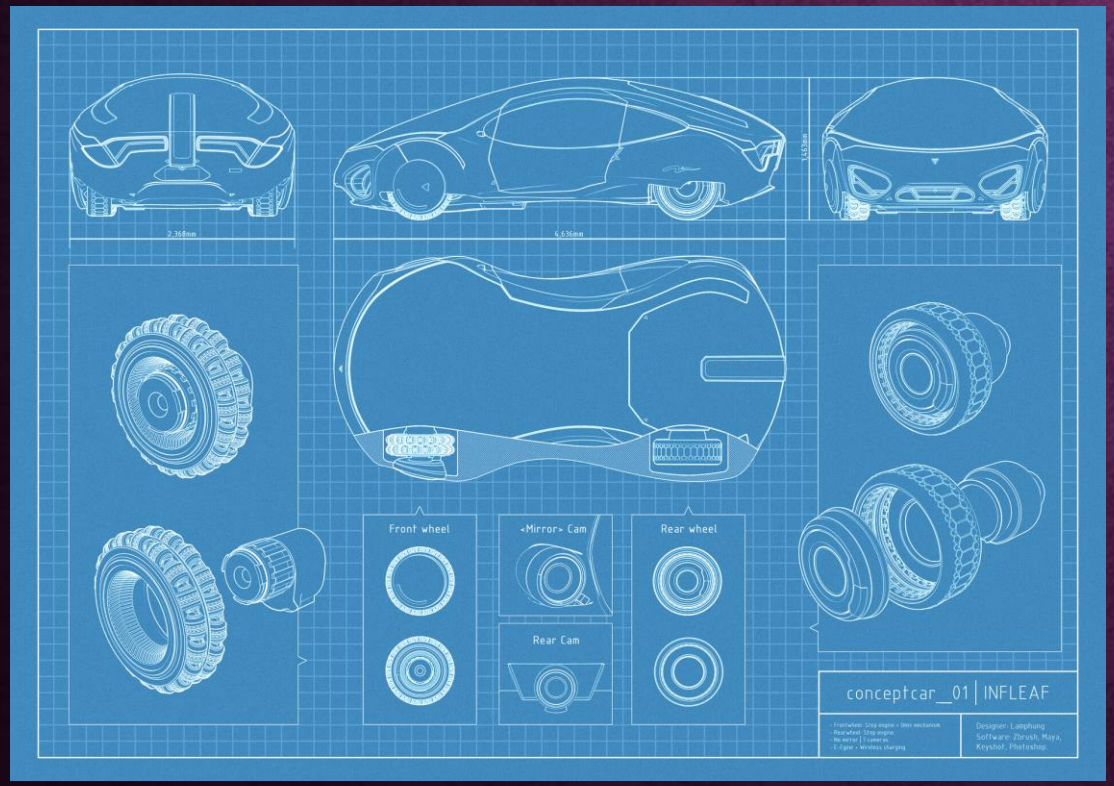
System.out.println(“WELCOME TO THIS COURSE”);

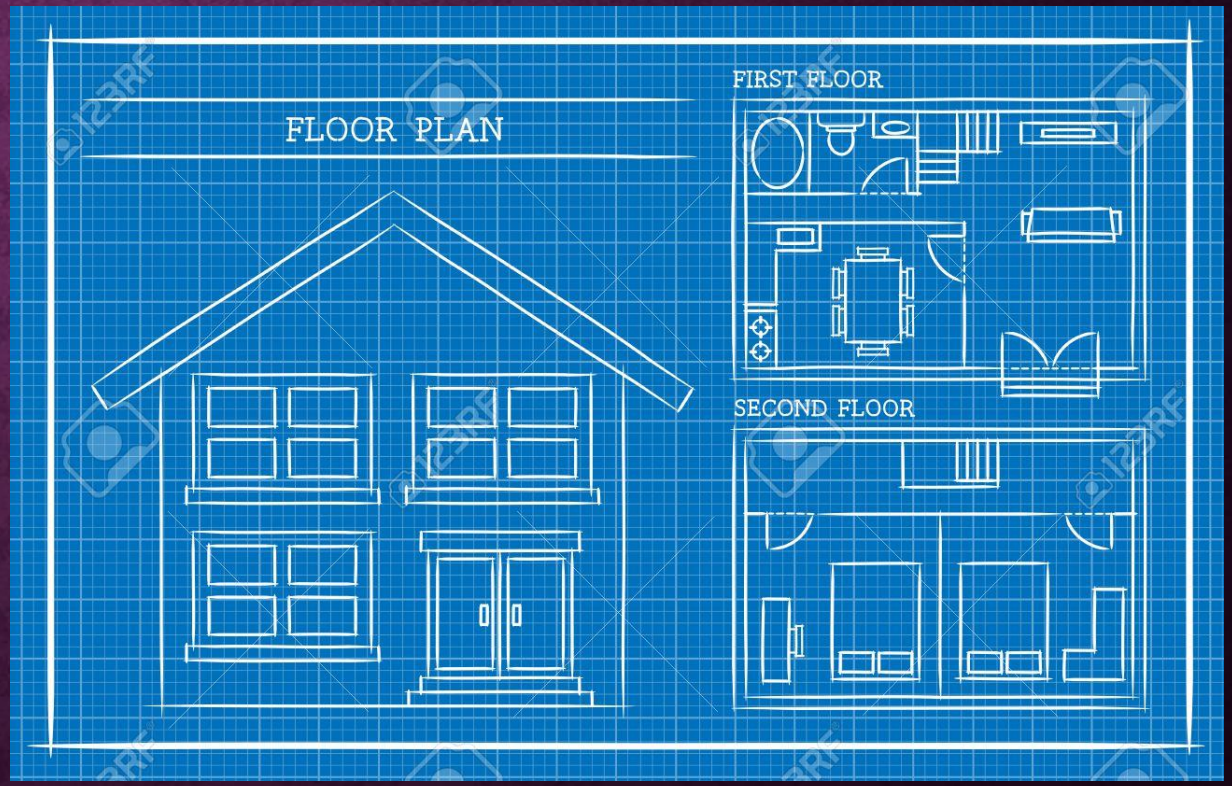
# GET SMART: WITH JAVA PROGRAMMING

# INTRODUCTION TO CLASSES

All programs require one or more classes that act as a model for the world.

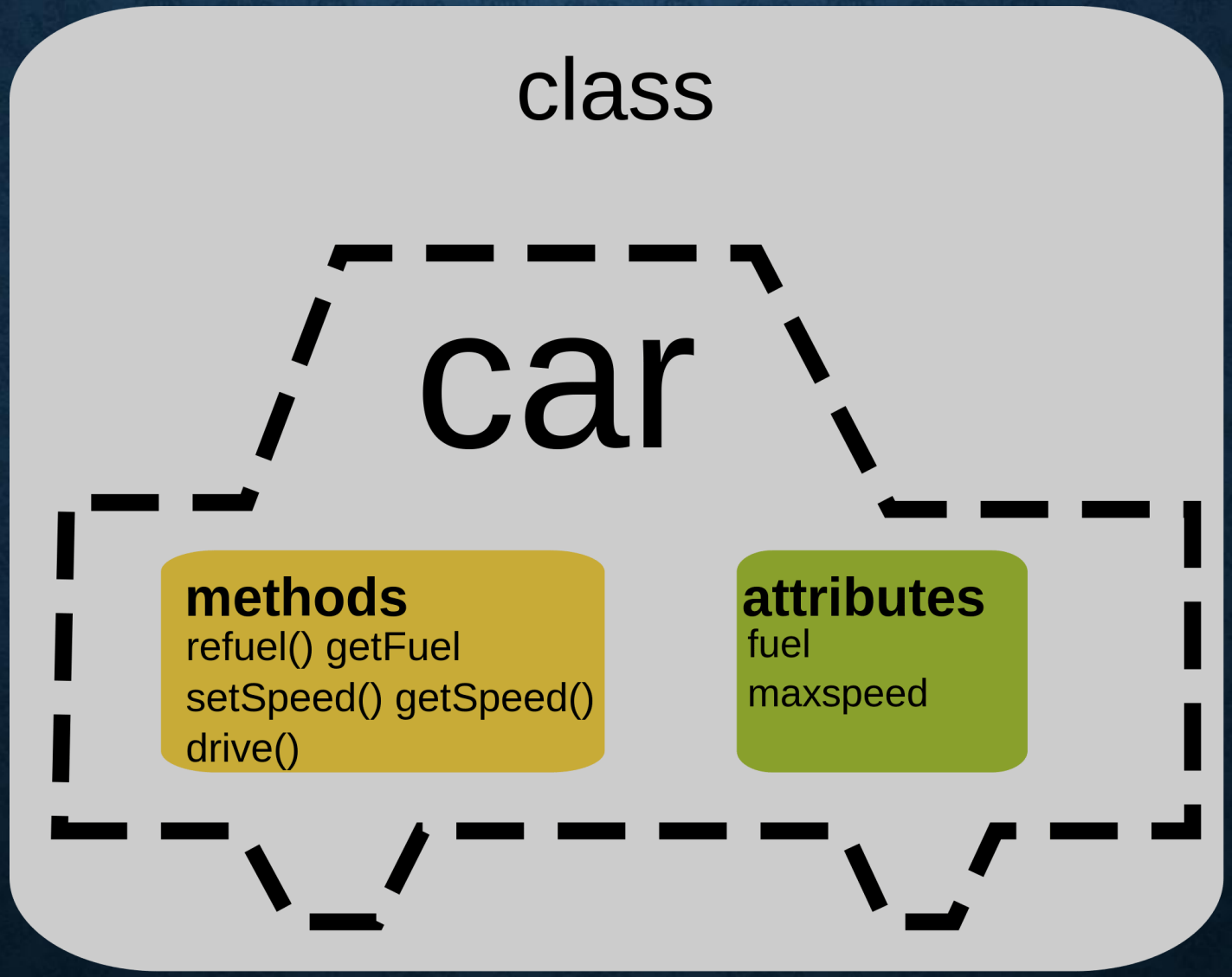
# Blueprints


Front wheel
‹Mirror› Cam
Rear wheel
Rear Cam
conceptcar_01 | INFLEAF


FLOOR PLAN
FIRST FLOOR
SECOND FLOOR

For example, a program to track student test scores might have
- Student,
- Course,
- and Grade classes.

We represent each student as an instance, or an object,
of the Student class.


class
car
methods
refuel() getFuel
setSpeed() getSpeed()
drive()
attributes
fuel
maxspeed

What is an object?

In code, we combine a group of related variables and methods (functions) into a unit, we called that unit an object.

What are classes?

Classes are the blueprints that define the behavior and information our objects will contain. They let us manufacture and instantiate new instances.

This is object-oriented programming because programs are built around objects and their interactions.
An object contains state and behavior.

An object-oriented programming is a way programming which enables programmers to think like they are working with real-life entities
(a thing with distinct and independent existence)
or objects

- OOP provides a clear modular structure for programs
- It aims to implement real life scenario
- It is easy to maintain and modify existing code
- New objects can be created with different behavior & attributes
- Implementation details are hidden from other modules
- Other modules has a clearly defined interface

# Basic Terminologies

Class:
It is a template or blue print about the capability of what an object can do.

Object:
It's the instance of a class/ it's the working entity of a class.

Method:
The behaviors of a class. It tells what a object can do.

Instance:
Object and Instance both are same with small difference.

# Basic Terminologies

A blueprint for a house design is like a class description.
All the houses built from that blueprint are objects of that class.
A given house is an instance.

Classes are a blueprint for objects.
Blueprints detail the general structure.

Let's review with another example, a savings account at a bank.

What should a savings account know?
The balance of money available.

What should a savings account do?
Deposit money & Withdraw money

Imagine two people have accounts that are instances of the SavingsAccount class.
They share behavior (how they deposit and withdraw)
but have individual state (their balances),
these accounts are separate entities.

Understand the difference between a class and object

A class is a blueprint / an idea / an description / a definition

it describes what something is

its not the thing its self .. its a well defined idea (like a blueprint for a house)

| class | objects |
| --- | --- |
| Fruit | Apple<br>Banana<br>Mango |

Another example:

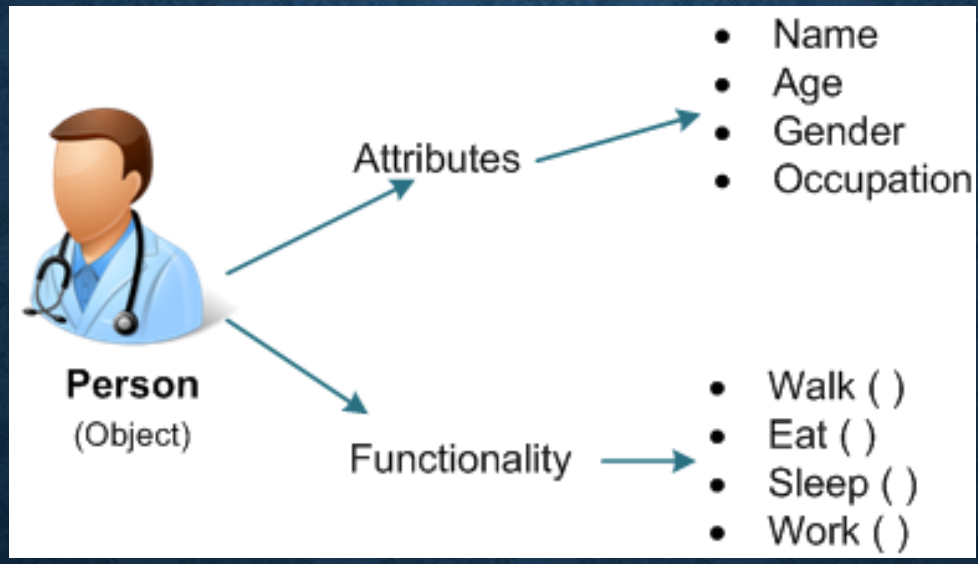
| class | objects |
| --- | --- |
| Car | Volvo<br>Audi<br>Toyota |

Classes describes two main things:

Attributes and behavior => (Properties - methods)

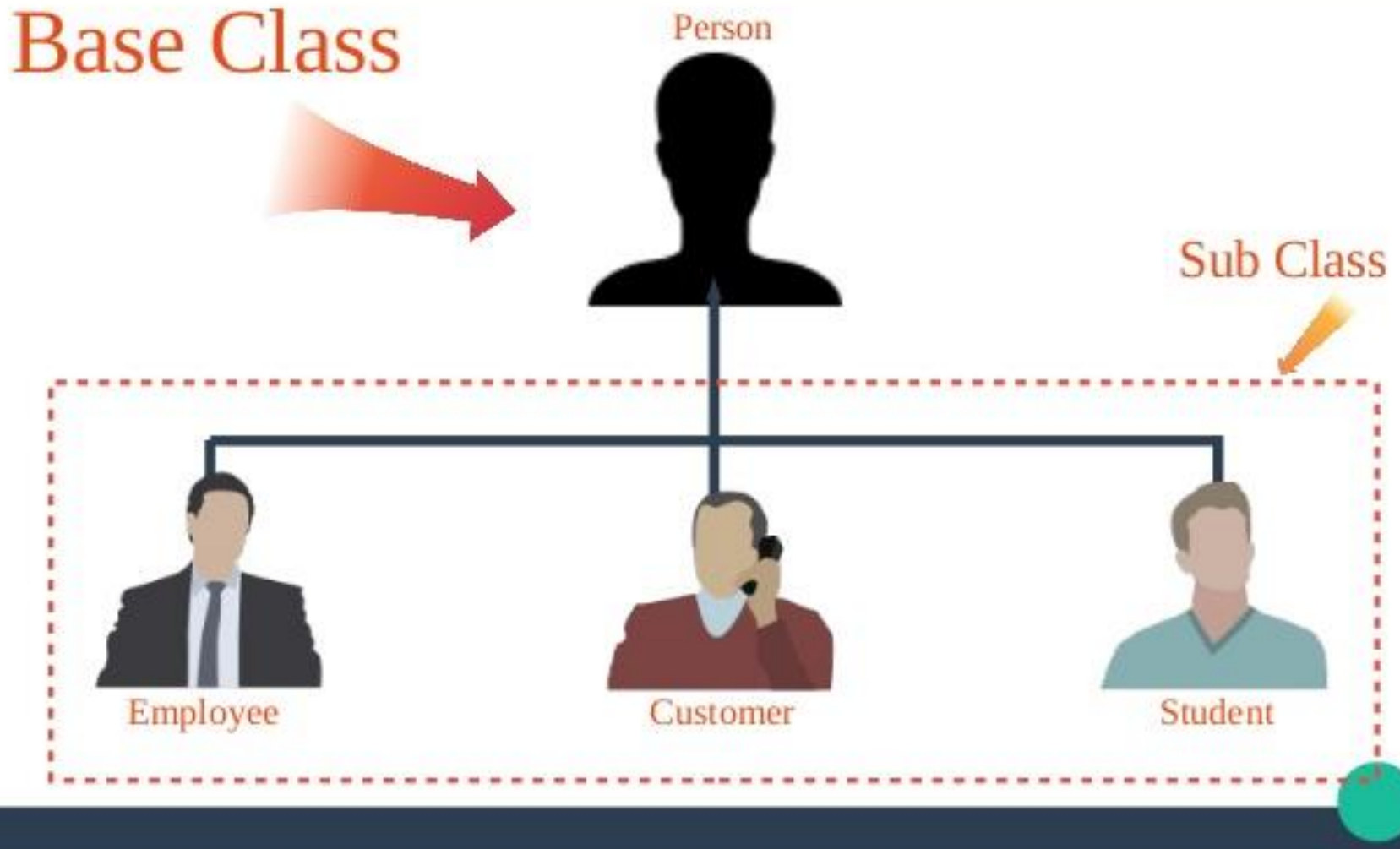
class person:

Attributes: name, height, weight, gender, age
Behavior: walk() , jump(), sleep();

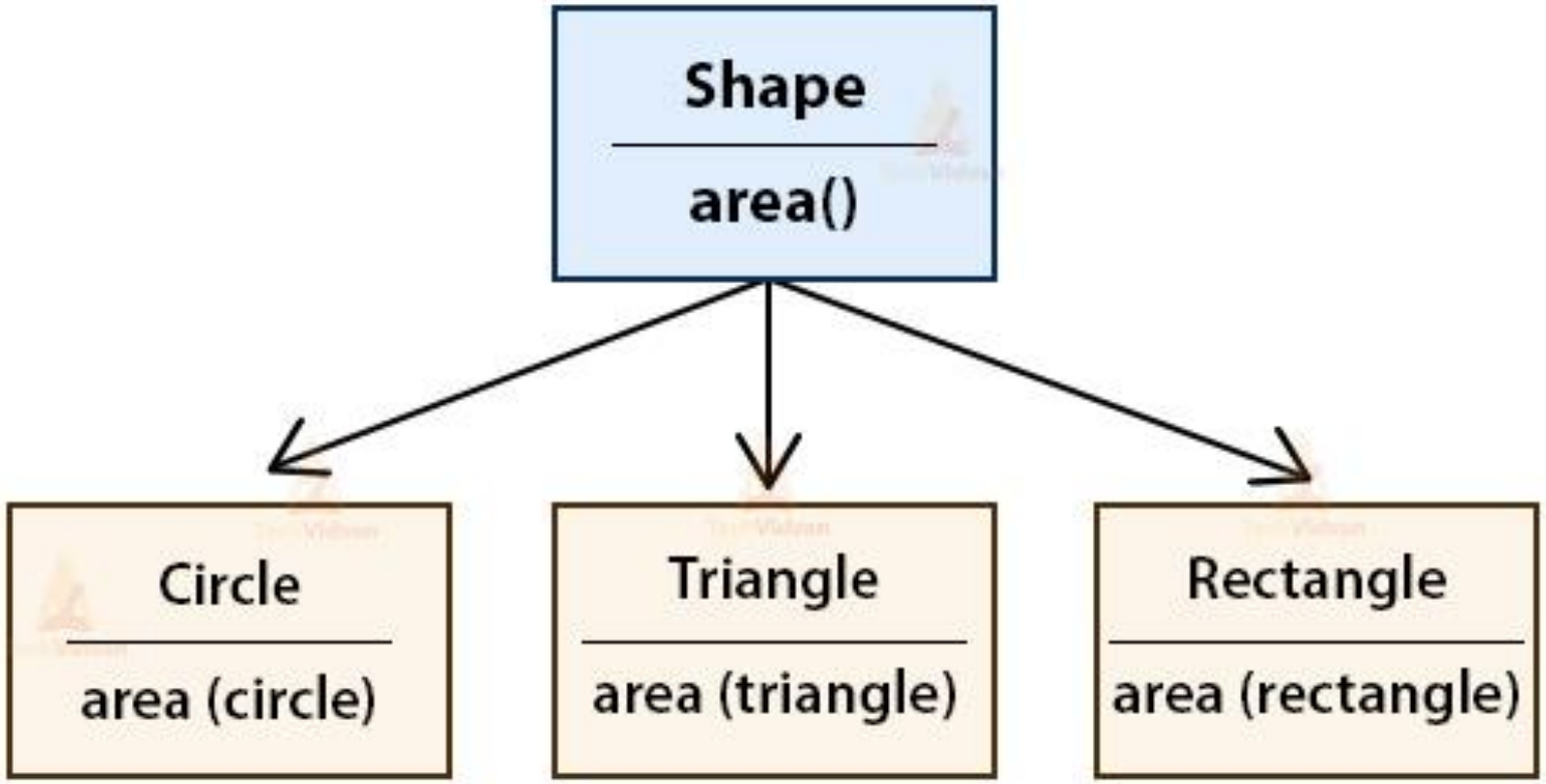



The class is the idea and the object is the thing itself
we create objects based on class
so you can create multiple objects from 1 class

# Inheritance: Example



# Example of Polymorphism in Java



```
abstract public class Remote {
    abstract public boolean powerSwitch();
    abstract public int increaseVolume();
    abstract public int decreaseVolume();
    abstract public double channelTuning(int channel);

}
```

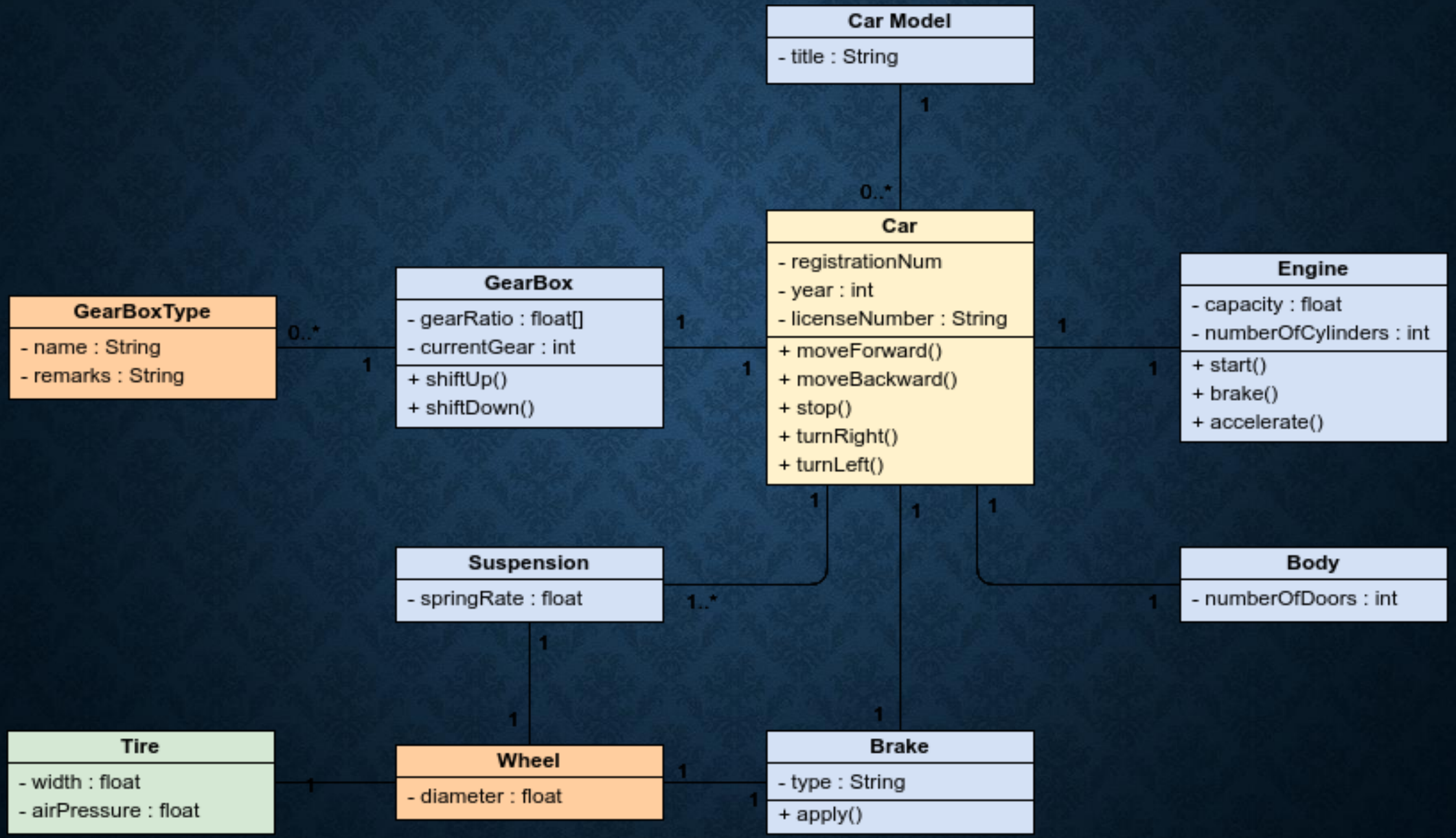
```
class Application {
 public static void main(String args[]){
      Remote remoteTelevision =
                    new Television(10.5,7,9);

     remoteTelevision.powerSwitch();
     remoteTelevision.decreaseVolume();
     remoteTelevision.increaseVolume();
     remoteTelevision.channelTuning(2);
```

```
class Television extends Remote{
    private double width
    private double height
    private double screenSize
    private int maxVolume
    private int volume
    private boolean power
public Television(double width,
          double height,
          double screenSize){
         this.width=width;
         this.height=height;
         this.screenSize=screenSize;

public double channelTuning(int channel){
         switch(channel){
           case 1: return 34.56;
           case 2: return 54.89;
           case 3: return 73.89;
           case 4: return 94.98;
         } return 0;
        }

public int decreaseVolume(){
         if(0<volume) volume--;
         return volume;
        }
public void powerSwitch(){
         this.power = !power;
        }
```

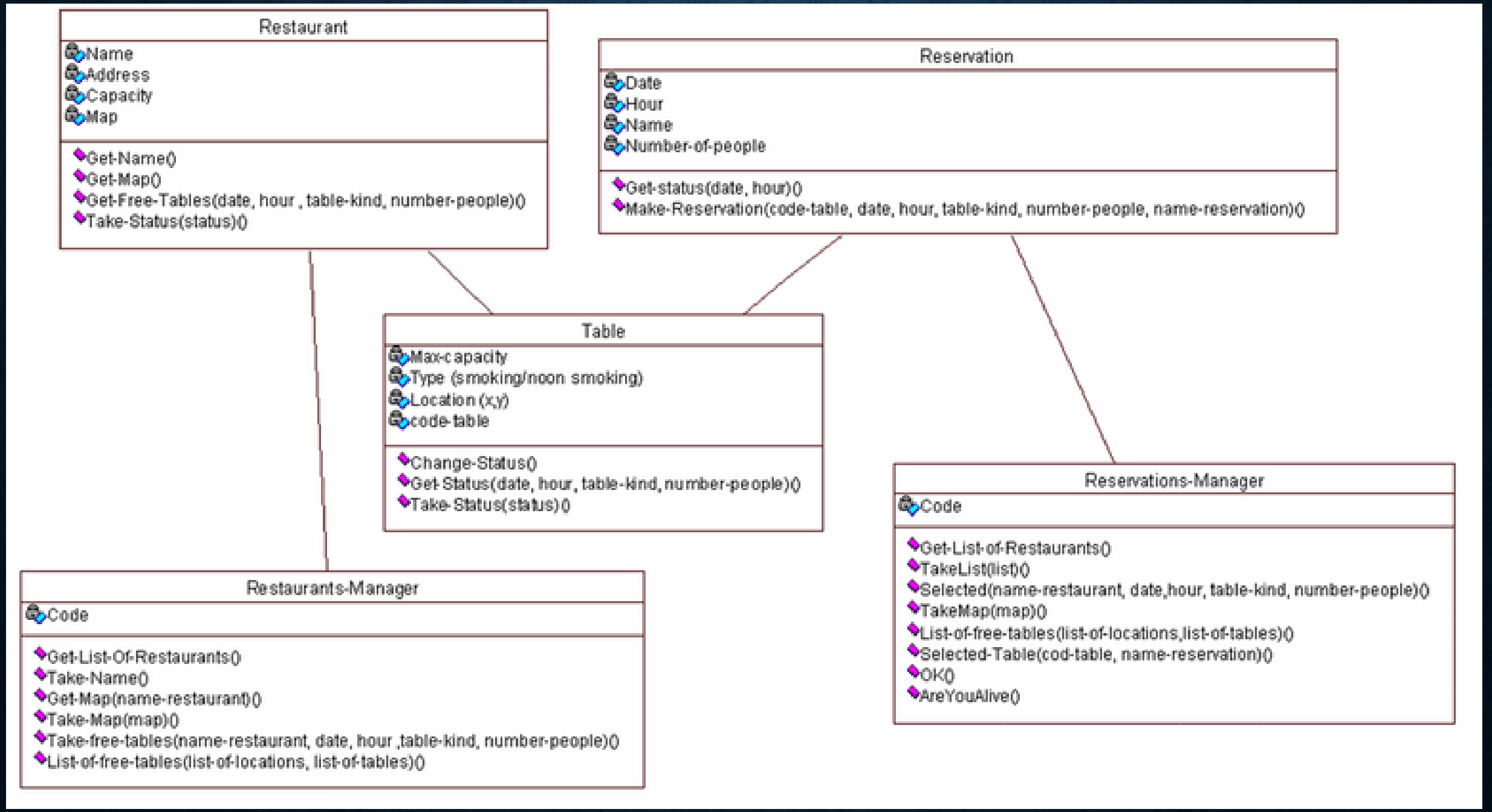
There are many different ways that your computer can connect to a local network (Ethernet, Wi-Fi, dial-up modem, etc.),

but your Web browser doesn't generally have to worry about which of these you're using at any given time, because lower-level software provides a common abstraction that your browser can rely on

. So "connection to the network" is the abstraction,
and Ethernet and Wi-Fi and so on are implementations of that abstraction




Building complex system by splitting the complexity level by level

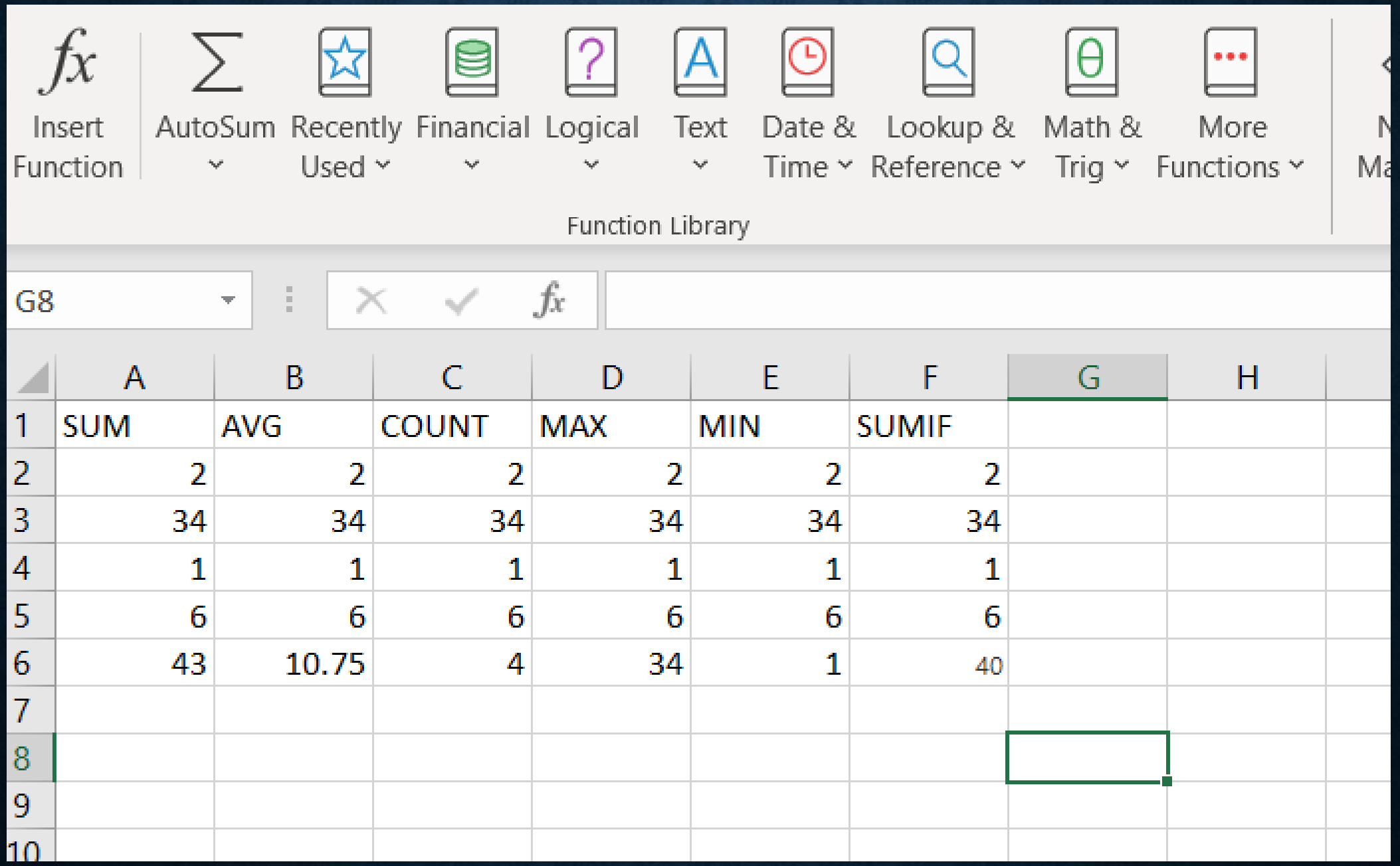

Restaurant
Name
Address
Capacity
Map
Get-Name()
Get-Map()
Get-Free-Tables(date, hour , table-kind, number-people)()
Take-Status(status)()
Reservation
Date
Hour
Name
Number-of-people
Get-status(date, hour)()
Make-Reservation(code-table, date, hour, table-kind, number-people, name-reservation)()
Table
Max-capacity
Type (smoking/noon smoking)
Location (x,y)
code-table
Change-Status()
Get-Status(date, hour, table-kind, number-people)()
Take-Status(status)()
Restaurants-Manager
Code
Get-List-Of-Restaurants()
Take-Name()
Get-Map(name-restaurant)()
Take-Map(map)()
Take-free-tables(name-restaurant, date, hour ,table-kind, number-people)()
List-of-free-tables(list-of-locations, list-of-tables)()
Reservations-Manager
Code
Get-List-of-Restaurants()
TakeList(list)()
Selected(name-restaurant, date,hour, table-kind, number-people)()
TakeMap(map)()
List-of-free-tables(list-of-locations,list-of-tables)()
Selected-Table(cod-table, name-reservation)()
OK()
AreYouAlive()

Facebook…

- Posts class
-- Image_Post
-- Text_Post
-- Video_Post

- Timeline class
-- User_Timeline
-- Public_Page_Timeline
-- Group_Timeline

- Reactions class
-- Like
-- Comment




| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | SUM | AVG | COUNT | MAX | MIN | SUMIF | | |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | | |
| 3 | 34 | 34 | 34 | 34 | 34 | 34 | | |
| 4 | 1 | 1 | 1 | 1 | 1 | 1 | | |
| 5 | 6 | 6 | 6 | 6 | 6 | 6 | | |
| 6 | 43 | 10.75 | 4 | 34 | 1 | 40 | | |
| 7 | | | | | | | | |
| 8 | | | | | | | | |
| 9 | | | | | | | | |
| 10 | | | | | | | | |