**Definiciones:**

**Ingeniería de software:** es la aplicación de un proceso sistemático, disciplinado y cuantificable al proceso de desarrollo operación y mantenimiento de software.

**Modelo:** representación o especificación desde un determinado punto de vista y con un objetivo en concreto.

**Diseño:** conjunto de planes y decisiones para definir un producto con los suficientes detalles como para permitir su realización física de acuerdo a unos requisitos.

**Patrón de Diseño:** solución probada que se puede aplicar con éxito a un determinado tipo de problemas que aparecen repetidamente en el desarrollo de software.

**Prototipo y Fábrica Abstracta.**

Ambos son patrones de creación de objetos. El patrón Prototipo especifica el tipo de objetos a crear usando una instancia prototipada y creando objetos copiando dicho prototipo. Sin embargo, la fábrica abstracta es un patrón que nos permite crear familias de objetos relacionados o dependientes entre ellos sin necesidad de especificar sus clases concretas. Ambos patrones se diferencian en que el prototipo instancia clases de una en una mientras que la fábrica abstracta es capaz de generar muchos objetos de golpe.

**Componente Software y Clase.**

Un componente software es una unidad de composición con todos sus interfaces y dependencias de contexto explícitamente definidas, mientras que una clase es una abstracción de un concepto físico. Ambos son abstracciones de conceptos, sin embargo, el componente está formado por muchas clases y dependencias. La clase, sin embargo, es sólo una unidad de clase.

**Ejercicio 1:**