

PROGRAMA DE: PROGRAMACION ORIENTADA A OBJETOS F.P.

CÓDIGO: 469

HORAS DE CLASE				DOCENTE RESPONSABLE
TEÓRICAS		PRÁCTICAS		Dr. Carlos Enrique BERGER
p/semana	p/cuatrim.	p/semana	p/cuatrim.	
4hs	64hs	4hs	64hs	

**DESCRIPCIÓN:**

La materia aborda contenidos correspondientes al paradigma de la Programación Orientada a Objetos, con un enfoque teórico/práctico que brinda al estudiantado las herramientas necesarias para la comprensión de conceptos generales y las técnicas recomendadas para su aplicación en la resolución de problemas concretos. La terminología, definiciones y conceptos teóricos se presentan de forma genérica para su interpretación de forma independiente al lenguaje de implementación, mientras que las actividades prácticas se desarrollan bajo el lenguaje Python, incorporando herramientas de programación específicas del mismo e intercalando ocasionalmente ejemplos de implementación en otros lenguajes.

Al finalizar la materia, el alumno contará con los conocimientos apropiados para incorporar la programación orientada a objetos en aplicaciones web, de escritorio y móviles, entendiendo sus ventajas y desventajas y siguiendo las pautas recomendadas como buenas prácticas de programación.

**PROGRAMA SINTÉTICO:****UNIDAD TEMÁTICA I:** Conceptos Básicos**UNIDAD TEMÁTICA II:** Objetos y clases**UNIDAD TEMÁTICA III:** Métodos en Python**UNIDAD TEMÁTICA VI:** Herencia**UNIDAD TEMÁTICA V:** Manejo de Errores**UNIDAD TEMÁTICA VI:** Patrones de diseñoVigencia a partir  
del año:

2024

PROGRAMA DE: PROGRAMACION ORIENTADA A OBJETOS F.P.

CÓDIGO: 469

**PROGRAMA ANALÍTICO:****UNIDAD TEMÁTICA I: Conceptos Básicos**

Reseña histórica y motivos para el desarrollo de POO. Evolución del concepto de tipo de datos abstracto. Encapsulamiento y Abstracción. Ocultamiento de la información. Encapsular atributos y comportamiento. Herramientas y Entorno de desarrollo: instalación y configuración.

**UNIDAD TEMÁTICA II: Objetos y clases**

Clases e instancias. Diagrama UML. Creación de Objetos: métodos, constructores, clases y prototipos. Ciclo de vida de los objetos. Calificadores de acceso. Protección de estado. Relaciones: Dependencias y Asociaciones, Composición y Agregación.

**UNIDAD TEMÁTICA III: Métodos en Python**

Tipos de métodos en Python: de clase, de instancia, estáticos y abstractos. Decoradores. Métodos dunder o mágicos. Diferencias, ventajas y desventajas. Aplicaciones. Métodos sobrecargados.

**UNIDAD TEMÁTICA IV: Herencia**

Definición y conceptos. Técnicas de herencia. Generalización-especialización. Jerarquía de clases. Extensión de clases. Herencia simple y múltiple. Clases abstractas y concretas. Herencia y Composición. Polimorfismo. Sobrecarga y sobreescritura.

**UNIDAD TEMÁTICA V: Manejo de Errores**

Gestión de errores. Ventajas del uso de excepciones. Detección de errores y excepciones. Lanzamiento de una excepción. Sentencias de captura y delegación de excepciones: sintaxis del bloque try-except-finally. Excepciones del lenguaje y excepciones personalizadas. Sentencia assert.

**UNIDAD TEMÁTICA VI: Patrones de diseño**

Concepto de Patrones. Importancia de su uso y conocimiento. Ejemplos de implementación de patrones de uso común.

PROGRAMA DE: PROGRAMACION ORIENTADA A OBJETOS F.P.

CÓDIGO: 469

**BIBLIOGRAFÍA**

- Booch, G. (2006). *Object oriented analysis & design with application*. Pearson Education India.
- Deitel, Paul J., Deitel, Harvey M. (2016). *C# 6 for Programmers*. (6th ed.). Pearson.
- Eckel, Bruce (2017). *Thinking in Java*. (4th ed.). Prentice Hall.
- Flanagan, David (2011). *JavaScript: The Definitive Guide*. (6th ed.). O'Reilly Media.
- Hasin Hayder. (2007). *Object-oriented Programming with PHP5: Learn to Leverage PHP5's OOP Features to Write Manageable Applications with Ease*. Packt Publishing.
- Joyanes Aguilar, L. (2008). *Fundamentos de programación: algoritmos, estructura de datos y objetos* (4.ª ed.). España: McGraw-Hill/Interamericana.
- Lutz, Mark (2014). *Learning Python*. (5th ed.). O'Reilly Media.
- Morero, Francisco (1999-2000). *Introducción a la OOP*. Versión 1.0.0. Grupo EIDOS.
- Phillips, Dusty (2018). *Python Object-Oriented Programming*. (3rd ed.). Packt Publishing.
- Ramalho, L. (2015). *Fluent Python: Clear, Concise, and Effective Programming*. (1st ed.). O'Reilly Media.
- Stroustrup, Bjarne (2013). *Programming: Principles and Practice Using C++*. (2nd ed.). Addison-Wesley Professional.
- Taylor, R. N., Medvidović N. and Dashofy, E. M. (2009). *Software architecture: Foundations, Theory and Practice*. John Willey & Sons.

El presente Programa se ha elaborado bajo responsabilidad del/de la, las/los docente/s cuyas firmas se exponen a continuación. Las autoridades de cada Facultad, y del Vicerrectorado del Área Académica o Dirección de Coordinación Educativa de esta Universidad, suscriben prestando conformidad.

Vigencia a partir del año:

2024



GOBIERNO DE LA PROVINCIA DE BUENOS AIRES  
2024 - Año del 75° Aniversario de la gratuidad universitaria en la República Argentina

**Hoja Adicional de Firmas**  
**Anexo de Firma Conjunta**

**Número:**

**Referencia:** Programa Creación Programación Orientada a Objetos F.P.

---

El documento fue importado por el sistema GEDO con un total de 3 pagina/s.

