Taller Programación Orientada a Objetos.

Temario y material del taller de tutorías de Programación orientada a objetos de la carrera TICS.

Requisitos del estudiante: Tener cursado el 3er semestre de la carrera TICS. Computadora propia con NETBEANS instalado.

Requisitos del propedéutico: aula, proyector, grupo máximo de 25 alumnos. 4-5 horas en una semana (lunes a viernes) 5 días.

**Día 1**

**Tema 1 Introducción al paradigma de la programación orientado a objetos.**

* Elementos del modelo de objetos: clases, objetos, abstracción, modularidad, encapsulamiento, herencia y polimorfismo.
* Lenguaje de modelado unificado: diagrama de flujos.

El alumno aprenderá en el primer día en realizar diagramas de flujos de un programa lineal, abstrayendo los pasos para realizar dicho programa realizado en clase. El alumno aprenderá que es una clase en java, un objeto, abstracción, encapsulamiento, herencia, polimorfismo y modularidad. (Identificar variables y condiciones)

**Día 2**

**Tema 2 Clases y objetos.**

* Declaración de clases: atributos, métodos, encapsulamiento.
* Instancia de una clase.
* Referencia al objeto actual.
* Métodos: declaración, mensajes, paso de parámetros, retorno de valores.
* Constructores y destructores: declaración, uso y aplicaciones.
* Sobrecarga de métodos.
* Sobrecarga de operadores: Concepto y utilidad, operadores unarios y binarios.

El alumno pondrá en práctica y creara un primer programa utilizando clases, métodos y encapsulamiento para el desarrollo de un programa sencillo en netbeans y podrá distinguir el uso de cada uno.

**Día 3**

**Tema 3 Herencia.**

* Definición herencia: clase base, clase derivada.
* Clasificación herencia. Herencia simple, herencia múltiple.
* Reutilización de miembros heredados.
* Referencia al objeto de la clase base.
* Constructores y destructores en clases derivadas.
* Redefinición de métodos en clases derivadas.

El alumno aprenderá a dar herencia de una clase a otra, dando sus atributos a la clase principal, a reutilizar objetos y métodos en dichos constructores y realizar un programa, que utilice estos recursos, siendo el segundo programa.

**Día 4**

**Tema 4 Polimorfismo.**

* Definición.
* Clases abstractas: definición, métodos abstractos, implementación de clases abstractas, modelado de clases abstractas.
* Interfaces definición, implementación de interfaces, herencia de interfaces.
* Variables polimórficas (plantillas): definición, uso y aplicaciones.
* Reutilización de código.

El alumno creara un programa final utilizando herencia y polimorfismo abstrayendo de otras clases reutilizando recursos ya creados en el transcurso del taller y se creara un punto de venta.

**Tema 5 Excepciones.**

* Definición.
* Tipos de excepciones.
* Propagación de excepciones.
* Gestión de excepciones: manejo de excepciones, lanzamiento de excepciones.
* Creación y manejo de excepción es definidas por el usuario.

El alumno le dará a su proyecto final punto de venta un aspecto más profesional a su programa realizando excepciones que en dichos campos vayan cuyos valores numéricos, o textos, para que así no truene el programa creándolos por TRY y CATCH

**COMPETENCIAS A DESARROLLAR EN EL TALLER DE PROGRAMACION**

Competencias específicas:

* Utilizar técnicas de modelado para la solución de problemas.
* Aplicar la sintaxis de un lenguaje orientado a objetos.
* Aplicar un lenguaje orientado a objetos para la solución de problemas.

Competencias que se desarrollaran después del taller:

* Conocimientos básicos de la carrera.
* Habilidades del manejo de la computadora.
* Solución de problemas.
* Toma de decisiones.
* Manejo y aprendizaje de la autoayuda de Netbeans