

Instituto de Formación Técnica Superior N° 18

Tecnicatura Superior en Desarrollo de Software

Desarrollo de Sistemas Orientado a Objetos

- Clase nro 1 -

Nombre y apellido del Profesor: Lic. Eduardo Iberti

1er Cuatrimestre 2025

Desarrollo de Sistemas Orientado a Objetos

- 1 - Presentación de la materia y modalidad de la cursada.
- 2 - Condiciones de aprobación.
- 3 - Aspectos de interés general.
- 4 - Objetivos de la materia.
- 5 - Fases del desarrollo de software.
- 6 - Paradigma de la Programación orientada a objetos (POO).
- 7 - Ventajas de la Programación orientada a objetos.
- 8 - Características principales de la POO.
- 9 - Clases y Objetos. Atributos y Métodos.

Presentación de la materia y modalidad de cursada

Desarrollo de Sistemas Orientado a Objetos es una materia cuatrimestral, que se dicta los días **Jueves de 18 a 22 hs** y los días **Viernes de 20.15 a 22 hs**.

La **cursada** de la materia **es presencial**. Se dará a través de encuentros **sincrónicos** conectados vía remota (Google Meet) y de instancias **presenciales** (en el IFTS 18).

Desde el **20 de Marzo al 04 de Julio de 2025**.

Cursada y Evaluación de la materia

Clases presenciales en el IFTS 18:

- En fechas a determinar. Se comunicará en la web del instituto (<https://www.ifts18.edu.ar/acad%C3%A9mico/calendario-presencialidad-2025>) y trataré de avisarles con tiempo.
- En principio, la **primera clase presencial** sería el **jueves 24 de Abril** (a confirmar).

Exámenes parciales y Trabajo Práctico Final:

- Tendrán dos instancias de evaluación, que serán un Exámen Parcial y un TP Final integrador, en fechas a determinar.

Condiciones de aprobación

Para **PROMOCIONAR** la materia (**APROBACIÓN FINAL**) el alumno deberá cumplir las siguientes condiciones:

- **Aprobar todos los Trabajos Prácticos** propuestos durante la cursada.
- **Aprobar el Exámen Parcial y el Trabajo Práctico Final** con una nota no inferior a 6 y obtener una **nota final de cursada (nota promedio) igual o superior a 7.**
- **Al finalizar la cursada, tener una Asistencia a clases mayor o igual al 75 %** (en forma remota y presencial).

Condiciones de aprobación

Para cursar la materia deben tener APROBADAS las siguientes **materias** que son **correlativas**:

- **Técnicas de Programación (1er Cuatrimestre)**
- **Administración de Bases de Datos (1er Cuatrimestre)**
- **Lógica Computacional (1er Cuatrimestre)**

Si algún alumno cursa la materia sin tener aprobadas estas dos materias, no podrá promocionar la materia, ni rendir el examen final hasta tanto apruebe las correlativas.

Es responsabilidad de cada alumno verificar la aprobación de las materias correlativas.

Condiciones de aprobación

El alumno/a que no haya rendido el examen parcial o haya obtenido una **nota entre 1 y 3**, tendrá la posibilidad de rendir un **recuperatorio** de ese examen.

Habrà **un solo exàmen recuperatorio**.

Tener en cuenta que el alumno que vaya a recuperatorio NO PODRÁ PROMOCIONAR LA MATERIA.

Condiciones de aprobación

El alumno/a que **apruebe TODOS los Trabajos Prácticos y el Exámen Parcial** con una **nota final de cursada entre 4 y 6** (recuperatorio incluido), y que tenga una **asistencia a clases mayor o igual al 75 %** (en forma remota y presencial), obtendrá la condición de **REGULAR**.

Lo cual lo habilitará a **rendir un examen final** para la aprobación final de la materia.

Quien **NO apruebe** alguno de los Trabajos Prácticos, o haya obtenido una **nota final de cursada entre 1 y 3** (recuperatorio incluido), o que tenga una **asistencia a clases menor al 75 %** (en forma remota y presencial), quedará **DESAPROBADO** y deberá recursar la materia.

Condiciones de aprobación

El alumno que quede en condición de **REGULAR** habiendo finalizado la cursada, **deberá rendir un examen final**. El cual será en modalidad oral, para lo cual deberán preparar una exposición de algunos de los temas del programa de la materia.

Recuerden que para poder rendir en las mesas de exámenes finales **deben inscribirse a través de la plataforma SIU Guaraní**.

Mesas de exámenes finales en Julio de 2025:

- Primer llamado. Del 14 al 18 de Julio de 2025.

Aspectos de interés general

- **Página web del Instituto:**
<https://www.ifts18.edu.ar/>
- **Sistema de gestión académica SIU Guaraní:**
<https://guarani-autogestionagencia.bue.edu.ar/>
- **Guía de uso para el estudiante**
<https://drive.google.com/file/d/16xwns6jgDL7knI7CbGsUqFTSJADS5H49/view>

Objetivos de la materia

Que los estudiantes sean capaces de:

- Analizar e interpretar **conceptos básicos de la programación orientada a objetos** tales como: clases, objetos, atributos, operaciones, abstracción, encapsulamiento, herencia y polimorfismo.
- Implementar modelos que dan solución a situaciones del mundo real basados en el paradigma de la programación orientada a objetos, empleando el **lenguaje de programación Python**.

Fases del desarrollo de software

1. Análisis.
2. Diseño.
3. Implementación.

Fases del desarrollo de software

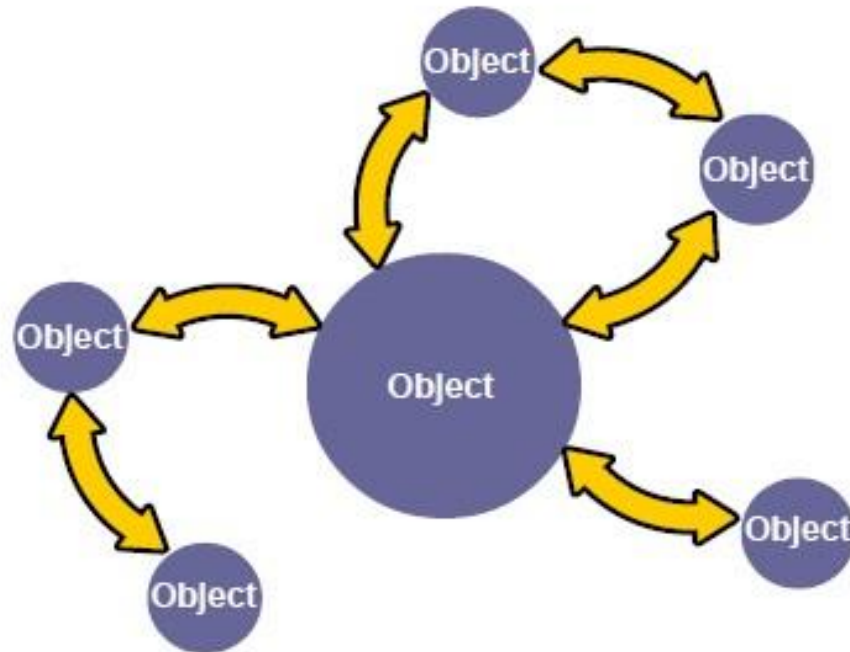
El **análisis** es la fase cuyo objetivo es estudiar y comprender el dominio del problema, es decir, el análisis se centra en responder al interrogante **¿QUÉ HACER?**

El **diseño**, por su parte, dirige sus esfuerzos en desarrollar la solución a los requisitos planteados en el análisis, esto es, el diseño se haya centrado en el espacio de la solución, intentando dar respuesta a la cuestión **¿CÓMO HACERLO?**

Por último, la fase de **implementación** sería la encargada de la traducción del diseño de la aplicación al lenguaje de programación elegido, adaptando por tanto la solución a un entorno concreto.

Paradigma de la Programación orientada a objetos (POO)

La programación orientada a objetos es un paradigma de programación que busca representar entidades u objetos agrupando datos y métodos que puedan describir sus características y comportamientos.



Ventajas de la Programación Orientada a Objetos

- ✓ Fomenta la reutilización y extensión del código.
- ✓ Permite crear sistemas más complejos.
- ✓ Relacionar el sistema al mundo real.
- ✓ Facilita la creación de programas visuales.
- ✓ Facilita la construcción de prototipos.
- ✓ Agiliza el desarrollo de software.
- ✓ Facilita el trabajo en equipo.
- ✓ Facilita el mantenimiento del software.

Características de la POO

✓ **Abstracción.**

Modelización a partir de una simplificación de la realidad. Expresa las características esenciales de un objeto (atributos y comportamientos), las cuales permitirán distinguir a un objeto respecto de los demás.

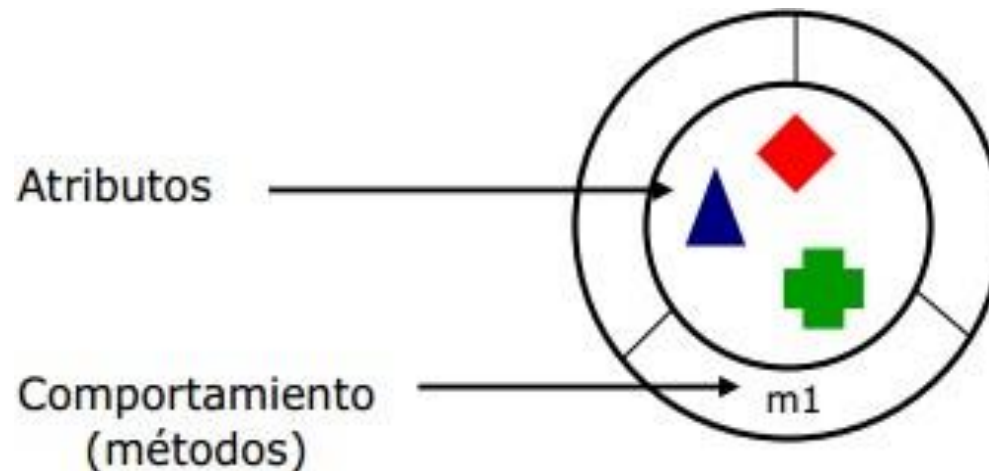
✓ **Modularidad.**

Fragmentación del problema a resolver, mediante la división del proyecto de software en un conjunto de módulos (en forma de paquetes y clases).

Características de la POO

✓ Encapsulamiento.

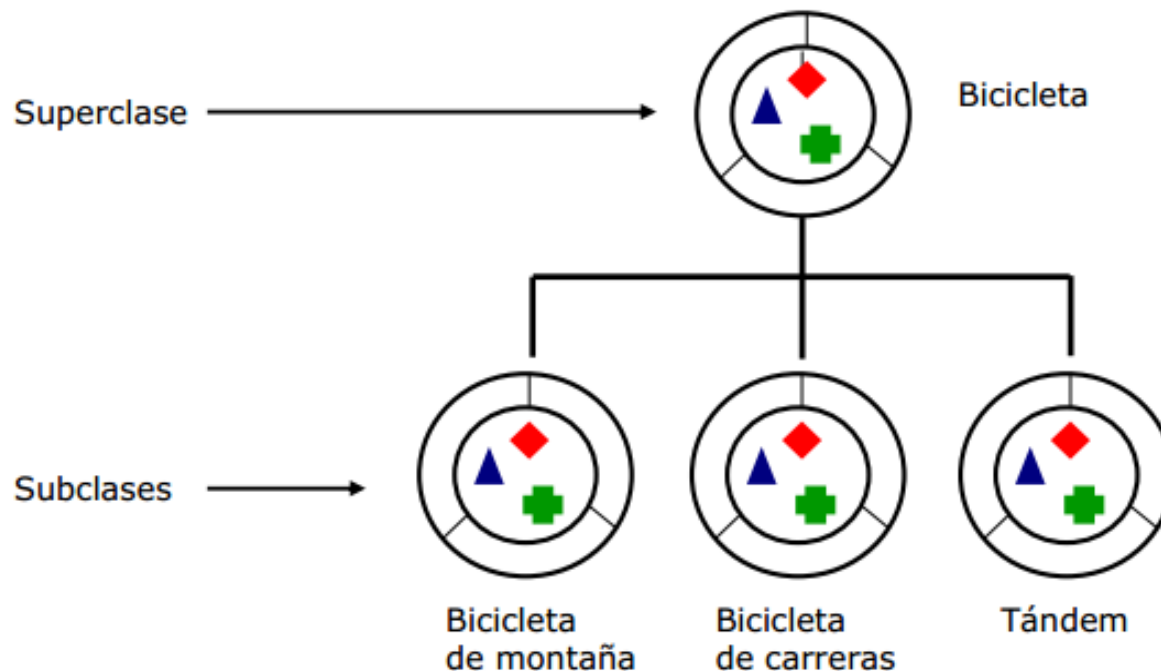
Este principio permite definir qué características (atributos y comportamientos) de un objeto serán públicas y cuáles estarán ocultas al resto de los objetos.



Características de la POO

✓ Herencia.

Es la capacidad de una clase (Subclase) para definirse en términos de otra clase y “heredar” atributos y comportamientos de la clase de orden superior (Superclase).



Características de la POO

✓ Polimorfismo.

El polimorfismo permite que distintos objetos pertenecientes a una misma clase “respondan” de diferentes formas a un mismo mensaje, esto se obtiene modificando en cada subclase el comportamiento de un método heredado.

Clases y Objetos. Atributos y Métodos

✓ Los Objetos y las Clases

En lugar de crear estructuras de datos por un lado y código por el otro, la idea es asociar piezas de código con los datos.

Debemos entender los objetos como abstracciones de la realidad con ciertas características representativas. Estos objetos representan “ideas” del mundo real.

Un Objeto es un ente que contiene un conjunto de atributos (estado) y operaciones (comportamiento).

Las Clases son plantillas que contienen atributos y operaciones, se consideran moldes para los objetos. Es decir que las Clases definen la estructura de sus objetos.

Clases y Objetos. Atributos y Métodos

✓ Los Atributos

Determinan el estado que tiene un objeto. Estos pueden ser tipos de datos primitivos u objetos de cualquier clase.

```
public class Empleado {  
    public int nro_legajo;  
    public String apellido;  
    public String nombre;  
    public Date fecha_nacimiento;  
    private double sueldoBasico;  
}
```

Clases y Objetos. Atributos y Métodos

✓ Los Métodos

Son las acciones u operaciones que podrán realizar los objetos. Determinan el comportamiento y la responsabilidad que tendrán las clases.

```
public void asignarNroLegajo(int nro){
    nro_legajo = nro;
}
public int obtenerNroLegajo(){
    return nro_legajo;
}
public double obtenerSueldoBasico(){
    return sueldoBasico;
}
```

Clases y Objetos. Atributos y Métodos

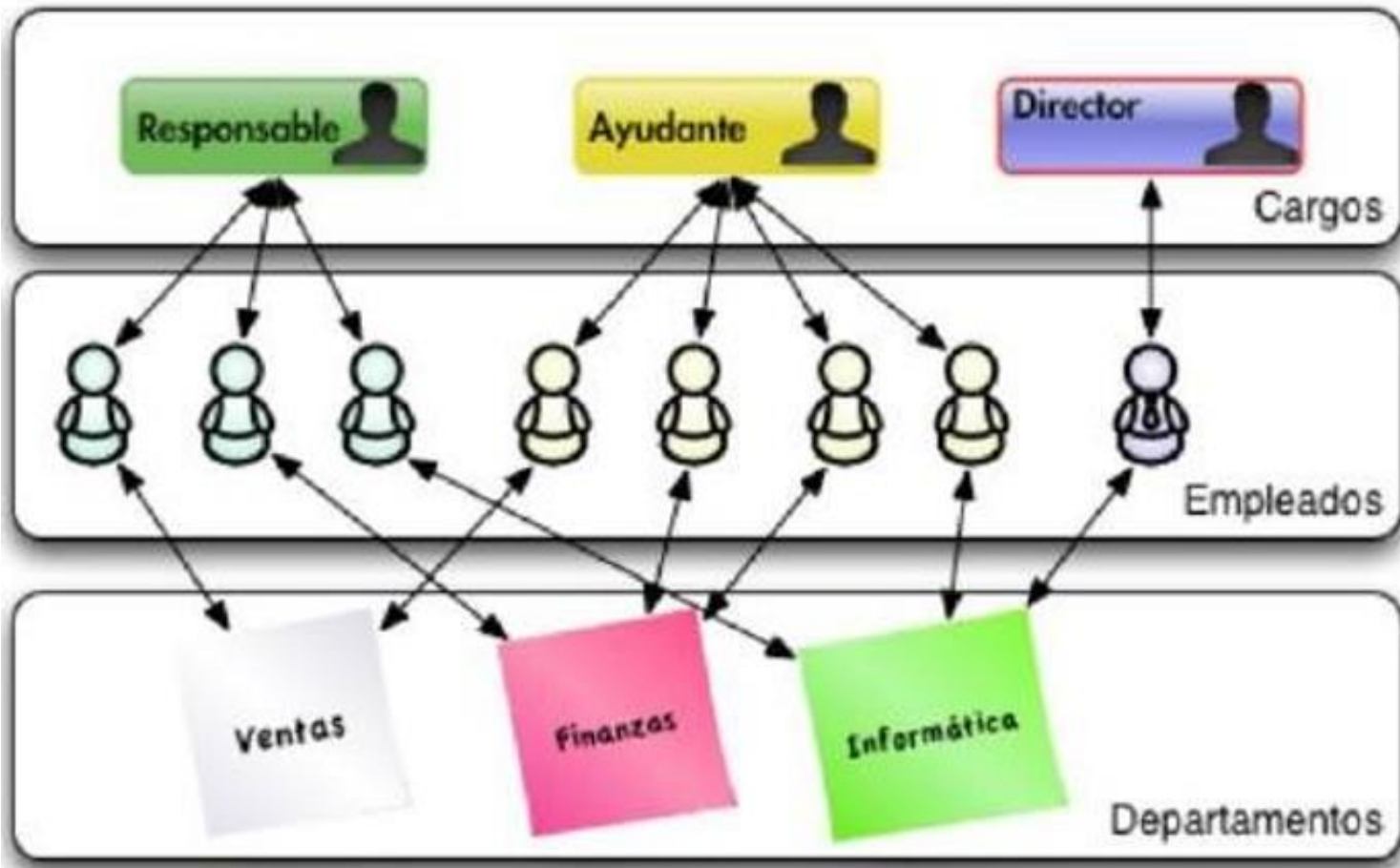
✓ Ejemplo de objetos en una empresa

Se tienen tres **clases de objetos**:

- Cargo
- Empleado
- Departamento

De la clase Cargo hay tres instancias (Director, Ayudante y Responsable). De Empleado, ocho instancias, y de Departamento, tres instancias (Ventas, Finanzas e Informática)

Clases y Objetos. Atributos y Métodos



Clases y Objetos. Atributos y Métodos

Las **responsabilidades de los objetos de una clase**, las podemos dividir en tres tipos:

- La información que conocen (**propiedades, datos o atributos**).
- Las acciones que llevan a cabo (**operaciones, métodos o funciones**).
- Los objetos que conocen (**las asociaciones**).

Ejemplo: un Empleado sabe cuál es su información (nombre, teléfono, etc.), existe una serie de tareas u operaciones que puede llevar a cabo (redactar una solicitud, aprobarla, etc.) y está asociado con otras instancias (su cargo y su departamento).

Clase 1

Consultas, dudas sobre
la clase?

