

## Exercicis d'Excepcions

### Exercici 1.

Realitza un programa, que controle mitjançant una excepció pròpia (en una classe pròpia que herete de Exception), que el correu electrònic introduït per l'usuari siga correcte quant al format. Quan el format del correu no siga correcte, el programa generarà un error d'aqueixa excepció pròpia que serà capturat pel propi programa, i que farà que es torne a demanar de nou.

### Exercici 2.

**Crea un programa que faça servir aquestes classes:**

```
Class Tangente{
    Double angulo;
    Tangente (double a) { angulo = a; }
    Double resul() throws MiError{
        if (angulo == Math.PI/2.0)
            throw new MiError(1);
        if (angulo == 3.0*Math.PI/2.0)
            throw new MiError(2);
        return Math.tan(angulo);
    }
}
Class MiError extends Exception{
    int i;
    MiError (int i) { this.i = i;}
    String R(){
        if (i ==1){
            return "No existe tangente de 90 grados";}
        if (i ==2){
            return "No existe tangente de 270 grados";}
        return "error indefinido";
    }
}
```

### Exercici 3.

Crea una classe. Aquesta classe contindrà un mètode que acceptarà dos Strings, i realitzarà la divisió entre aquests.

Utilitza el mètode parseInt() de la classe Integer (static int parseInt(String s)), per a transformar de String a int.

Si els Strings no són números vàlids es genera un NumberFormatException.

Aquesta excepció ha de ser capturada i un missatge d'error ha de ser mostrat.

El segon número és usat per a dividir el primer número.

Si el segon número és zero, una excepció ArithmeticException és llançada.

Aquesta excepció ha de ser atrapada i un missatge d'error ha de ser mostrat.

### Exercici 4

Realitza la lectura de fitxers diferents. La ruta d'aquests serà sol·licitada per pantalla i escriurà el contingut de cadascun d'ells, en un nou i únic fitxer (aquest nou fitxer en finalitzar haurà de tindre el contingut de tots ells).

Utilitzarem un mètode per a la lectura i altre per a la escriptura del fitxer. Aquests mètodes seran invocat tantes vegades com siga necessari.

Realitza un control d'errors per les possibles fallades en la lectura i escriptura dels fitxers, però aquests errors seran llançats per a que siguin controlats al programa principal.

### Exercici 5:

Executa el següent programa i analitza que realitza la classe Runtime:

```
import java.io.IOException;
public class ejercicio {
    private static String direccion = "https://www.solvam.es/";

    public static void abrirNavegadorPredeterminadorWindows(String url) throws IOException{
        Runtime.getRuntime().exec("rundll32 url.dll,FileProtocolHandler " + url);
    }
    public static void abrirNavegadorPredeterminadorLinux(String url) throws IOException{
        Runtime.getRuntime().exec("xdg-open " + url);
    }
}
```

```

    public static void abrirNavegadorPredeterminadorMacOsx(String url) throws IOException{
        Runtime.getRuntime().exec("open " + url);
    }

    public static void abrirNavegadorPorDefecto(String url) throws IOException{
        String osName = System.getProperty("os.name");
        if(osName.contains("Windows"))
            abrirNavegadorPredeterminadorWindows(url);
        else if(osName.contains("Linux"))
            abrirNavegadorPredeterminadorLinux(url);
        else if(osName.contains("Mac OS X"))
            abrirNavegadorPredeterminadorMacOsx(url);
    }

    public static void main(String[] args){
        try {
            abrirNavegadorPorDefecto(direccion);
        } catch (IOException e) {
            System.out.println("Fallo al iniciar el navegador para abrir la url " + direccion);
            e.printStackTrace();
        }
    }
}

```

---

## EXERCICI 6

L'exercici amb dates:

Opció 1.- S'introdueix la data de naixement i el programa retorna l'edat de la persona en setmanes (la resposta ha de mostrar-se en nombre enter).

Opció 2.- S'introdueix una data i retorna el nom de l'estació en la qual aquesta situada aquesta data.

Opció 3.- S'introdueixen dues dates i retorna la quantitat de diumenges compresos entre totes dues dates.

***Nota: Ajutat de la llibreria `java.time`, `java.util.Date` i `java.util.Calendar`.***

## EXERCICI 7

Crea un programa per a un taller de cotxes.

Tota la informació generada es guardarà en un únic fitxer. El usuari decidirà on vol que es guardi aquest fitxer.

Menú:

1. Introduir cotxe per a reparació.
2. Indicar data de finalització de reparació.
3. Esborrat de cotxe reparat.

En l'opció 1, s'inserirà una matrícula i un data d'entrada.

En l'opció 2, es buscarà per matrícula, es llegirà la data d'entrada i s'indicarà que la data final de reparació serà 7 dies després.

En l'opció 3, s'esborrarà la línia del cotxe reparat.

Recorda utilitzar control de errors necessari.