	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		


# Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación

---

Universidad Politécnica Salesiana

## Vicerrectorado Docente

Código del Formato:	GUIA-PRL-001
Versión:	VF1.0
Elaborado por:	Directores de Área del Conocimiento Integrantes Consejo Académico
Fecha de elaboración:	2016/04/01
Revisado por:	Consejo Académico
Fecha de revisión:	2016/04/06
Aprobado por:	Lauro Fernando Pensantes Aveles Vicerrector Docente
Fecha de aprobación:	2016/14/06
Nivel de confidencialidad:	Interno

	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

## Descripción General

### Propósito


El propósito del presente documento es definir un estándar para elaborar documentación de guías de práctica de laboratorio, talleres o centros de simulación de las Carreras de la Universidad Politécnica Salesiana, con la finalidad de lograr una homogenización en la presentación de la información por parte del personal académico y técnico docente.

### Alcance

El presente estándar será aplicado a toda la documentación referente a informes de prácticas de laboratorio, talleres o centros de simulación de las Carreras de la Universidad Politécnica Salesiana.

### Formatos


- Formato de Guía de Práctica de Laboratorio / Talleres / Centros de Simulación – para Docentes
- Formato de Informe de Práctica de Laboratorio / Talleres / Centros de Simulación – para Estudiantes


	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		


		<b>FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA DOCENTES</b>	
<b>CARRERA:</b> COMPUTACIÓN		<b>ASIGNATURA:</b> Programación Aplicada	
<b>NRO. PRÁCTICA:</b>	1	<b>TÍTULO PRÁCTICA:</b> Clase Genéricas en Java	
<b>OBJETIVO:</b> Identificar los cambios importantes de Java Diseñar e Implementar las nuevas tecnicas de programación Entender la cada uno de las características nuevas en Java			
<b>INSTRUCCIONES</b> (Detallar las instrucciones que se dará al estudiante):		1. Revisar los conceptos fundamentales de Java	
		2. Establecer las características de Java en programación genérica	
		3. Implementar y diseñar los nuevos componentes de programación genérica	
		4. Realizar el informe respectivo según los datos solicitados.	
<b>ACTIVIDADES POR DESARROLLAR</b> (Anotar las actividades que deberá seguir el estudiante para el cumplimiento de la práctica)			
1. Revisar la teoría y conceptos de Java 8, 9 ,10, 11, 12			
2. Diseñar e implementar las características de Java para generar una abstracción que permita realizar un CRUD,			
3. Probar su funcionamiento y rendimiento dentro de los equipos de cómputo de programación genérica y ordenar una lista, buscar.			
4. Realizar práctica codificando los codigos de las nuevas características de Java y su uso dentro de una agenda telefónica			
<b>RESULTADO(S) OBTENIDO(S):</b> Realizar procesos de investigación sobre los cambios importantes de Java Entender las aplicaciones de codificación de las nuevas características en base a la programación genérica Entender las funcionalidades adicionales de Java.			
<b>CONCLUSIONES:</b> Aprenden a trabajar en grupo dentro de plazos de tiempo establecidos, manejando el lenguaje de programación de Java.			
<b>RECOMENDACIONES:</b> Realizar el trabajo dentro del tiempo establecido.			

**Docente / Técnico Docente:** \_\_\_\_\_

**Firma:** \_\_\_\_\_

	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

		<b>FORMATO DE INFORME DE PRÁCTICA DE LABORATORIO / TALLERES / CENTROS DE SIMULACIÓN – PARA ESTUDIANTES</b>	
<b>CARRERA:</b> COMPUTACIÓN		<b>ASIGNATURA:</b> Programación Aplicada	
<b>NRO. PRÁCTICA:</b>	1	<b>TÍTULO PRÁCTICA:</b> Clase Genéricas en Java	
<b>OBJETIVO ALCANZADO:</b> <ul style="list-style-type: none"> <li>• Identificar los cambios importantes de Java 8.</li> <li>• Diseñar e Implementar las nuevas técnicas de programación aprendidas en las sesiones virtuales.</li> <li>• Entender la cada uno de las características nuevas en Java 8.</li> </ul>			
<b>ACTIVIDADES DESARROLLADAS</b>			
<p>1. Revisar la teoría y conceptos de Java 8, 9 ,10, 11, 12</p> <p>- <b>Arquitectura MVC:</b> Uno de los conceptos aplicados en el desarrollo de esta práctica es la arquitectura MVC en la cual se separa en tres capas (Modelo, Vista, Controlador), cada una de estas está asignada a una función del programa.</p> <p>- <b>Métodos CRUD:</b> Estos métodos directamente relacionados con el almacenamiento de datos, es decir con la base de datos, el nombre es la abreviación de Create, Read, Update, Delete,</p> <p>- <b>Clase Genéricas:</b> Las clases Genéricas son el tipo de clases que sirve para el reutilizamiento de código, disminuyendo la cantidad de líneas de código y encapsula sus operaciones para que no sean utilizadas por un solo tipo de dato.</p> <p>- <b>Javax Swing:</b> Paquete de datos donde se incluye los JFrame para las ventanas, paneles, botones, tablas, etc. Para desarrollar una interfaz gráfica y funcione como intermediario entre el usuario y la aplicación.</p> <p>- <b>Optional:</b> El optional es un objeto contenedor que puede contener o no a un objeto.</p> <p>- <b>TableRowSorter:</b> El TableRowSorter permite ordenar y filtrar las filas de una tabla.</p> <p>- <b>SortKey:</b> Crea una columna con el orden específico de los datos agregados en la tabla.</p>			
<p>2. Diseñar e implementar las características de Java para generar una abstracción que permita realizar un CRUD.</p> <p>Para comenzar a programar un CRUD en una clase genérica tenemos que hacer genérica a la clase que la va a contener estos métodos.</p> <p>Esto se obtiene al agregarle alado del nombre una letra mayúscula entre los símbolos más y menos que.</p>			

	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

```
public class ControladorAgenda <P>{
```

Ya que no disponemos del DAO, se agregó una variable de tipo lista para guardar los datos como objetos para que sea compatible con todos los datos que se desea almacenar y se inicializa en el constructor de la clase.

```
    private List<P> lista;

    public ControladorAgenda() {
        lista = new ArrayList<>();
    }
```

Una vez que esta inicializada la lista se implementa el primer método que es el Create, este método recibe como parámetro una variable tipo objeto y le añade a la lista.

```
    public void create(P objeto) {
        lista.add(objeto);
    }
```

Para implementar el método Read se utilizó el concepto de lambda que fue añadido en Java 8, como parámetro de tipo predicate recibe un argumento del cual se busca mediante streams en la lista hasta coincidir con el objeto buscado y se retorna el mismo encapsulado en un Optional.


```
    public boolean update(P objeto, P objetoNuevo) {
        int posicion = lista.indexOf(objeto);
        lista.set(posicion, objetoNuevo);
        return false;
    }
```

El penúltimo método del CRUD implementado es el Update, en este método recibe dos objetos como parámetro, el primero es el objeto que se va a modificar y el segundo es el modificado. Se obtiene la posición del objeto que se va a modificar con el método .indexOf() y se manda a remplazar el objeto modificado con el valor que se obtuvo con el método .indexOf().

```
    public boolean update(P objeto, P objetoNuevo) {
        int posicion = lista.indexOf(objeto);
        lista.set(posicion, objetoNuevo);
        return false;
    }
```

En el método implementado Delite, recibe un parámetro de tipo objeto, primero se comprueba si el objeto existe en la lista, utilizando el método .contains(), si este existe en la lista se remueve de la lista.

```
    public boolean delite(P objeto) {
        if(lista.contains(objeto)) {
            lista.remove(objeto);
            return true;
        }
        return false;
    }
```

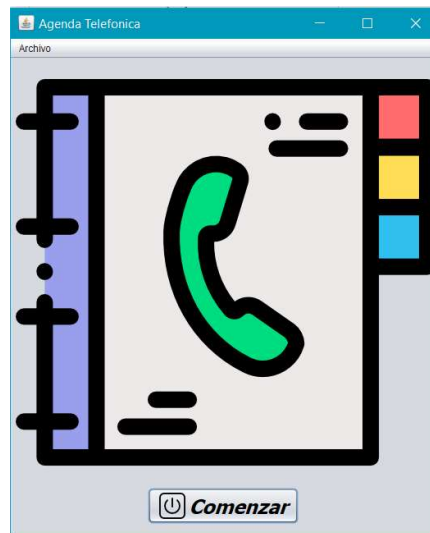
	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

El último método de esta clase devuelve la lista completa de objetos agregados.

```
public List<P> findAll () {
    return lista;
}
```


3. Probar su funcionamiento y rendimiento dentro de los equipos de cómputo de programación genérica y ordenar una lista, buscar.

Cuando ejecutamos el programa primero comienza con una pantalla para comenzar la agenda:




Al aplastar el botón se va a aparecer la pantalla en la que se puede buscar el usuario para mostrar la información perteneciente y añadir, eliminar, editar, listar los teléfonos que desea el usuario:

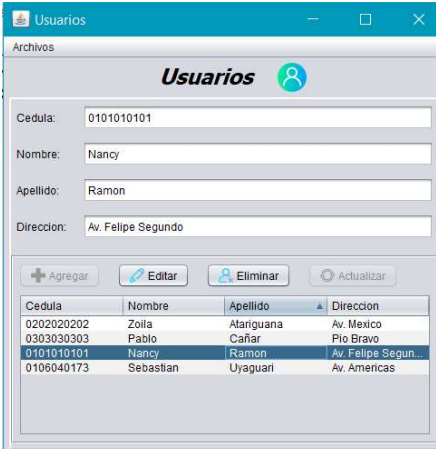


	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

Si no tenemos agregado un usuario en la parte superior de la pantalla, en la barra de menús, está un menú con el nombre de Usuarios, al hacer click ahí aparece la opción en la cual puede controlar los usuarios:



Tenemos la opción de agregar, editar, eliminar y en la tabla inferior se lista todos los usuarios que estén registrados y el botón actualizar tiene la función de actualizar la tabla y una vez que registremos usuarios, estos aparecen ordenados por orden alfabético de los apellidos.




Cedula	Nombre	Apellido	Direccion
0202020202	Zolia	Atariguana	Av. Mexico
0303030303	Pablo	Cañar	Pio Bravo
0101010101	Nancy	Ramon	Av. Felipe Segun...
0106040173	Sebastian	Uyaguari	Av. Americas

Si hacemos click en la tabla se cargan los datos en los textos de la parte superior del usuario seleccionado y se puede editar o eliminar.

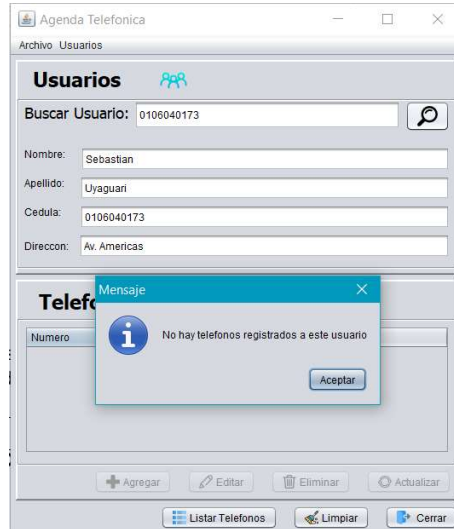
Una vez que ya hemos creado los usuarios podemos buscar en la ventana de los teléfonos al usuario que queremos agregar los teléfonos o el teléfono.

Esta es la línea de código que se empleó para buscar al usuario:

```
Optional<Persona> persona = controladorAgendaPersona.read(p -> cedula.equals (
p.getCedula()));
```

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

Cuando se recibe en el encapsulamiento Optional se des encapsular, para eso se consigue con el método .get() y antes de des encapsular se puede comprobar si es que el opcional no está vacío se utiliza el .isPresent().




Cuando buscamos al usuario se activa los botones para poder agregar y actualizar la lista.

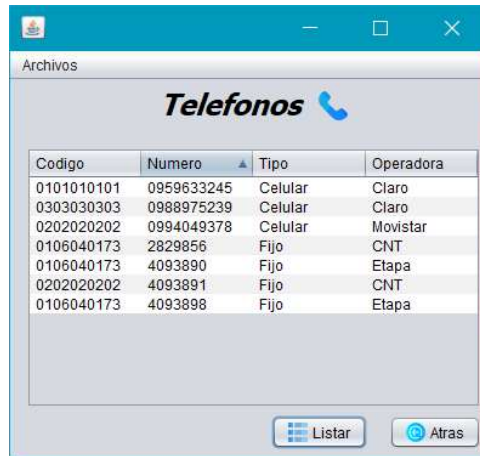


Igualmente al hacer click en el algún teléfono de la tabla se activa los botones de editar y eliminar además se ordena los teléfonos agregados.



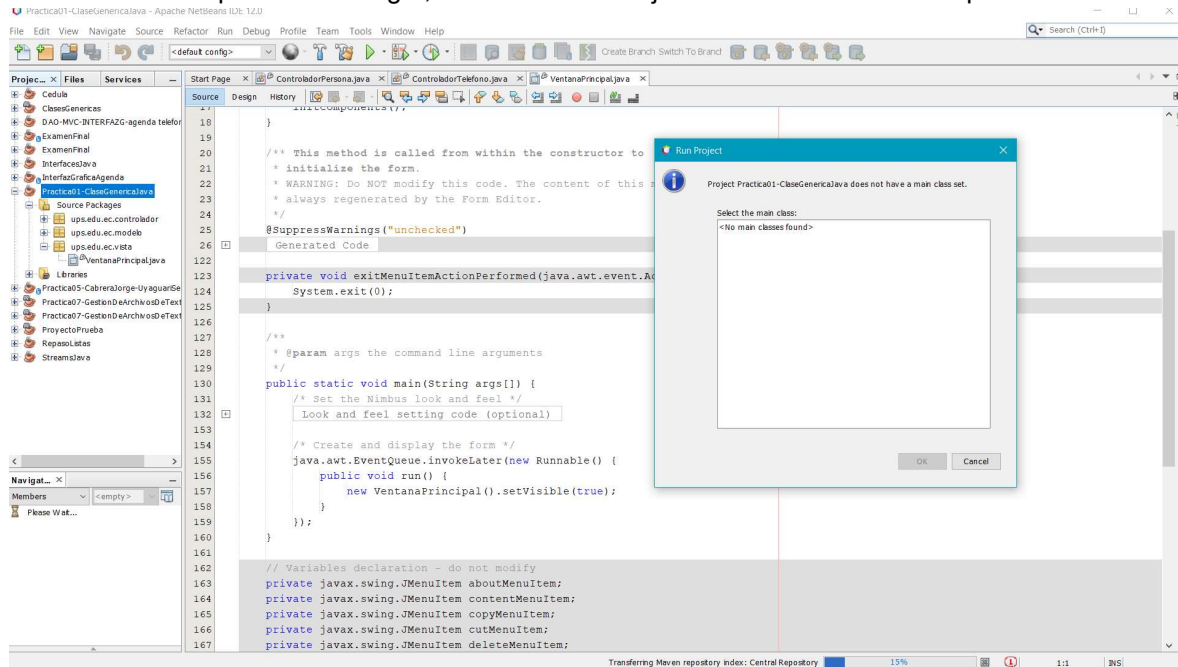
	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		


Por ultimo tenemos la ventana en la que se visualiza todos los teléfonos que estén agregados sin importar a que usuario pertenece.



#### 4. Realizar práctica codificando los códigos de las nuevas características de Java y su uso dentro de una agenda telefónica

En este punto tuve un problema con el Netbeans y no pude integrar las nuevas características de Java 9, 10 11, 12 ya que el Netbeans no me reconoce las clases principales cuando son de tipo JFrame. Trate de resolver pero no conseguí, además no me deja desinstalar el Netbeans por falta de un archivo.



	<b>VICERRECTORADO DOCENTE</b>	<b>Código:</b> GUIA-PRL-001
	CONSEJO ACADÉMICO	<b>Aprobación:</b> 2016/04/06
<b>Formato:</b> Guía de Práctica de Laboratorio / Talleres / Centros de Simulación		

**RESULTADO(S) OBTENIDO(S):**

- Realizar procesos de investigación sobre los cambios importantes de Java.
- Entender las aplicaciones de codificación de las nuevas características en base a la programación genérica.
- Entender las funcionalidades adicionales de Java.

**CONCLUSIONES:**

Este trabajo ha empleado el conocimiento básico adquirido el ciclo pasado de java y los nuevos métodos de programación, pero lo más valioso de este trabajo es que aprendes nuevas funciones que pueden ser aplicadas en el desarrollo de un software.

**RECOMENDACIONES:**

Mayor uso de las herramientas tecnológicas para buscar información sobre el lenguaje de programación que se desee para ampliar los conocimientos que se tiene o se desea.

**Nombre de estudiante:** Sebastián Uyaguari



**Firma de estudiante:**