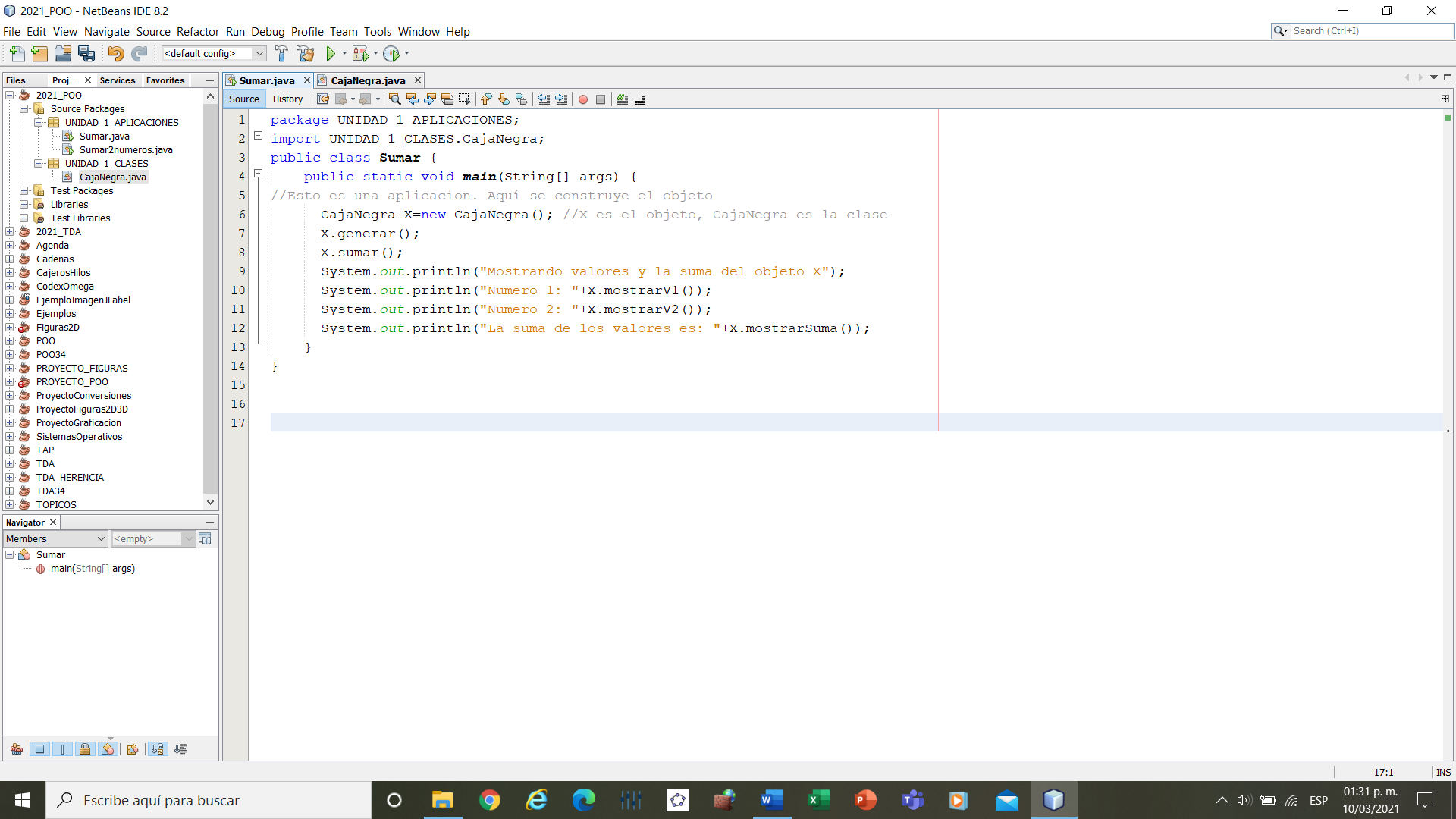


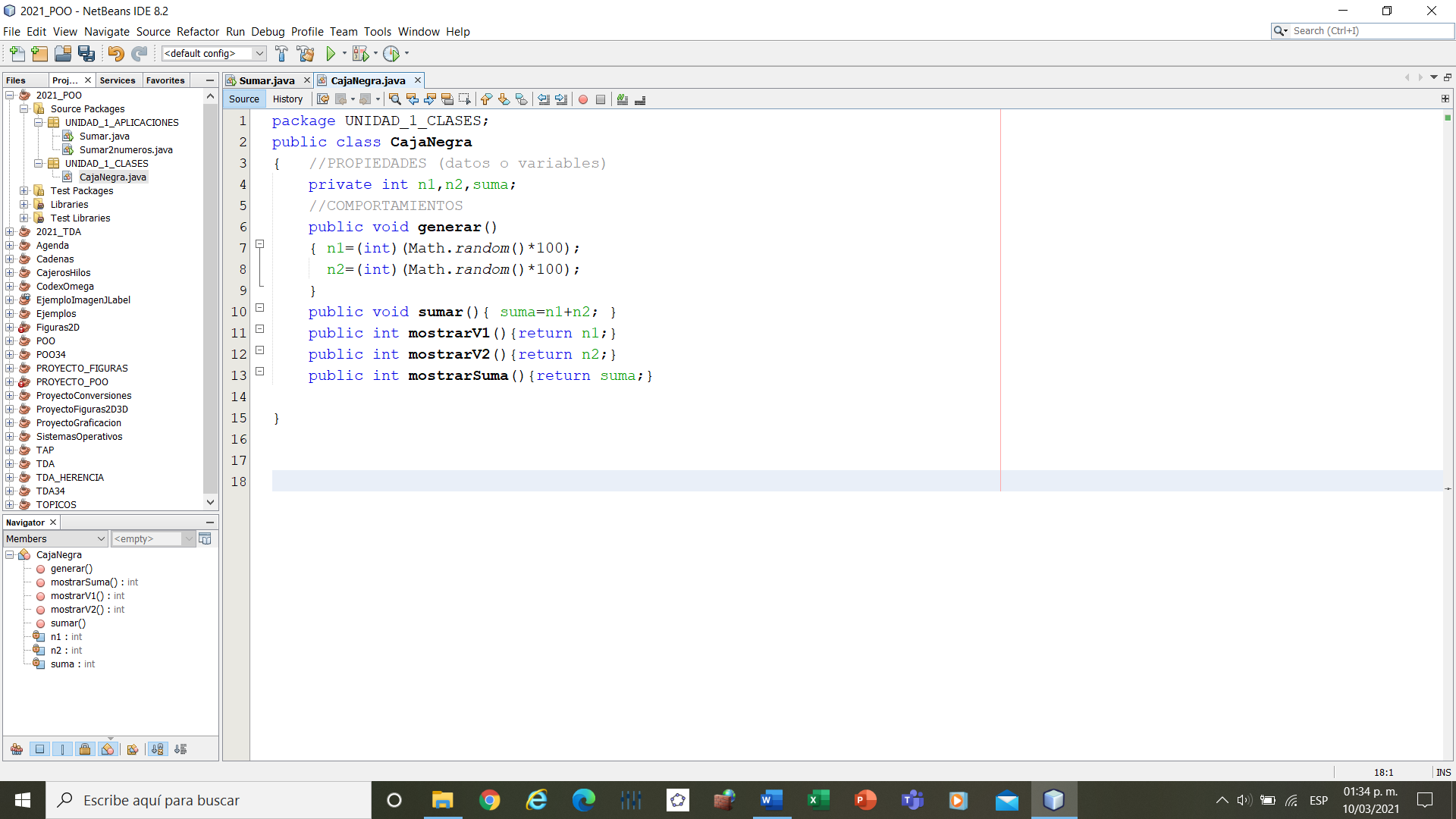
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NOMBRE DEL ALUMNO** | | **NUM. CTROL.** | | **MATERIA** | **HORA** |
|  | |  | | **POO** |  |
| **UNIDAD** | **TEMA** | | **SUBTEMA** | | |
| **1** | **Introducción al paradigma de la programación orientada a objetos** | | * 1. **Elementos del modelo orientado a objetos: clases, objetos, abstracción, modularidad.** | | |

**INSTRUCCIONES: Observa detenidamente los siguientes archivos (clase y la aplicación).**

**APLICACIÓN**



**CLASE**



**TRABAJO DE PARTICIPACION (VALOR 25 PUNTOS)**

**INSTRUCCIONES: Responde las siguientes preguntas que tiene que ver con el código de la APLICACIÓN (Donde se crea el objeto) y la CLASE (Donde se detallan los comportamientos).**

* **Puedes responder con tus palabras**
* **Puedes reforzar tus respuestas escribiendo la línea de código que refuerce tu respuesta.**

|  |  |
| --- | --- |
| **APLICACIÓN**  **(Valor 3 puntos)** | **¿Cómo se construye un objeto?**  **Instanciando la clase.**  **Primero escribimos el nombre de la clase que vamos a utilizar y le damos un nombre al objeto, después instanciamos la clase que vamos a utilizar creando el constructor.** |
| **APLICACIÓN**  **(Valor 3 puntos)** | **¿Qué comportamientos tiene el objeto?**  **El objeto tiene todos los comportamientos que se hayan creado en la clase para el objetivo del programa. En este caso los comportamientos son los siguientes:** |
| **APLICACIÓN**  **(Valor 3 puntos)** | **¿El objeto, expresa o refleja claramente lo que hace, por medio de sus comportamientos?**  **Si, ya que cada método fue nombrado basado en la acción que hace.** |
| **CLASE**  **(Valor 3 puntos)** | **¿En qué parte del código crees que se activa el encapsulamiento?**  **Al crear las propiedades** |
| **CLASE**  **(Valor 3 puntos)** | **Busca en internet las características de los modificadores de acceso a una clase, propiedades (variables) o comportamientos (métodos)**  **Acceso público (public):**  **El método de acceso público permite que cualquier clase en cualquier paquete puede acceder al método.**  **Acceso privado (private):**  **El método de acceso privado permite que solo en la clase en que se creo se pueda acceder a los datos de ese método**  **Acceso protegido (protected):**  **El modificador de acceso protected puede aplicarse a todos los miembros de una clase, es decir, tanto a campos como a métodos o constructores. En el caso de métodos o constructores protegidos, estos serán visibles/utilizables por las subclases y otras clases del mismo package. El acceso protegido suele aplicarse a métodos o constructores, pero preferiblemente no a campos, para evitar debilitar el encapsulamiento. En ocasiones puntuales sí resulta de interés declarar campos con acceso protegido.** |