	<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 Pesántez Avilés Vicerrector Docente
Fecha de aprobación:	2016/14/06
Nivel de confidencialidad:	Interno

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: 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> Investigación de las versiones de Java	
<b>OBJETIVO:</b> Identificar los cambios importantes de Java Diseñar e Implementar las nuevas técnicas 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 las nuevas versiones de Java	
		2. Establecer las características de Java	
		3. Implementar y diseñar los nuevos componentes de programación	
		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 9 ,10, 11, 12			
2. Diseñar e implementar las características de Java,			
3. Probar su funcionamiento y rendimiento dentro de los equipos de computo			
4. Realizar práctica codificando los códigos de las nuevas características de Java.			
<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 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:** \_\_\_\_\_

	VICERRECTORADO DOCENTE	Código: GUIA-PRL-001
	CONSEJO ACADÉMICO	Aprobación: 2016/04/06
Formato: 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> Investigación de las versiones de Java	
<b>OBJETIVO ALCANZADO:</b> Identificar los cambios importantes de Java 12 Entender la cada uno de las características nuevas en Java			
<b>ACTIVIDADES DESARROLLADAS</b>			
<p>1. Revisar la teoría y conceptos de Java 12.</p> <p><b>Java 12:</b>          El 19 de marzo de 2019, Oracle realizó el lanzamiento del JDK12, más conocido como Java 12 incorporando nuevas funcionalidades de forma incremental. No es una versión LTS (Soporte a Largo Plazo) ya que dejará de tener sus actualizaciones cuando Oracle lance Java 13. Según Oracle la siguiente versión LTS será la 17.</p>			
<p>2. Establecer las características de Java</p> <ul style="list-style-type: none"> <li>• <b>Expresiones Switch:</b> Las expresiones <i>switch</i> permiten quitar varias sentencias <i>if else</i> encadenadas. Cada rama de la sentencia <i>switch</i> devuelve un valor y no hace falta usar la sentencia <i>break</i>, se pueden utilizar varios casos para cada rama.</li> <li>• <b>Teeing Collectors:</b> Los <i>stream</i> proporcionan un flujo de elementos a procesar. En el caso de querer recolectar dos valores de ese flujo requiere usar un <i>reduce</i> que complica el código. Se ha añadido un nuevo colector <code>Collectors.teeing()</code> para enviar un elemento de un <i>stream</i> a dos <i>streams</i>.</li> <li>• <b>Devolución rápida de la memoria comprometida no utilizada en G1:</b> Mejora el recolector de basura G1 para devolver automáticamente un conjunto de memoria de Java al sistema operativo cuando está inactivo.</li> <li>• <b>El recolector de basura de bajo tiempo de pausa Shenandoah:</b> Agrega un nuevo algoritmo de recolector de basura que reduce los tiempos de pausa del propio recolector mediante un trabajo de evacuación concurrente con los hilos de Java en ejecución.</li> <li>• <b>Microbenchmark suite:</b> Se ha añadido una suite de microbenchmarks al código fuente del JDK, y la intención de la inclusión de esta característica es hacer más fácil a los desarrolladores ejecutar microbenchmarks existentes y poder crear otros nuevos.</li> <li>• <b>API de constantes en la JVM:</b> Se ha introducido una API para modelar descripciones nominales de archivos de clase clave y artefactos de tiempos de ejecución, en particular las constantes que se pueden cargar desde la agrupación constante.</li> <li>• <b>Archivos CDS por defecto:</b> Mejora el proceso de compilación del JDK a la hora de generar un archivo CDS (Clase de Compartición de Datos) usando la lista de clases predeterminada en plataformas de 64-bit.</li> </ul>			
<p>3. <b>Probar su funcionamiento y rendimiento dentro de los equipos de computo</b></p> <p>Java 12 está disponible en el sitio web de Oracle bajo licencia comercial y como software libre bajo licencia GPLv2 a través del proyecto OpenJDK. Sin embargo, posiblemente la instalación de OpenJDK</p>			

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

fuera de Linux no resulte muy amigable para el usuario, motivo por el cual existe la iniciativa AdoptOpenJDK, desde donde se pueden adquirir binarios precompilados.

#### 4. Realizar práctica codificando los codigos de las nuevas características de Java

```

1  switch (day) {
2      case MONDAY, FRIDAY, SUNDAY -> System.out.println(6);
3      case TUESDAY                  -> System.out.println(7);
4      case THURSDAY, SATURDAY       -> System.out.println(8);
5      case WEDNESDAY                -> System.out.println(9);
6  }
7
8  String numericString = switch (integer) {
9      case 0 -> "zero";
10     case 1, 3, 5, 7, 9 -> "odd";
11     case 2, 4, 6, 8, 10 -> "even";
12     default -> "N/A";
13 };

```

#### Bibliografía:

Medina, E. (2019, 8 octubre). *Oracle anuncia la disponibilidad de Java 12* ». MuyLinux.  
<https://www.muylinux.com/2019/03/22/oracle-jdk-12-java-12/>

P. (2019, 22 marzo). *Novedades de Java 12*. Blog Bitix  
<https://picodotdev.github.io/blog-bitix/2019/03/novedades-de-java-12/>

#### RESULTADO(S) OBTENIDO(S):

Realizar procesos de investigación sobre los cambios importantes de Java  
 Entender las aplicaciones de codificación de las nuevas características  
 Entender las funcionalidades adicionales de Java.


#### CONCLUSIONES:

Aplicar el autoaprendizaje mediante trabajos de investigación.  
 Llegar a poseer más conocimientos sobre las versiones de Java.

#### RECOMENDACIONES:

Ejercer la tarea en los tiempos establecidos por el docente.  
 Buscar fuentes confiables de investigación dentro de la red.  
 Aplicar buenas prácticas de programación.

**Nombre de estudiante:** Edith Anahí Cabrera Bermeo

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

**Firma de estudiante:**

