## Programación 2

Lenguaje Java - Manejo de Excepciones (*3era parte*)

Típos de Clases

Profesor: Eduardo Godoy. eduardo.gl@gmail.com

Escuela de Ingeniería Civil Informática. Universidad de Valparaíso.

23 de octubre de 2017

### Contenido

1 Acceso a Datos

2 Java Swing: Data Table

# Manejo de archivos - Archivo Secuencial

- Forma básica de almacenar registros en un archivos.
- Sus registros son de similares en exstructura y tamaño.
- Se ordenan de forma secuencial en base a un valor de campo o atributo, en nuestro caso un ID.

## Manejo de archivos

- Archivo Origen o Base: Encargado de almacenar los registros de forma persistente. Además gestiona y mantene su estado en el tiempo.
- Archivo Auxiliar: Encargado de dar soporte a las operaciones básicas sobre el archivo origen, permitiendo hacer cambios de estado. Debido a que se emplea como medida de respaldo.
- Las operaciones básicas son los métodos: crear, actualizar y eliminar

# Manejo de archivos - crear()

- Se recorre archivo de inicio a fin.
- Se procede a la copia de registros desde archivo origen a archivo auxiliar
- Por cada linea leida se guarda su id siendo este sobre escrito por el valor id de la siguiente linea procesada.
- Cada Registro Leído se almacena en el archivo axiliar.
- Se continua con la inserción del nuevo registro al final del archivo auxiliar con el ultimo id obtenido
- incrementado en 1, con esto se obtienen claves únicas para cada registro.
- Se finaliza con el cipiado del archivo auxiliar al orchivo base.

# Manejo de archivos - actualizar()

- Se recorre archivo de inicio a fin.
- Se procede a la copia de registros desde archivo origen a archivo auxiliar
- Por cada linea leida se comparar su id con el id pasado por parámetro.
- De ser iguales se inserta en el archivo auxiliar el registro actualizado que viene como pa rámetro,
- dejando el registro leido desde origen sin ser escrito en auxiliar.
- De lo contrario se siguen escribiendo los registros en el archivo auxiliar sin ser afectados por cambios.
- Se finaliza con el cipiado del archivo auxiliar al orchivo base.

# Manejo de archivos - eliminar()

- Se recorre archivo de inicio a fin.
- Se procede a la copia de registros desde archivo origen a archivo auxiliar
- Por cada linea leida desde origen se compara su id con el id pasado por parámetro.
- De ser iguales ese registros se marca como nulo.
- Cada registro Leído se almacena en el archivo axiliar. excepto el que ha sido marcado como nulo.

# Manejo de archivos - grabar()

- Genera una copia del archivo origen con un nombre de respaldo.
- Toma el archivo origen y borra su contenido.
- Escribe la cabecera del archivo en origen.
- Recorre el archivo auxiliar leyendo cada registro.
- Para cada registro leido desde auxiliar es escrito en origen.

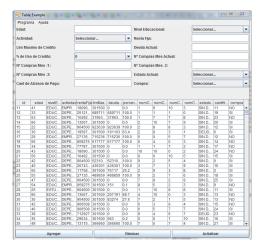
## Concepto de mantenedor

- Interfaz de Usuario que permite administrar una base de datos mediante las operaciones de agregar, modificar y eliminar.
- Utilizados para gestionar los registros almacenados en una base de datos, ya sea, una tabla de datos, un archivo u otra fuente de almacenamiento.

### Concepto de mantenedor - Java

- Componente Java Swing cuya responsabilidad es la tabulación de datos.
- Java provee de varios componentes swing para poder implementar de forma correcta mantenedores de fuentes de datos.
- En general para esto se utiliza el componente JTable.
- Se compone de Filas y Columnas.
- Es posible seleccionar filas y extraer su valo mediante eventos de seleccion con perisfericos. Ejemplo: Mouse.

### Tips - Ocultamiento de Información - Ejemplo.



#### JTable: Instanciar Nueva

#### Ejemplo.

### JTable: Agregar Regitro.

#### Ejemplo.

```
public void agregarFila(Persona p) {
    DefaultTableModel model = (DefaultTableModel) table.
       getModel();
    model.addRow(new Object[] {
        p.getId(), p.getEdad(), p.getNivelEduc(),
        p.getActividad(), p.getRentaFija(),
        p.getLimMaxCredito(), p.getDeudaActual(),
        p.getPorcentUsoCredito(), p.
            getNumCompraMesActual(),
        p.getNumCompraMesActual_1(), p.
            getNumCompraMesActual_2(),
        p.getNumCompraMesActual_3(), p.getEstadoActual
            (),
        p.getCantHistAtrasosPago(), p.getCompra()
    });
```

Eduardo Godoy Programación 2 23 de octubre de 2017 13 / 14

## JTable: Obtener Regitro.

#### Ejemplo.

```
class EliminarBotonListener implements ActionListener{
  @Override
  public void actionPerformed(ActionEvent e){
    int row = table.getSelectedRow();
    String id = table.getModel().getValueAt(row, <N\'
        umero de campo a extraer>).toString();
    ...
}
```