

adaptium

Programación orientada a objetos



Área: Programación con lenguajes orientados a objetos y bases de datos relacionales

Modalidad: Teleformación

Duración: 250 h

Precio: Consultar

[Curso Bonificable](#)

[Contactar](#)

[Recomendar](#)

[Matricularme](#)

OBJETIVOS

Desarrollar componentes software en lenguajes de programación orientados a objetos.

CONTENIDOS

PRINCIPIOS DE LA PROGRAMACIÓN ORIENTADA A OBJETOS - 1 introducción al paradigma orientado a objetos - 1.1 ciclo de desarrollo del software - 1.2 análisis del proceso de construcción de software - modularidad - 1.3 identificación de objetos como abstracciones de las entidades - 2 clases y objetos - 2.1 distinguir el concepto de clase y sus atributos, métodos y mecanismo de encapsulación - 2.2 análisis de los objetos - estado, comportamiento e identidad - 2.3 usos de objetos como instancias de clase. instancia actual - 2.4 identificación del concepto de programa en el paradigma orientado a objetos - 3 generalización-especialización - 3.1 descripción del concepto de herencia - simple y múltiple - 3.2 distinción de la herencia múltiple - 3.3 creación de objetos en la herencia - 3.4 clasificación jerárquica de las clases - 4 relaciones entre clases - 4.1 distinción entre agregación - composición - 4.2 distinción entre generalización - especialización - 4.3 identificación de asociaciones - 5 análisis del polimorfismo - 5.1 concepto - 6 técnicas de programación estructurada - 6.1 identificación de elementos básicos - 6.2 análisis de estructuras de control - 6.3 distinción entre funciones y procedimientos - 6.4 demostración de llamadas a funciones y procedimientos - 6.5 empleo de llamadas a funciones y procedimientos incluidos en las clases - 7 estructura de la información - 7.1 enumeración de datos simples - 7.2 datos estructurados - 7.3 listas enlazadas, pilas y colas - 7.4 otras estructuras complejas - 7.5 mecanismos de gestión de memoria - 8 lenguajes de programación orientados a objetos - 8.1 análisis del lenguaje de programación orientado a objetos y paradigma orientado a objetos - 8.2 comparación entre los lenguajes de programación orientados a objetos más habituales - 8.3 librerías de clases - 9 implementación del paradigma - lenguaje de programación - 9.1 elección del lenguaje - 9.2 enumeración de los tipos de aplicaciones - 9.3 herramientas de desarrollo - 9.4 tipos de datos y elementos básicos característicos del lenguaje - 9.5 estudio y utilización de las clases básicas incluidas en la librería de clases - 9.6 definición de clases - 9.7 construcción de la interfaz de clase - 9.8 construcción de clases con herencia - 9.9 empleo de excepciones - 9.10 gestión de eventos - 9.11 empleo de hilos - 9.12 definición y análisis de programación en red - 9.13 acceso a bases de datos desde aplicaciones - 9.14 cuestionario: cuestionario de evaluación - MODELO DE PROGRAMACIÓN WEB Y BASES DE DATOS - 1 introducción al desarrollo de aplicaciones en el modelo de programación web - 1.1 análisis de la arquitectura web - 1.2 enumeración de protocolos y tecnologías habituales - 1.3 análisis de los modelos de programación estándares de facto - 1.4 uso de componentes orientados a objeto como base en el desarrollo de aplicaciones - 2 arquitectura multicapa (n-tier) - 2.1 análisis de la arquitectura multicapa - 3 la capa de presentación - 3.1 descripción de la capa de presentación - el lenguaje de hipertexto - 3.2 descripción de la capa de presentación avanzada - 3.3 análisis de lenguajes orientados a la preparación de la capa de presentación - 4 diseño de bases de datos relacionales - 4.1 definición de bases de datos relacionales - 4.2 diseño de bases de datos en varios niveles - 4.3 análisis de los distintos tipos de relaciones y su implementación - 4.4 descripción del lenguaje de acceso a bases de datos - 4.5 descripción de correlaciones entre el modelo relacional y modelo orientado a objetos - 5 acceso a bases de datos relacionales - 5.1 análisis del api de acceso a la base de datos - 5.2 análisis del nivel aplicación - 5.3 integración de los tipos de datos propios del lenguaje de acceso a base - 5.4 procedimientos almacenados - 5.5 transacciones distribuidas - 6 lenguajes de definición de datos - 6.1 conceptos básicos, nociones y estándares - 6.2 lenguaje de definición de datos (ddl sql) y aplicación en sgbd actuales - 6.3 discriminación de los elementos existentes en el estándar sql-92 - 7 manipulación de los datos - 7.1 lenguaje de manipulación de datos (dml sql) - 7.2 consultas de datos - 7.3 cuestionario: cuestionario de evaluación - EL CICLO DE VIDA DEL DESARROLLO DE APLICACIONES - 1 proceso de ingeniería del software - 1.1 distinción de las fases del proceso de ingeniería software - 1.2 análisis de los modelos del proceso de ingeniería - 1.3 análisis de metodologías de desarrollo orientado a objeto - 1.4 resolución de un caso práctico de metodologías de desarrollo que utilizan

adaptium

uml - 1.5 definición del concepto de herramienta case - 2 planificación y seguimiento - 2.1 realización de estimaciones - 2.2 planificaciones - modelos de diagramado. diagrama de gantt - 2.3 análisis del proceso del seguimiento - reuniones e informes - 3 diagramado - 3.1 identificación de los principios básicos de uml - 3.2 ejemplo de diagramas de uso - 4 desarrollo de la gui - 4.1 análisis del modelo de componentes y eventos - 4.2 identificación de elementos de la gui - 4.3 presentación del diseño orientado al usuario - 4.4 empleo de herramientas de interfaz gráfica - 5 calidad en el desarrollo del software - 5.1 enumeración de criterios de calidad - 6 pruebas - 6.1 identificación de tipos de pruebas - 6.2 análisis de pruebas de defectos - 7 excepciones - 7.1 definición - 8 documentación - 8.1 generación automática de documentación - 8.2 estructura del documento. como producir un documento - 8.3 cuestionario: cuestionario de evaluación - 8.4 cuestionario: cuestionario de evaluación -

METODOLOGIA

- **Total libertad de horarios** para realizar el curso desde cualquier ordenador con conexión a Internet, **sin importar el sitio desde el que lo haga**. Puede comenzar la sesión en el momento del día que le sea más conveniente y dedicar el tiempo de estudio que estime más oportuno.
- En todo momento contará con un **asesoramiento de un tutor personalizado** que le guiará en su proceso de aprendizaje, ayudándole a conseguir los objetivos establecidos.
- **Hacer para aprender**, el alumno no debe ser pasivo respecto al material suministrado sino que debe participar, elaborando soluciones para los ejercicios propuestos e interactuando, de forma controlada, con el resto de usuarios.
- **El aprendizaje se realiza de una manera amena y distendida**. Para ello el tutor se comunica con su alumno y lo motiva a participar activamente en su proceso formativo. Le facilita resúmenes teóricos de los contenidos y, va controlando su progreso a través de diversos ejercicios como por ejemplo: test de autoevaluación, casos prácticos, búsqueda de información en Internet o participación en debates junto al resto de compañeros.
- **Los contenidos del curso se actualizan para que siempre respondan a las necesidades reales del mercado**. El departamento multimedia incorpora gráficos, imágenes, videos, sonidos y elementos interactivos que complementan el aprendizaje del alumno ayudándole a finalizar el curso con éxito.
- **El curso puede realizarse exclusivamente a través de Internet pero si el estudiante lo desea puede adquirir un manual impreso**. De esta manera podrá volver a consultar la información del curso siempre que lo desee sin que necesite utilizar el Campus Virtual o, estar conectado a Internet.

REQUISITOS

Los requisitos técnicos mínimos son:

- Navegador Microsoft Internet Explorer 5.5 o superior, con plugin de Flash, cookies y JavaScript habilitados. No se garantiza su óptimo funcionamiento en otros navegadores como Firefox, Netscape, Mozilla, etc.
- Resolución de pantalla de 800x600 y 16 bits de color o superior.
- Procesador Pentium II a 300 Mhz o superior.
- 32 Mbytes de RAM o superior.