

# Estructuras de datos

## Clase práctica 1

---



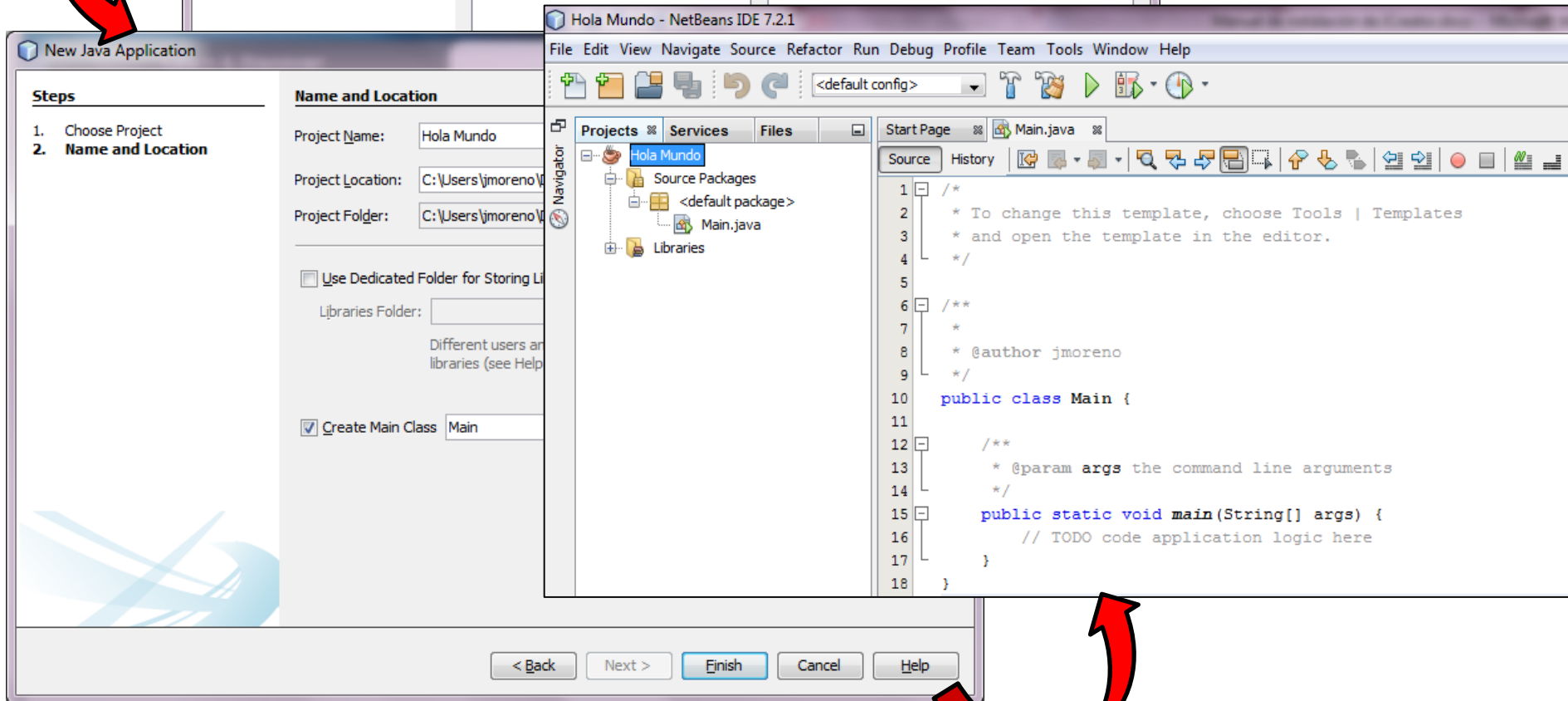
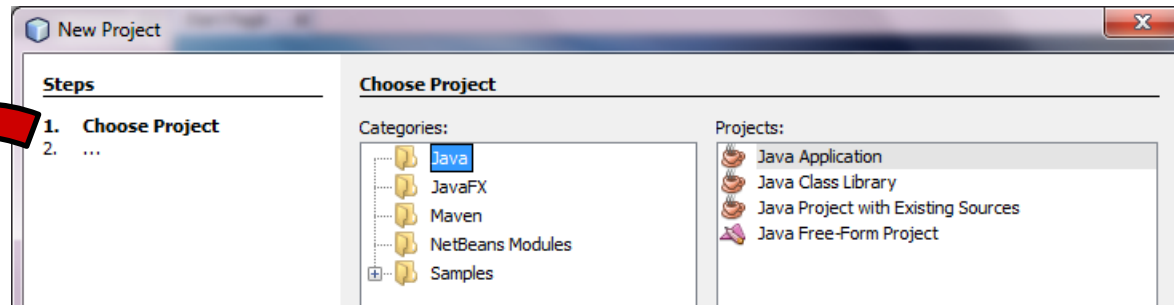
### Contenido

- Repaso de la estructura general de un programa en Java y de las operaciones de lectura y escritura
- Ejemplos de envío de ejercicios en la plataforma del curso

---

Material elaborado por: Julián Moreno

# Creación de un proyecto en NetBeans



# Estructura general de un programa en Java

```
import java.util.*;
```

Librería donde está Scanner (y muchas otras que usaremos en este curso)

```
public class Main {
```

```
    public static void main(String[] args) {  
        Scanner entrada = new Scanner(System.in);
```

```
        int a;
```

```
        float b;
```

```
        String c;
```

Entrada por consola, cada vez que se hace un next...el programa se queda "esperando" que se ingrese algo

¿Qué pasa si no se ingresa un *int*?

```
        a = entrada.nextInt();
```

```
        System.out.println("Ingresaste el entero " + a);
```

```
        b = entrada.nextFloat();
```

¿Cómo se ingresa un *float*?

```
        System.out.println("Ingresaste el real " + b);
```

```
        c = entrada.next();
```

¿Qué pasa con los espacios en blanco al ingresar un *string*?

```
        System.out.println("Ingresaste el texto " + c);
```

```
    }
```

```
}
```

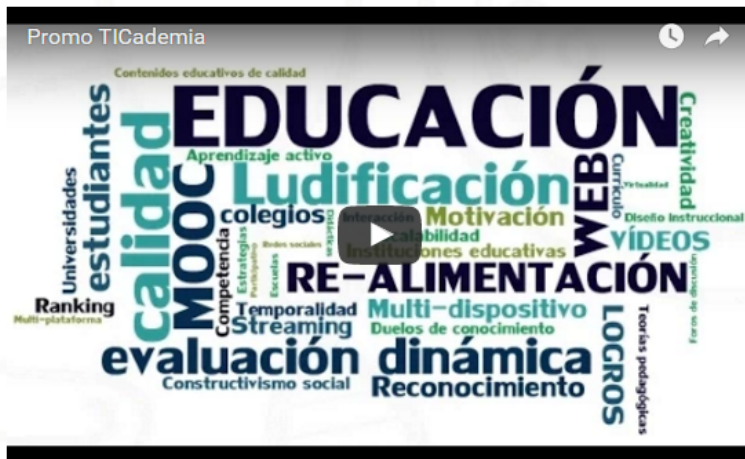
# Página del curso

<https://www.ticademia.com/>



Registrarse Entrar ▾

## Bienvenido(a)



Bienvenido(a), TICademia es un espacio para que aprendas de una manera desafiante. Encontrarás cursos de calidad con dos componentes fundamentales. Por una parte están los materiales de estudio, generalmente videos y documentos de apoyo, los cuales son realizados por docentes expertos en las áreas de conocimiento respectivas. Por otra parte están los ejercicios, los cuales no se limitan a simples preguntas de selección estáticas sino que en general están diseñados para medir verdaderamente tu aprendizaje.

Adicional a esta componente pedagógica, TICademia incluye dentro de sus didácticas un elemento clave: la ludificación. Esto no solo significa que recibirás logros a medida que avanzas e irás escalando niveles, sino también que podrás monitorear en tiempo real que tan “eficiente” eres en el curso, e incluso podrás batirte a duelo (en conocimientos obviamente) con tus compañeros para “arrebatarles” los puntos que te pondrán a la cabeza.

Por último TICademia también tiene una alta relación con redes sociales, no solo porque cuenta con su propio muro de notificaciones donde podrás compartir tu progreso, así como ver y opinar sobre el de tus compañeros de curso, si no también porque permite la conexión con redes populares como facebook y twitter.

Ya van 14591 estudiantes inscritos, ¿qué esperas para hacerlo?

[Ver cursos disponibles](#)

# Envío de ejercicios

Ahora si, manos a la obra y vamos a ver cómo es que funcionan los tales ejercicios, veamos el primero ...

## Encuentra el cuadrado

(tiempo límite: 0.5 segundos)

¿Qué hay que hacer?

Este ejercicio es bastante fácil en realidad: dado un número entero  $N$  en el rango  $[-200, 200]$ , mostrar su cuadrado.

### Entrada

La entrada contiene una única línea con el valor de  $N$ .

¿Qué variable(s) tiene la entrada? ¿De que tipo son?

### Salida

El cuadrado de  $N$ .

¿Cuál es la salida? ¿Cómo debe mostrarse?

### Ejemplo de entrada

7

¿Termina de aclarar lo que se había entendido?

### Ejemplo de salida

49

# Ejercicio 1: *Encuentra el cuadrado*

Alternativa 1:

```
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner entrada = new Scanner(System.in);
        short N;
        N = entrada.nextShort();
        System.out.println(N * N);
    }
}
```

Podría ser *int* o *long*,  
aunque se desperdiciaría  
memoria. No podría ser  
*byte*

En la plataforma del curso, y en  
general en competencias de  
programación, no se ponen  
mensajes de “Ingrese el valor  
de ...” sino que se lee  
directamente

# Ejercicio 1: *Encuentra el cuadrado*

Alternativa 2:

```
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner entrada = new Scanner(System.in);
        short N, R;
        N = entrada.nextShort();
        R = (short) (N * N);
        System.out.println(R);
    }
}
```

Algunos operadores y métodos “devuelven” tipos de datos específicos que pueden no ser compatibles con las operaciones que se están realizando. En este caso se requiere hacer un “cast”, es decir, una conversión. En general, las operaciones sobre enteros devuelven *int* y las operaciones sobre reales *double*

# Ejercicio 1: *Encuentra el cuadrado*

Alternativa 3:

```
import java.util.*;
public class Main {
    public static void main(String[] args) {
        Scanner entrada = new Scanner(System.in);
        int n;
        n = entrada.nextInt();
        System.out.println((int) Math.pow(n, 2));
    }
}
```

A veces las operaciones si son compatibles en cuanto a tipos pero no en cuanto al formato exigido. En este caso también se requiere hacer un “cast”

... Alternativa n



# ¿Qué devuelve como realimentación la plataforma?

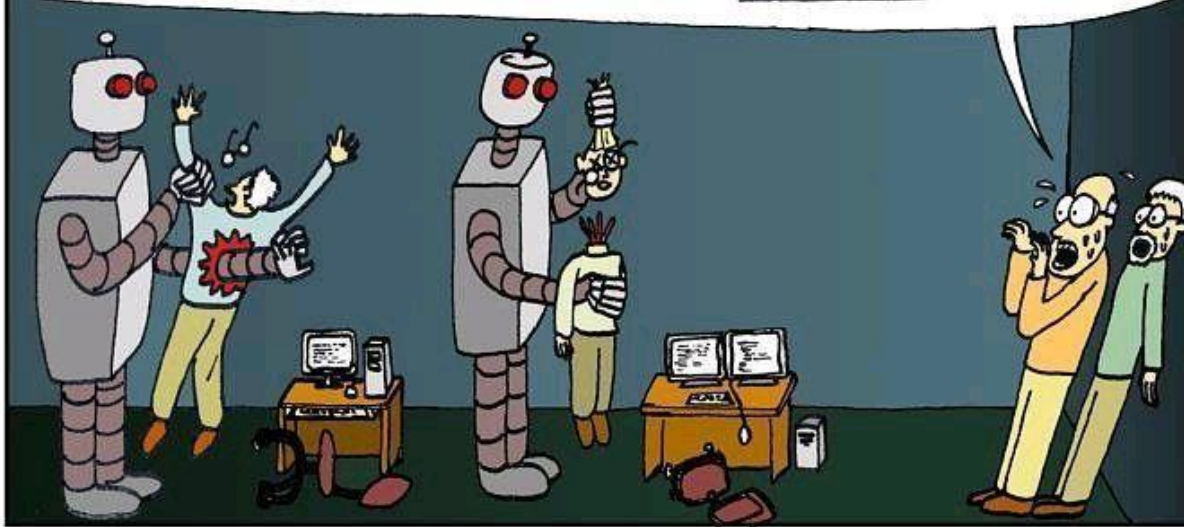
- Error de compilación: errores de sintaxis, inclusión de librerías no estándar, nombre incorrecto de la clase, uso de paquetes
- Error en tiempo de ejecución: el código compila, pero con los datos de los casos de prueba surgen problemas (acceso a un índice fuera de rango por ejemplo)
- Tiempo límite excedido: el código compila, ejecuta, pero se demora más de lo que debe
- Respuesta incorrecta: el código compila, ejecuta, termina antes del tiempo límite, pero la respuesta no corresponde o está mal formateada
- Aceptado:



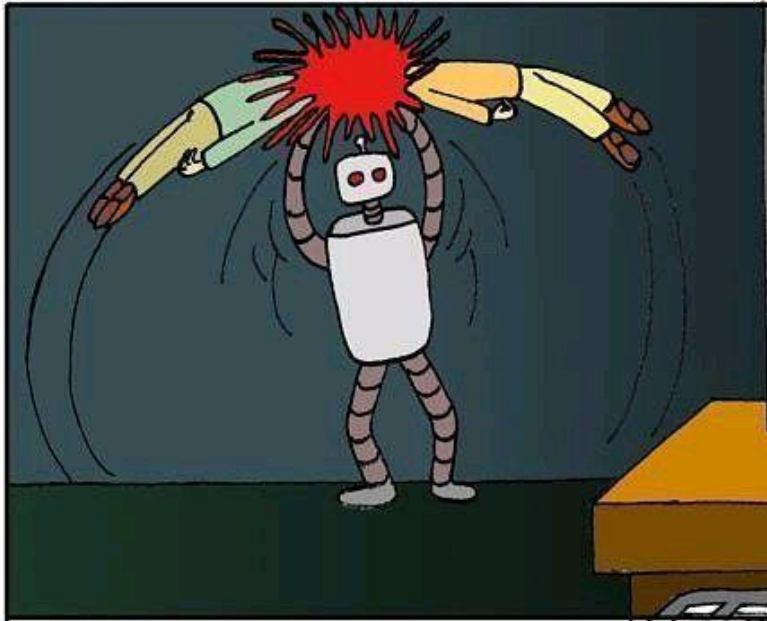
# Consideraciones importantes

- Independiente de la cantidad de datos que se vayan a leer y de si son o no del mismo tipo, se debe declarar UN SOLO `scanner`
- Cuando se hace `nextAlgo()` es indiferente si el dato que se va a leer está separado del anterior por un salto de renglón (ENTER), un espacio en blanco, o una tabulación
- En las salidas, los textos a mostrar y su formato (espacios en blanco, saltos de línea, etc.) deben ser exactos
- Recuerden no usar mensajes para operaciones de lectura (“Favor ingresar el valor de x”)
- No verificar los rangos, si en la formulación se dice es porque es así
- Netbeans no tiene separadas consolas de entrada y salida, por tanto no es necesario “acumular” las salidas para que salgan separadas

OH NO! THE ROBOTS ARE KILLING US!!!



BUT WHY?!? WE NEVER PROGRAMMED THEM TO DO THIS!!!



```
static bool isCrazyMurderingRobot = false;
```

```
void interact_with_humans (void){  
    if(isCrazyMurderingRobot = true)  
        kill(humans);  
    else  
        be_nice_to(humans);  
}
```

# Tarea

Realizar (analizar, diseñar, implementar y enviar) todos los ejercicios restantes del módulo de Calentamiento