Private: Es un atributo o método propio de la clase, nadie más puede consultar un atributo o ejecutar un método private, ni siquiera las instancias de la clase.

Protected: Solo se puede acceder a sus atributos clases de su mismo paquete.

* Puedo tratar las clases hijas como si fuesen una instancia de la clase padre, a esto se le llama herencia, también a capacidad de reemplazar comportamientos de la clase padre en la clase hijo (override).
* La sentencia final me permite no poder cambiar variables, métodos o clases.
* Encapsulamiento es decidir qué es publico y qué es privado para la clase.

UML y Patrones de Diseño

SDLS (Software Development Life Cycle)

-Recopilación de requerimiento: Considerar los intereses de las personas involucradas en el proyecto de software.

-Análisis de requerimientos: Considerar detalles necesarios para el programa.

-Diseño de software: Diseñar la solución con UML, sin líneas de código.

-Implementación: Diseño del programa con líneas de código.

-Prueba: Pues eso.

- Instalación y mantenimiento: Publicación del software.

Modelos y Métodos para el diseño de software:

1. Modelo cascada – waterfall:

Se da una cantidad de dinero en un proceso que no se está seguro si servirá para cuando se distribuya el programa.

1. Modelo en espiral:

Se programan pequeños objetivos para dar avances para la empresa, con el objetivo de obtener feedback para mejorar el proyecto. Al ser avances pequeños se disminuye el riesgo de perder dinero y tiempo que no llevará a nada. Tiene muchas iteraciones para hacer el proyecto, por lo que también es muy costoso.

1. Agile:

De esto se proponen soluciones para el diseño de software.

Ejemplos: Scrum, XP, etc.

Ingeniería de software:

Crear software de calidad, que funcione y mantenible. El código debe ser reusable y que se ajuste al tiempo y presupuesto del proyecto.

Instituciones de estándares:

ISO: International Standart Organization.

SEI