**ACTIVIDAD DE APRENDIZAJE**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Sigla Asignatura** | | **POO2201** | **Nombre Asignatura** | | | **Programación Orientada a Objetos I** | | | |
| Créditos | 10 | Hrs. Semestrales Totales | | 90 | Requisitos | | POO1501 | Fecha Actualización | 23-ABRIL-13 |

|  |  |  |  |
| --- | --- | --- | --- |
| Escuela o Programa Transversal | ESCUELA DE INFORMÁTICA Y TELECOMUNICACIONES | Currículum | 1446703 |
| Carrera/s | **INGENIERÍA EN INFORMÁTICA**  **INGENIERÍA EN GESTIÓN DE TECNOLOGÍAS DE INFORMACIÓN**  **ANALISTA PROGRAMADOR COMPUTACIONAL** | Unidad de Aprendizaje | Herencia y Colecciones |

|  |  |  |
| --- | --- | --- |
| **APRENDIZAJE(S) ESPERADO(S)**  Construye clases abstractas para que sean heredadas.  Programa clases e interfaces en una aplicación orientada a objetos para resolver un problema.  Programa clases que utilizan herencias en una aplicación orientada a objetos para resolver un problema.  Construye clases que cuenten con atributos del tipo colecciones para resolver un problema. | | |
| **NOMBRE DE LA ACTIVIDAD**  Farmacia | | |
| **Ambiente de aprendizaje**  √ Presencial  □ No Presencial  □ Mixta | **Duración de la actividad de aprendizaje (horas pedagógicas presenciales y no presenciales):**  10 horas  Presentación Inicial 3 horas  Desarrollo de la actividad 5 horas  [Heteroevaluación](file:///F:\Correcciones\PDA%20terminado\PDA%20terminado\Unidad%20II\Experiencia%203\Heteroevaluación%20Discográfica.docx) 1 hora  Ejecución del docente y conclusiones 1 hora | |
| **Forma de trabajo:**  □ Individual  √ Grupal   * Tamaño del grupo:   √ 2 □ 3-5 □ 6-8 | **Forma de retroalimentación:**  **¿Quién evalúa?**  □ Estudiante  √ Pares  □ Docente  **¿Cómo se evalúa?**  √ Rúbrica  □ Escala de valoración  □ Lista de cotejo  Otra: | |
| **Infraestructura (lugar):**  □ Sala de clases  √ Laboratorio con Netbeans  □ Taller  □ Terreno  □ Otros | | **Insumos y equipamiento para la actividad de aprendizaje:**  PC |
| **Recursos de información:**  √ Textos  □ Recursos audiovisuales  □ Recursos web  √ Programas computacionales: NetBeans | | **Recursos didácticos:**  □ Textos  √ Recursos audiovisuales: Video del programa  □ Recursos web  √ Programas computacionales: Pseudocódigo fuente |
| DESCRIPCIÓN DE LA ACTIVIDAD: Caso e Instrucciones  CASO  La Farmacia `La Sanita`, está automatizando sus procesos administrativos. Dentro de estos procesos, está la venta de sus medicamentos y suplementos alimenticios. Se solicita su participación dentro del equipo de desarrollo. Para ello se le anexa la descripción de cada clase que le corresponde implementar en Java.    La cadena de farmacias maneja dos tipos de productos: medicamentos y suplementos alimenticios. Existe un tipo particular de medicamento denominado “medicamentos formulados” de los cuales se almacenan sus contraindicaciones. Para los suplementos alimenticios es importante almacenar la información referente a las vitaminas que contienen.  A continuación se describen las clases:  Producto: código, precioBase (mayor a 0) y nombre (largo mínimo 3 caracteres).  Medicamento: genérico (si es genérico o no).  SuplementoAlimenticio: cantidadVitaminas e informacionVitaminas.  Formulado: contraindicaciones.  La interface Controlable tiene:  descuento: 10%  Métodos solicitados:  a) descontar: realiza el descuento a los productos si el día de la semana es “lunes” (parámetro del  método).  Si es medicamento se aplica sólo a los genéricos.  Si es suplemento alimenticio se aplica a todos.  b) mostrar: retorna un String con todas las características de un Producto.  Métodos solicitados:   1. recargar: El precio de venta de un producto se calcula dependiendo de si es un medicamento o un suplemento alimenticio. Para los medicamentos, el precio de venta se calcula aplicando un recargo al precio base. Si se trata de un medicamento genérico no se aplica recargo, mientras que, si es un medicamento NO genérico se aplica un 20% de recargo. En cuanto a los suplementos alimenticios, el precio se incrementa en un 2% por cada vitamina que contiene.   Este método es abstracto y se define así en la clase Producto.   1. totalizar: El precio total de la venta se calcula: cantidad (parámetro) \* precioBase considerando las recargas y descuentos.     Crear una clase que utilizando una colección permita lo siguiente:    Clase RegistroProductos   1. Agregar Producto (Medicamento o Suplemento Alimenticio), sólo si no existe el código. 2. Listar todos los Productos (Medicamento o Suplemento Alimenticio). 3. Eliminar Producto: Eliminará todos los Medicamentos cuyo precio base es menor a $1.000, indicando cuántos fueron.   Desde la clase Principal (main()):   * Crear medicamento y obtener el total. * Invocar a todos los métodos de la clase RegistroProductos. | | |