## Classes & Objects:

* Class : →
* A class is an entity which acts like a blue-print or template to create objects / instances.
* It is used to represent a real world entity.
* It has no memory of it's own and is assigned 1 byte by default.

## Object / Instance / Reference:

An object is an instance of a class. It means that it acts like a key, by which we can access all the data inside a class.

```
PNC
Class Employee {
    int Age;
    String Name;
    String Company;
}
Emp obj → [Sanjay, 21, TCS]
    → Speak();
    → Write();
```

Attributes / Properties

Methods

xxx

## OOP Properties:

* Reliability
* Efficiency
* Code Re-usability
* Data Encryption
* Data Security
* Clean Code

## Constructor: →

→ It is a special method used to initialize / instantiate objects.
→ It should be always public.
→ It should be same name as the class.
→ If we don't create, compiler generates a default constructor.
→ If we create our own, the default constructor is deleted.

## Two types of Constructor:

① Default Constructor
or
No-argument Constructor

② Parameterized Constructor

## What is a Package?

Industry Standard of creating a package.

⇒ com.company.project.module.file;

com → tcs.icici.login.login.java;

com
tcs
icici
login → login.java;

A collection of similar classes & interfaces.

* How do you resolve variable name clashes inside a class?
"this" keyword
or Pointer.

## Pillars of OOPs: →

① Encapsulation:

The process of wrapping the code & the data members inside a block → class, so that they are not accidentally modified, is called encapsulation. We achieve encapsulation by making the data members "private".

We can access them outside the class by two special functions: (2 Data Types)
① Getters → Fetch
② Setters → Assign
void

Afna College ****

Free Code Camp | GFG
Telusko
Code With Mosh
Programming With Mosh
Prep Insta → TCS Accenture