

1. Indica la salida por pantalla que produciría el siguiente código:

```
import java.io.IOException;

public class Excepciones05 {
    public static int devuelveEntero(int num) throws Exception {
        try {
            if (num % 2 == 0) {
                throw new Exception("Lanzando excepcion rana");
            } else {
                throw new IOException("Lanzando excepcion pezespada");
            }
        } catch (IOException io) {
            System.out.println(io.getMessage());
            return 2;
        }
    }

    public static void main(String[] args) {
        int a = 99;
        for (int i = 0; i < 10; i++) {
            try {
                a = devuelveEntero(i);
            } catch (Exception e) {
                System.out.println(e.getMessage());
                a = 99;
            }
            System.out.println(a);
        }
    }
}
```

2. Escribe una clase de nombre **ClaseExcep1** que incluya un método denominado **dividirEntreArray**. Este método recibirá por parámetro un número entero y un array de elementos del mismo tipo. El método mostrará por pantalla el resultado de la división entre el número recibido y cada uno de los elementos del array. A continuación, se escribirá un programa que invoque al método con el número 2 y un array con los elementos -2, -1, 0, 1 y 2.

- Reescribe el método anterior para capturar la excepción derivada del intento de división entre 0, de forma que no se interrumpa la ejecución del programa y se continúe dividiendo entre los otros elementos del array.
- Reescribe el código anterior con las sentencias de código necesarias para que la excepción no llegue a producirse (comprobar que el divisor debe ser distinto de cero)

3. Escribe un método, de nombre **mostrarCadenasArray**, que reciba por parámetro un array con cadenas de caracteres. El método mostrará por pantalla el primer carácter de cada una de las cadenas contenidas en el array.
- Se debe evitar que se produzca una excepción **NullPointerException** si alguna de las posiciones del array contiene una referencia no inicializada (valor null) o una excepción **StringIndexOutOfBoundsException** si la cadena está vacía
 - Probar el método con los siguientes vectores:
 - .1. `String[] x = {"Hola", "Pepe", "Maria", "Miguel Angel"};`
 - .2. `String[] m = new String[5];`
 - .3. `String[] z = {"hola", null, "pepe", "", "quinta"};`
4. Escribe un método, de nombre **enviarMensaje**, que reciba por parámetro una cadena de caracteres correspondiente a una dirección de correo electrónico. El método comprobará que la dirección recibida es correcta lanzando la excepción **DirCorreoIncorrectaExcepcion** en caso contrario. La comprobación consistirá en verificar que la dirección contiene el carácter (@), algún carácter después de él antes del carácter (.) y algún carácter después de éste