TADs | Tipos abstractos de datos

* **Abstracción**: consiste en excluir los detalles no deseados y poco significativos del problema a resolver.
* **Abstracción de datos**: Inventar nuevos tipos de datos, comúnmente conocidos como TADs.
* TADs se definen para los siguientes propósitos:
  + Modelar el problema.
  + Esconder los detalles.
* TAD = datos + operaciones.
* De un TAD no se conoce la implementación.
* El ejemplo más común de un TAD son los tipos de datos simples (no se conoce de su implementación, solo se usan).

Ventajas de los TADs

* Separa la implementación de la especificación o diseño de tipos.
* Permite la extensibilidad o evolución de los sistemas.

UNA APLICACIÓN DE SOFTWARE ES UN CONJUNTO DE TAD’s QUE SE RELACIONAN ENTRE SÍ, TAL COMO SE RELACIONAN EN EL MUNDO, POR LO CUAL SI HAY CAMBIOS EN EL MUNDO, LA LOCALIZACIÓN DE LOS CAMBIOS EN LA APLICACIONES MAS FACIL.

Estructuras de datos y TAD’s

* Estructura de datos: Son nuevos tipos de datos, que representan colecciones de datos y que tienen un comportamiento típico.
* Metodología de diseño de un TAD:
  + Establecer una representación abstracta.
  + Determinar condiciones básicas sobre los datos(determinar la invariante).
    - Invariante: Establecer condiciones sobre la estructura interna del TAD y sus componentes.
      * Ejemplo: Coordenada x puede ser un número real cualquiera.
  + Definir las operaciones sobre los datos del TAD.
    - Cada operación debe de llevar:
      * Nombre.
      * Objetivo.
      * Entradas.
      * Salidas.
      * Clasificación.
  + Indicar el manejo del error.
* Resulta necesario tener una representación abstracta del objeto del mundo sin comprometerse con el lenguaje de programación.

Diseño del TAD:

1. Crear una representacion abstracta del objeto (UML o cualquier forma de representacion).
2. Definir la invariante del TAD.
3. Definir las operaciones sobre el TAD.
   1. Nombre.
   2. Tipos de datos que reciben y retornan.
   3. Breve descripción de lo que hacen.
   4. Clasificar las operaciones del TAD.
      1. Constructora: crea instancias del TAD.
      2. Modificadora: cambia el valor de un atributo del TAD.
      3. Analizadora: Consulta valores.
      4. Salida en pantalla: imprime consultas de valores.
      5. Destructora: Destruye una instancia del TAD.
      6. Comparación: Compara dos instancia del TAD.
      7. Por copia: Crea una instancia del TAD compiada de otra.
      8. De persistencia:Salva/carga los componentes del TAD.

* Nota: Formato

**nombreOperacion:tiposEntrada->tiposSalida**

**// comentario acerca del proposito**

4. Manejo del error en los TADs.

1. Capturar el error y definir cómo se maneja.

