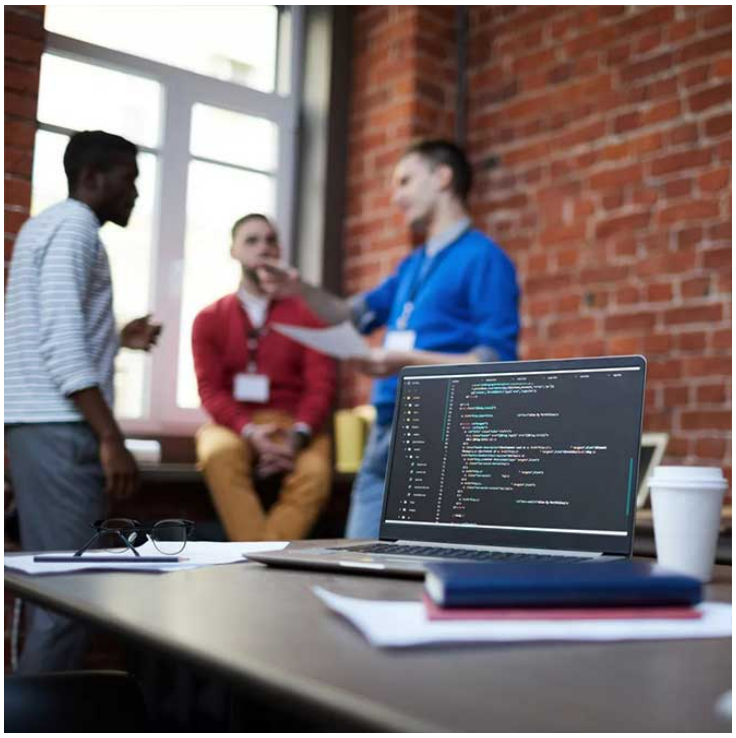
# Programador TIC en Orientación a Objetos (200 horas)



**Categoría**: Informática, Programación y Comunicaciones
**Página del curso**: https://www.ensenanzadual.com/cursos/programador-tic-en-orientacion-a-objetos-200-horas/

## Objetivo

Con el Curso de **200 horas** de **Programador TIC en Orientación a Objetos** adquirirás los conocimientos necesarios para cambiar de carrera o actualizar tus competencias profesionales, no lo dudes más y solicita información sobre este curso.

## Descripción

**UNIDAD FORMATIVA 1. PRINCIPIOS DE LA PROGRAMACIÓN ORIENTADA A OBJETOS UNIDAD DIDÁCTICA 1. INTRODUCCIÓN AL PARADIGMA ORIENTADO A OBJETOS.** - Ciclo de desarrollo del software bajo el paradigma de orientación a objetos: Análisis, diseño y programación orientada a objetos. - Análisis del proceso de construcción de software: Modularidad. - Distinción del concepto de módulo en el paradigma orientado a objetos.

Identificación de objetos como abstracciones de las entidades del mundo real que se quiere modelar.

- - Descripción de objetos: Conjunto de datos que definen un objeto y conjunto comportamientos que pueden solicitarse a los objetos. - - Identificación del comportamiento de un objeto: Concepto de mensaje. **UNIDAD DIDÁCTICA 2. CLASES Y OBJETOS.**

Distinguir el concepto de clase y sus atributos, métodos y mecanismo de encapsulación:

- - Relación entre interfaz y clase. - - Distinción de los tipos de datos y clases.

Análisis de los objetos: Estado, comportamiento e identidad:

- - Análisis de mensajes. - - Tipos de métodos y su clasificación: Métodos de acceso, de selección o consulta, de construcción, de destrucción. - Uso de objetos como instancias de clase. Instancia actual (this, self, current). - Identificación del concepto de programa en el paradigma orientado a objetos. POO = Objetos + Mensajes. **UNIDAD DIDÁCTICA 3. GENERALIZACIÓN/ESPECIALIZACIÓN: HERENCIA.**

Descripción del concepto de herencia: Simple y múltiple:

- - Relación de herencia: Características. - - Reglas y características que definen una relación de herencia: Regla «Es-un». - - Transmisión de atributos y métodos. - - Regla de especialización de la superclase en la subclase. - - Acceso a los atributos de una clase y acoplamiento entre las clases. - - Utilización de objetos this (current, self u otros) y super. - - Leyes de Demeter.

Distinción de la herencia múltiple:

- - Problemas: Conflictos de nombres, herencia repetida. - - Soluciones. - Creación de objetos en la herencia.

Clasificación jerárquica de las clases:

- - Clase raíz. - - Clases abstractas. - - Métodos virtuales. - - Redefinición de métodos. **UNIDAD DIDÁCTICA 4. RELACIONES ENTRE CLASES.** - Distinción entre Agregación/Composición. - Distinción entre Generalización / Especialización. - Identificación de asociaciones. **UNIDAD DIDÁCTICA 5. ANÁLISIS DEL POLIMORFISMO.** - Concepto.

Tipos:

- - Polimorfismo en tiempo de compilación (sobrecarga). - - Polimorfismo en tiempo de ejecución (ligadura dinámica). - - Objetos polimórficos. - - Comprobación estática y dinámica de tipos. **UNIDAD DIDÁCTICA 6. TÉCNICAS DE PROGRAMACIÓN ESTRUCTURADA.** - Identificación de elementos básicos: constantes, variables, operadores y expresiones. - Análisis de estructuras de control: Secuencial, condicional y de repetición.

Distinción entre funciones y procedimientos:

- - Interfaz. - - Paso de parámetros: Por valor y por referencia. - - Parámetros actuales y formales. - - Funciones: valor de retorno. - - Procedimientos. - - Ámbito de las variables. - - Almacenamiento de las variables. - Demostración de llamadas a funciones y procedimientos.

Empleo de llamadas a funciones y procedimientos incluidos en las clases:

- - Llamadas calificadas y no calificadas (instancia actual). - - Paso de parámetros. - - Los atributos de la clase. **UNIDAD DIDÁCTICA 7. ESTRUCTURA DE LA INFORMACIÓN.** - Enumeración de datos simples: Numéricos (enteros y reales), lógicos, carácter, cadena de caracteres, puntero o referencia a memoria.

Datos estructurados: Arrays:

- - Listas enlazadas, pilas y colas. - - Estructuras. - - Ficheros. - - Otras estructuras complejas: Tablas hash e Introducción a los árboles y grafos.

Mecanismos de gestión de memoria:

- - Uso de la gestión automática de memoria. - - Construcción y destrucción de objetos. - - Objetos inalcanzables. - - Recolección de «basura». - - Métodos constructores y destructores. **UNIDAD DIDÁCTICA 8. LENGUAJES DE PROGRAMACIÓN ORIENTADOS A OBJETOS.**

Análisis del lenguaje de programación orientado a objetos y paradigma orientado a objetos:

- - Lenguajes de programación orientados a objetos. - - Lenguajes de programación basados en objetos. - - Lenguajes de programación que utilizan objetos. - Comparación entre los lenguajes de programación orientados a objetos más habituales. Características esenciales.

Librerías de clases:

- - Definición de su estructura. - - Creación y utilización. **UNIDAD DIDÁCTICA 9. IMPLEMENTACIÓN DEL PARADIGMA UTILIZANDO UN LENGUAJE DE PROGRAMACIÓN ORIENTADO A OBJETOS.** - Elección del lenguaje. - Enumeración de los tipos de aplicaciones. - Herramientas de desarrollo. - Tipos de datos y elementos básicos característicos del lenguaje. Instrucciones. - Estudio y utilización de las clases básicas incluidas en la librería de clases.

Definición de clases:

- - Construcción de métodos. Sobrecarga. - - Construcción de atributos. - - Construcción de la interfaz de la clase. - - Construcción de clases incluyendo relaciones de Agregación /Composición y Asociación. - - Construcción de clases con herencia. - - Construcción de clases con herencia múltiple. - - Definición de clases abstractas. - - Construcción de clases con herencia incluyendo polimorfismo. - - Empleo de excepciones.

Gestión de eventos:

- - Eventos, fuentes y auditores de eventos. - - Tipos de eventos. Mecanismos de gestión de eventos. - - Librerías de clases asociadas.

Empleo de hilos:

- - Fundamentos. - - Creación. - - Prioridad. - - Comunicación. - - Sincronización.

- - Estados. - - Creación y ejecución de hilos en el lenguaje. - - Librerías de clases asociadas. - - Programación multihilo.

Definición y análisis de programación en red:

- - Aplicaciones cliente servidor. - - Sockets. - Acceso a bases de datos desde las aplicaciones. Librerías de clases asociadas. **UNIDAD FORMATIVA 2. MODELO DE PROGRAMACIÓN WEB Y BASES DE DATOS UNIDAD DIDÁCTICA 1. INTRODUCCIÓN AL DESARROLLO DE APLICACIONES EN EL MODELO DE PROGRAMACIÓN WEB.** - Análisis de la arquitectura web: Cliente ligero, servidor web, servidor de aplicaciones, servidor de datos. - Enumeración de protocolos y tecnologías habituales. - Análisis de los modelos de programación estándares de facto. - Uso de componentes orientados a objeto como base en el desarrollo de aplicaciones en el modelo de programación web. **UNIDAD DIDÁCTICA 2. ARQUITECTURA MULTICAPA (N-TIER).** - Análisis de la arquitectura multicapa. - Distinción y estudio del modelo de tres capas en web: presentación, aplicación y datos. - Diseño de arquitecturas de aplicación basadas en el modelo multicapa. - Análisis del concepto de lógica de negocio y significado de la capa lógica. **UNIDAD DIDÁCTICA 3. LA CAPA DE PRESENTACIÓN.** - Descripción de la capa de presentación: El lenguaje de hipertexto. - Descripción de la capa de presentación avanzada: Lenguajes de scripting y lenguaje de hipertexto dinámico. - Análisis de lenguajes orientados a la preparación de la capa de presentación y a la ejecución de solicitudes desde clientes ligeros web. (JSP, Servlets, ASP, PHP). **UNIDAD DIDÁCTICA 4. DISEÑO DE BASES DE DATOS RELACIONALES.** - Definición de bases de datos relacionales. - Diseño de bases de datos en varios niveles. - Análisis de los distintos tipos de relaciones y su implementación en base de datos. - Descripción del lenguaje de acceso a base de datos. - Descripción de correlaciones entre el modelo relacional y modelo orientado a objetos. - Nociones sobre el almacenamiento de objetos en las bases de datos relacionales. **UNIDAD DIDÁCTICA 5. ACCESO A BASES DE DATOS RELACIONALES: CAPA DE ACCESO A DATOS.** - Análisis del API de acceso a la base de datos. - Nivel controlador. - Interfaz de acceso a la base de

datos (driver). - Análisis del nivel aplicación. - Establecimiento de la conexión con una base de datos. - Operar sobre la base de datos. Sentencias del lenguaje de acceso a base de datos. Objetos que permiten ejecutar una consulta. Objetos que permiten manipular el resultado de una consulta. - Integración de los tipos de datos propios del lenguaje de acceso a base de datos en el lenguaje de programación de la aplicación. - Procedimientos almacenados. - Transacciones distribuidas. **UNIDAD DIDÁCTICA 6. LENGUAJES DE DEFINICIÓN DE DATOS.** - Conceptos básicos, nociones y estándares. - Lenguaje de definición de datos (DDL SQL) y aplicación en SGBD actuales. - Discriminación de los elementos existentes en el estándar SQL-92 de otros elementos existentes en bases de datos comerciales.

Sentencias de creación: CREATE:

- - Bases de datos. - - Tablas. - - Vistas. - - Disparadores o Triggers. - - Procedimientos.

Sentencias de modificación: ALTER:

- - Bases de datos. - - Tablas. - - Vistas. - - Disparadores o Triggers. - - Procedimientos.

Sentencias de borrado: DROP, TRUNCATE:

- - Bases de datos. - - Tablas. - - Vistas. - - Disparadores o Triggers. - - Procedimientos. **UNIDAD DIDÁCTICA 7. MANIPULACIÓN DE LOS DATOS.** - Lenguaje de manipulación de datos (DML SQL). - Consultas de datos: SELECT. - Inserción de datos: INSERT. - Modificación de datos: UPDATE. - Eliminación de datos: DELETE. - Agregación de conjuntos de datos para consulta: JOIN, UNION.

## Información adicional

- Online: Si
- Tipo: Profesiones
- Horas: 200

- Unidades: 16