

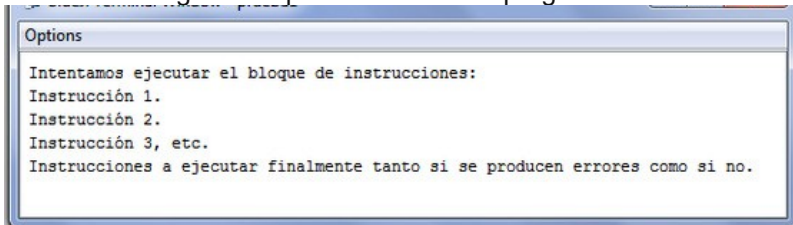
EXEMPLE TRY – CATCH – FINALLY

EXEMPLE SENSE ERRORS

A continuació anem a veure com es comporta un programa amb tractament d'errors però on no es produeix ninguna errada.

```
public class Programa {
    public static void main (String [] args)  {
        try{
            System.out.println("Intentamos ejecutar el bloque de instrucciones.");
            System.out.println("Instrucción 1.");
            System.out.println("Instrucción 2.");
            System.out.println("Instrucción 3, etc.");
        }
        catch (Exception e) { System.out.println("Instrucciones a ejecutar cuando se produce un error"); }
        finally{System.out.println("Instrucciones a ejecutar finalmente tanto si se producen errores como si no."); }
    }
}
```

La eixida obtinguda després d'executar el programa anterior es:



Com podem observar, s'han executat totes les instruccions del bloc **try** i finalment es va executar la instrucció del bloc **finally**.

No s'executa el bloc catch perquè no va haver-hi error.

EXEMPLE AMB ERRADES

A continuació veurem com es comporta un programa amb tractament d'errors quan es produeix un error i com afecta al control de flux del programa.

```
public class Programa {
    public static void main (String [] args)  {
        try {
            System.out.println("Intentamos ejecutar el bloque de instrucciones.");
            System.out.println("Instrucción 1.");
            int n = Integer.parseInt("M");
            System.out.println("Instrucción 2.");
            System.out.println("Instrucción 3, etc.");
        }
        catch (Exception e) {
            System.out.println("Instrucciones a ejecutar cuando se produce un error");
        }
        finally {
            System.out.println("Instrucciones a ejecutar finalmente tanto si se producen errores como si no.");
        }
    }
}
```

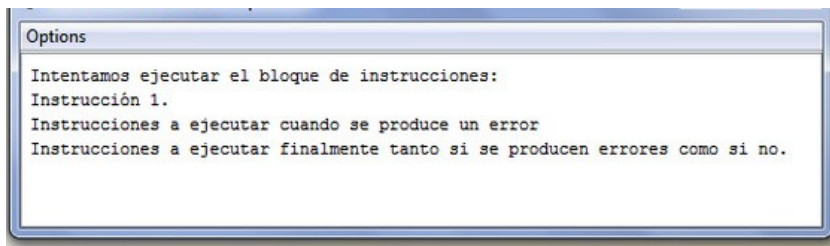
Es produeix un error perquè el mètode `parseInt` de la classe `Integer` espera que dins de les cometes arribi un número i no una lletra.

Per exemple `int n = Integer.parseInt("65");`

serveix per a transformar el `String` 65 en un `int` de valor 65.

Al no trobar un valor vàlid es produeix un error de tipus **`java.lang.NumberFormatException`**.

L'eixida obtinguda en aquest cas on es produeix el error és:



Com podem observar, executem les instruccions del bloc try que no donen errors, però quan en una instrucció es produeix un error o excepció inesperada es deixa d'executar el codi del bloc try, i passem a executar el codi del bloc catch.

Hi ha un salt o canvi en el flux del programa.

Finalment s'executen, en tot cas, les instruccions del bloc finally com hem comentat anteriorment.

El bloc finally no és obligatori, és a dir, pot existir un bloc try catch i no existir bloc finally.