



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)**

Formato para prácticas de laboratorio

CARRERA	PLAN DE ESTUDIO	CLAVE DE UNIDAD DE APRENDIZAJE	NOMBRE DE LA UNIDAD DE APRENDIZAJE
Ingeniero en Computación	2009-2	12099	Programación Orientada a Objetos

PRÁCTICA No.	LABORATORIO DE	Ingeniero en computación	DURACIÓN (HORAS)
2	NOMBRE DE LA PRÁCTICA	Introducción a Java y al entorno de desarrollo NetBeans	2

1. INTRODUCCIÓN

El entorno de desarrollo integrado (IDE por sus siglas en inglés) NetBeans facilita el proceso de desarrollo de aplicaciones en Java y otros lenguajes de programación. NetBeans es utilizado por los desarrolladores profesionales de todo el mundo y por lo tanto cuenta con una gran variedad de herramientas que pueden ser abrumadoras para un programador principiante. En esta práctica se introducirá al estudiante a algunas de las herramientas que empleará durante el semestre.

Java es un lenguaje de programación orientado a objetos que se utiliza para desarrollar aplicaciones para una gran variedad de plataformas y dispositivos. Comparte mucha de su sintaxis con el lenguaje C, por lo tanto no es complicada la transición para los programadores entre estos dos lenguajes. En esta práctica se realizarán ejercicios que se apoyan en las similitudes de estos dos lenguajes a la vez que se presentan algunas de sus diferencias.

2. OBJETIVO (COMPETENCIA)

Analizar y comprender los métodos de impresiones básicas de lenguaje javas. Utilizando los operadores aritméticos y relacionales con actitud creativa y respetuosa.

Formuló Ing. María Luisa González Ramírez/ M.C. Cecilia M. Curlango Rosas	Revisó M.C. Gloria Etelbina Chávez Valenzuela	Aprobó	Autorizó M.C. Maximiliano de las Fuentes Lara
Nombre y Firma del Maestro	Nombre y Firma del Responsable de Programa Educativo	Nombre y Firma del Responsable de Gestión de Calidad	Nombre y Firma del Director de la Facultad

Código: GC-N4-017
Revisión: 3



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

3. FUNDAMENTO

Operadores Aritméticos

Como se puede ver en las siguientes dos tablas, en Java los operadores aritméticos son iguales que los que se utilizan en C. Las expresiones aritméticas se evalúan de izquierda a derecha aplicando el orden de las operaciones. El uso de paréntesis se emplean cuando se quiere alterar este orden, por ejemplo cuando se quiere realizar una suma antes de una multiplicación

Operador	Nombre	Ejemplo
+	Suma	3+4
-	Diferencia	3-4
*	Producto	3*4
/	Cociente	20/7
%	Módulo	20%7

Expresión	Significado
$x+=y$	$x=x+y$
$x-=y$	$x=x-y$
$x*=y$	$x=x*y$
$x/=y$	$x=x/y$

Operadores Relacionales

Los operadores relacionales también son los mismos en Java que en C. Una diferencia importante es que en Java cuando se comparan dos valores con un operador relacional, el resultado es un valor booleano ya sea **true** ó **false**.

Operador	nombre	ejemplo	significado
<	menor que	$a < b$	a es menor que b
>	mayor que	$a > b$	a es mayor que b
==	igual a	$a == b$	a es igual a b
!=	no igual a	$a != b$	a no es igual a b
<=	menor que o igual a	$a <= 5$	a es menor que o igual a b
>=	mayor que o igual a	$a >= b$	a es menor que o igual a b



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

Operadores Lógicos

Los operadores lógicos se utilizan para unir dos o mas expresiones. Su uso acostumbrado es para unir dos expresiones formuladas con operadores relacionales, por ejemplo, $(5 < 3) \parallel (3 > 4)$.

Operador	Nombre	Descripción
&&	AND	Si ambas expresiones son verdaderas el resultado es verdadero.
	OR	Si alguna de las expresiones es verdadera el resultado es verdadero.
!	NOT	Invierte el resultado de la expresión

Tipos de datos primitivos

La siguiente tabla muestra los tipos de datos que se usan en Java para declarar variables que guardan datos primitivos. Observe que los tipos **byte**, **short**, **int** y **long** almacenan valores enteros, mientras que **float** y **double** se utilizan para valores que utilizan un punto decimal. En la tabla también se muestran los valores mínimos y máximos que se pueden asignar a las variables.

Tipo	Contenido	Valor por defecto	Tamaño	Rango
boolean	true o false	FALSE	1 bit	NA
char	Caracter Unicode	\u0000	16 bits	\u0000 a \uFFFF
byte	Entero con signo	0	8 bits	-128 a 127
short	Entero con signo	0	16 bits	-32768 a 32767
int	Entero con signo	0	32 bits	-2147483648 a 2147483647
long	Entero con signo	0	64 bits	-9223372036854775808 a 9223372036854775807
float	IEEE 754 punto flotante	0	32 bits	$\pm 1.4E-45$ a $\pm 3.4028235E+38$
double	IEEE 754 punto flotante	0	64 bits	$\pm 4.9E-324$ a $\pm 1.7976931348623157E+308$



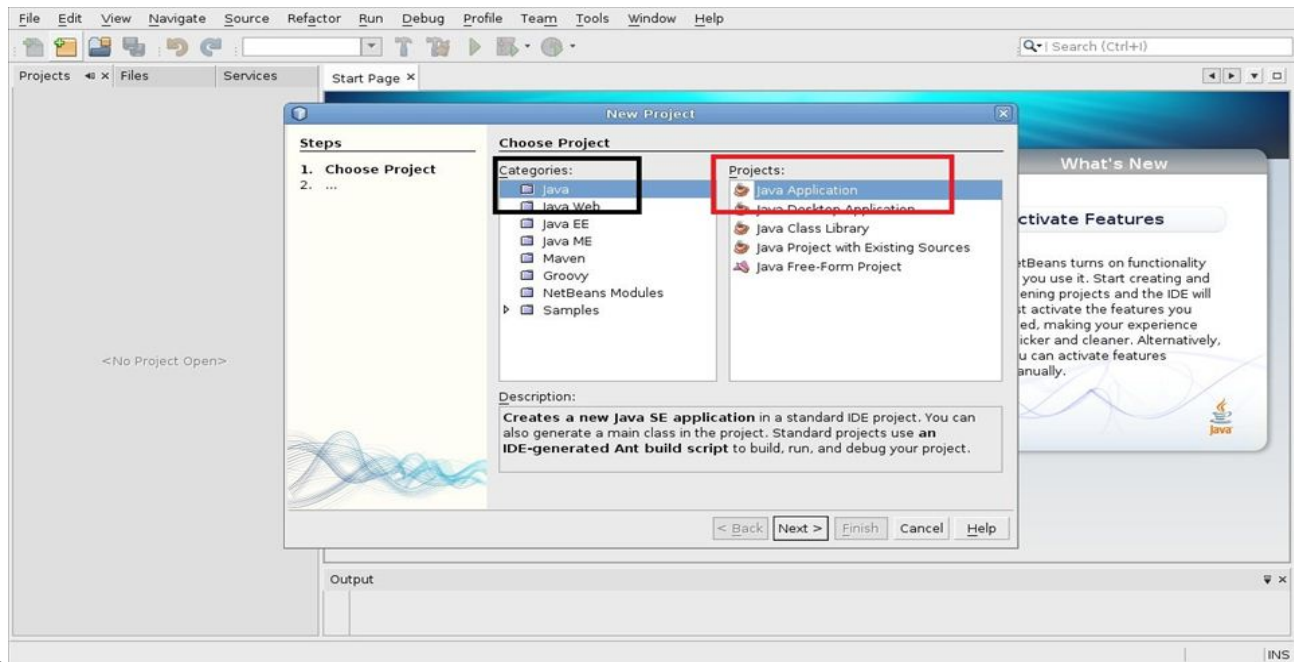
**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

Introducción a NetBeans

Para trabajar en NetBeans, primero debemos crear un proyecto. Esto se logra seleccionando del menú la opción **File --> New Project**.

El proyecto crea una serie de directorios en el sistema, para organizar los archivos de Java y otros



Como se muestra en la figura debemos seleccionar la categoría que debe ser **Java** y el proyecto, que debe ser **Java Application**. Y debemos presionar el botón de **Next>**. Enseguida aparecerá la siguiente ventana donde se solicita el nombre del proyecto.



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name:

Project Location:

Project Folder:

Use Dedicated Folder for Storing Libraries

Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

Create Main Class

Set as Main Project

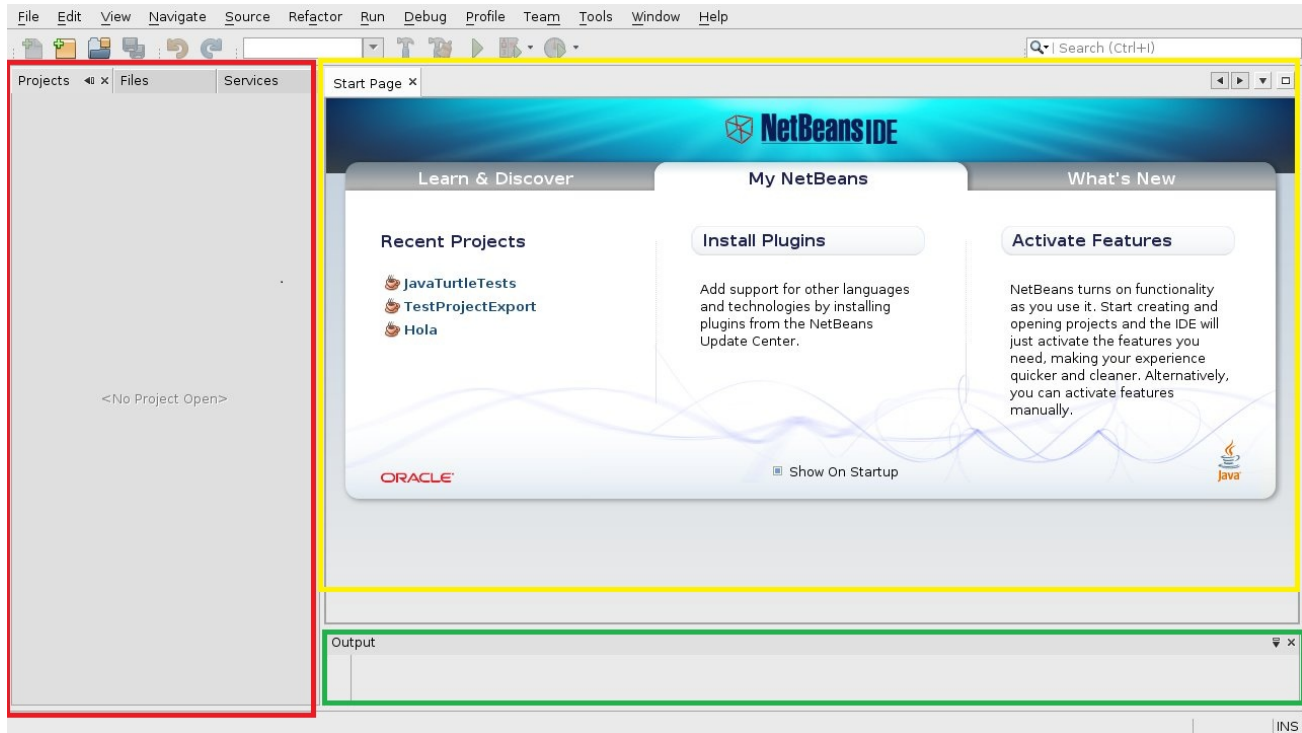
En esta imagen se muestra la sección donde se debe introducir el nombre del proyecto, también el directorio en donde se encuentra, y como se llamará el directorio para crear el proyecto. Solo necesitan escribir el nombre del proyecto, las secciones **Project Location** y **Project Folder** las pueden dejar como están. El cuadro rojo muestra dos opciones ha seleccionar. La opción de **Create Main Class** la seleccionamos si deseamos que se cree una clase que contenga el método main(), si queremos crear nosotros mismos la clase desmarcamos esta opción. La opción de **Set as Main Project** es para indicarle a NetBeans que este nuevo proyecto es el principal, esto nos ayuda, ya que por defecto todos los botones del NetBeans funcionan para el proyecto principal.



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

Enseguida aparecerá la ventana de bienvenida, como se muestra a continuación.



El recuadro amarillo indica la página de inicio, cuando abrimos o creamos una clase nueva en esta sección se colocará el editor. La sección en el recuadro rojo muestra los proyectos que tenemos, también en esta sección se muestran los archivos y servicios. En el recuadro verde se muestra la salida o ejecución de un programa en java.

En la siguiente figura se muestra el proyecto ImpresionConsola.

En el recuadro rojo se puede observar el proyecto. El proyecto esta dividido en varias secciones

4. PROCEDIMIENTO (DESCRIPCIÓN)

A) EQUIPO NECESARIO

MATERIAL DE APOYO

1. Computadoras con capacidad para ejecutar el entorno de desarrollo NetBeans.
2. Entorno de desarrollo NetBeans.
3. Proyecto para la introducción de NetBeans.



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

B) DESARROLLO DE LA PRÁCTICA

1. Cree un proyecto siguiendo los pasos indicados en la Introducción a NetBeans.
2. Modifique el método main() para que contenga el código que se muestra a continuación. Escriba el código, no lo copie. Recuerde que puede utilizar el "shortcut" *sout + Tab* para evitar escribir cada enunciado *System.out.println*.

```
public static void main(String[] args) {
    // TODO code application logic here
    System.out.println("Ejecución del primer programa de Java!");
    // operadores aritméticos
    System.out.println("Ejemplo del funcionamiento de operadores
aritméticos.");
    System.out.println("A) 3 + 4 = ");
    System.out.println("B) 25 * 32 = ");
    System.out.println("C) 17%3=");
    System.out.println("D) (3+4)/2=");
    System.out.println("E) 3+4/2=");
    System.out.println("F) 6.23-15.17=");
}
```

3. Describa el resultado obtenido.
4. Describa el resultado obtenido.
5. Modifique el código que escribió para que el programa también muestre los resultados de las operaciones indicadas.
6. Explique la diferencia entre la operación D y la E.
7. Agregue el siguiente código al método main().

```
int x,y,z;
```

```
double a,b,c;
```

```
x=3;
```

```
y=25;
```

```
z=18;
```

```
x+=z;
```

```
z/=x;
```

```
y%=7;
```

```
System.out.println("x=");
```

```
System.out.println("y=");
```

```
System.out.println("z=");
```

```
a=3.14;
```

```
b=70.5;
```



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

```
c=92.8;
```

```
c-=a;
b*=2;
a=b*c;
System.out.println("a=");
System.out.println("b=");
System.out.println("c=");
```

8. Qué valores tienen las variables a , b , c , x , y , z ?
9. Modifique el código para que se muestren también los valores de las variables y ejecute el programa.
10. Conciernen los valores del punto 8 con los que imprimió el programa?
11. Cierre el proyecto y cree un proyecto nuevo llamado Logicos que tenga un método `main()`.
12. Agregue el siguiente código al método `main()`

```
boolean i,j,k;
```

```
i=true;
j=false;
k=i&&j;
System.out.println("i && j = "+k);
k=i||j;
System.out.println("i || j = "+k);
```

13. Modifique el método `main()` para que también se imprima la siguiente línea utilizando el valor de la variable k en lugar de las palabras "true" y "false".
k vale true no vale false
14. Cierre el proyecto y abra el proyecto anterior.
15. Convierta en comentario todas las líneas que dicen `System.out.println` para que no pierda su trabajo.
16. Reproduzca la salida que obtuvo anteriormente de este programa pero ahora utilice `System.out.printf` para imprimir.

C) CÁLCULOS Y REPORTE

5. RESULTADOS Y CONCLUSIONES

6. ANEXOS



**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA
FACULTAD DE INGENIERÍA (UNIDAD MEXICALI)
DOCUMENTO DEL SISTEMA DE CALIDAD**

Formatos para prácticas de laboratorio

7. REFERENCIAS

Formatos de impresión para `System.out.printf`:

<http://download.oracle.com/javase/6/docs/api/java/util/Formatter.html#syntax>

Tipos de datos primitivos

http://docstore.mik.ua/oreilly/java-ent/jnut/ch02_04.htm