



INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO  
ACADEMIA DE INGENIERÍA DE SOFTWARE  
DEPARTAMENTO DE INGENIERÍA EN SISTEMAS  
COMPUTACIONALES



**Número de Páctica: 1**

**Nombre de la Práctica:** Aplicación Web para manejos de movimientos de inventarios.

**Problemática:** Desarrollar Aplicaciones Web con la especificación Java Servlets con conexión a base de datos relacionales.

**Elemento de Competencia:** Utilizar un Sistema Manejador de Base de datos como motor de persistencia y dar soporte a los datos procesados por la aplicación Web a realizar, del mismo modo utilizar la especificación Java Servlets para desarrollar una aplicación con acceso a base de datos entablando la comunicación entre los mismos utilizando el API JDBC como canal de comunicación.

**Requerimientos:**

**Hardware:**

- Laboratorio de Cómputo.
- Computadora Personal.
- Red Computacional.
- Proyector.

**Software:**

- Sistema Operativo.
- Sistema Manejador de Base de Datos(MySql versión x.x).
- Java Development Kit (JDK 8 o superior).
- Apache NetBeans IDE.
- MySQL Connector/J para la version del manejador de datos.

**Actividades Previas:**

El alumno deberá presentarse en el laboratorio con el diseño lógico de una Base de Datos Relacional, con las características solicitadas por el docente.

## Fundamentos Teóricos:

- ♣ **Bases de Datos.** Instrucciones básicas de SQL en su forma mas simple para interactuar con el manejador MySql.

Instrucción	Detalle
<b>create database nombreBaseDatos;</b>	Comando para crear una base de datos.
<b>use database nombreBaseDatos;</b>	Le indica a MySQL que use la base de datos nombreBaseDatos como la base de datos por defecto para los comandos siguientes.
<b>CREATE TABLE [IF NOT EXISTS] <i>tbl_name</i></b>	crea una tabla con el nombre dado.
<b>SELECT <i>select_expr</i>, ...[FROM <i>table_references</i>] [WHERE <i>where_definition</i>]</b>	<p>se usa para recibir registros seleccionados desde una o más tablas.</p> <p>Cada <b><i>select_expr</i></b> indicata una columna que quiere recibir.</p> <p><b><i>table_references</i></b> indicata la tabla o tablas desde la que recibir registros</p> <p><b><i>where_definition</i></b> consiste en la palabra clave WHERE seguida por una expresión que indica la condición o condiciones que deben satisfacer los registros para ser seleccionados.</p>
<b>INSERT [INTO] <i>tbl_name</i> [(<i>col_name</i>,...)] VALUES { <i>VALUE</i>}</b>	Inserta una nueva fila en una tabla existente.
<b>UPDATE <i>tbl_name</i> SET <i>col_name1</i>=<i>expr1</i> [, <i>col_name2</i>=<i>expr2</i> ...] [WHERE <i>where_definition</i>]</b>	<p>El comando UPDATE actualiza columnas en registros de tabla existentes con nuevos valores.</p> <p><b>La cláusula SET</b> indica qué columna modificar y los valores que puede recibir.</p> <p><b>La cláusula WHERE</b> , si se da, especifica qué registros deben actualizarse. De otro modo, se actualizan todos los registros.</p>
<b>DELETE FROM <i>tbl_name</i> [WHERE <i>where_definition</i>]</b>	<p>Borra los registros de <i>tbl_name</i> que satisfacen la condición dada por <b><i>where_definition</i></b>, y retorna el número de registros borrados.</p> <p>Si realiza un comando DELETE <b>sin</b> cláusula <b>WHERE</b> se borran todos los registros</p>

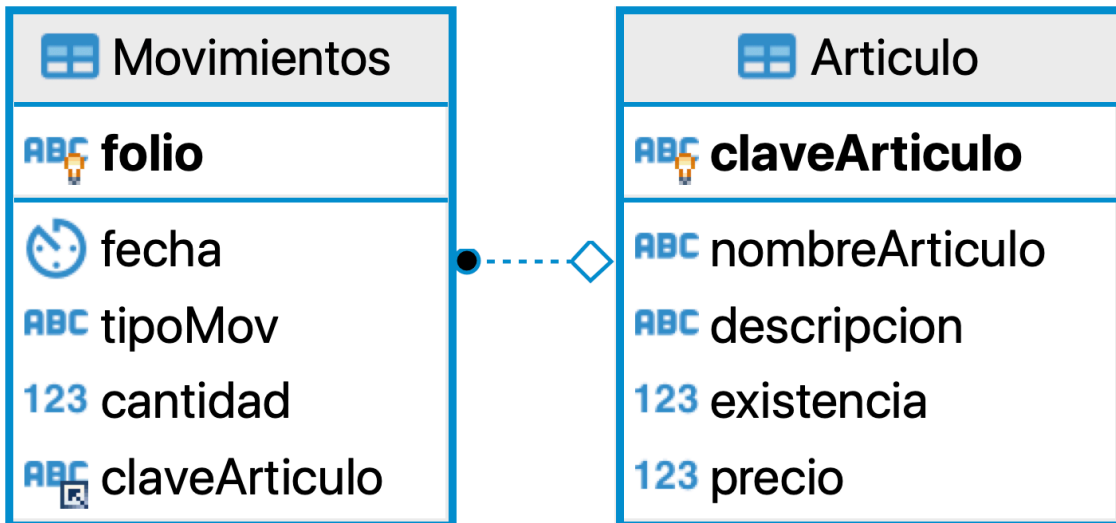
- ♣ **Servlet Java.** Ejemplo básico para obtener un parametro proveniente de una petición y posteriormente interactuar con la base de datos eliminando el registro coincidente.

```
@WebServlet(name = "EliminarArticulo", urlPatterns = {"/EliminarArticulo"})
public class EliminarArticulo extends HttpServlet {
    /** Processes requests for both HTTP <code>GET</code> and <code>POST</code> ...9 lines */
    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
            out.println("<!DOCTYPE html>");
            out.println("<html>");
            out.println("<head>");
            out.println("<title> Eliminar Articulo</title>");
            out.println("<script src='https://cdn.jsdelivr.net/npm/sweetalert2@9'></script>");
            out.println("</head>");
            out.println("<body>");
            Articulo a = new Articulo();
            a.setClaveArticulo(request.getParameter("id"));
            ArticuloDAO dao = new ArticuloDAO();
            try {
                dao.delete(a);
                out.println(
                    "<script type='text/javascript'> "
                    + " Swal.fire('Artículo','El registro se eliminó satisfactoriamente','success') "
                    + "</script>"
                );
                RequestDispatcher rd = getServletContext().getRequestDispatcher("/MostrarArticulo");
                rd.include(request, response);
            } catch (SQLException e) {
                out.println(
                    "<script type='text/javascript'> "
                    + " Swal.fire('Artículo','Problemas al Eliminar'" + e.toString() + "','error') "
                    + "</script>"
                );
            }
            out.println("</body>");
            out.println("</html>");
        }
    }
}
```

HttpServlet methods. Click on the + sign on the left to edit the code.

Instrucciones:

1. Crear la Siguinete Base de datos:



2. Crear una aplicación Web que utilice la especificación Servlet Java para dar soporte a lo indicado en las siguinetes pantallas.

A Web Page

http://

### Registrar Movimiento

Fecha del Movimiento: 2020/02/13

Tipo de Movimiento: E  
S

Cantidad: 5

Clave del Artículo: Computadora  
Celular  
Lavadora

Registrar Movimiento

El tipo de Movimiento almacena un caracter que se utilizará para aumentar o disminuir la cantidad de existencias según el Tipo de Movimiento.

La cantidad se utilizará para comparar la existencia contra la cantidad solicitada cuando se trate de un movimiento de tipo S (Salida).





Clave del Artículo libre de utiizar un select o un autocomplete para mostrarle al usuario el nombre del Artículo a afectar. Aunque se trabaje con el valor interno correspondiente a la clave del Artículo.

No se podra registrar una salida del inventario si las existencias son menores a la cantidad solicitada

A Web Page

http://

### Movimientos de Artículos

Folio	Fecha	Tipo Movimiento	Clave Artículo	Cantidad	Acciones
1	2020-02-13	E	art01	10	 
2	2020-02-13	S	art02	10	 

MiniInventario

El registro se eliminó satisfactoriamente

En caso de seleccionar la opción eliminar deberá mostrar el respectivo mensaje de "El registro se eliminó satisfactoriamente" y volver a la vista de Movimientos de Artículos la cual deberá aparecer sin el registro previamente eliminado.

En caso de elegir la opción actualizar deberá de mostrarse la vista indicada en la imagen siguiente.


A Web Page


http://


### Actualizar Movimiento

Folio

Clave del Artículo

Fecha del Movimiento  

Tipo de Movimiento  

Cantidad  

El folio deberá aparecer en un aetiqueta.

La clave del Artículo deberá aparecer en un cuadro de texto de solo lectura.

Ustedes decidan si pueden o no actualizar la Fecha del Movimiento.

Tipo de Movimiento deberá aparecer en un cuadro de texto de solo lectura.

La cantidad deberá ser editable.

En caso de actualizar la cantidad afectada en el tipo de movimiento, se debera obtener el valor anterior de dicho movimiento, aumentar la existencia en caso de un error en la cantidad de salida.

En caso de ser un movimiento de tipo entrada debera hacer lo conducente.

3. Se deberá de realizar las actividades pertinentes a los métodos CRUD, para la entidad Movimientos de la Base de Datos Antes Mostrada.
4. Entregar Reporte de Práctcas según formato establecido.
5. Mostrar el funcionamiento de la aplicación en la fecha indicada.
6. Entrgar el proyecto completo (Código Fuente, Bibliotecas empleadas, Script de base de datos, etc).

