



## CFGS - Desenvolupament d'aplicacions multiplataforma

### Mòdul 9 – Programació de serveis i processos

### UF2 – Processos i fils

Per l'activitat heu de lliurar un PDF amb el següent contingut:

- Nom o noms dels membres que fan l'activitat (màxim 2 alumnes).
- URL del repositori on hi ha els fonts (github, bitbucket,...).
- Noms d'usuari utilitzats per cada membre del grup del repositori del núvol.
- Explicacions, línies de codi creades/modificades i bolcats de pantalla de l'execució dels diferents punts que es demanen en la tasca.

El pdf s'ha d'anomenar seguint el format NomCognom1Cognom2.pdf del membre que fa el lliurament. Si un grup està format per dos membres NOMÉS UN l'ha de lliurar.

#### Activitat 2

A partir del següent codi subministrat, fes els canvis que calgui per a que:

- Crea un pool de 4 fils.

```
// Crea un pool de 4 fils
final ScheduledExecutorService schExService = Executors.newScheduledThreadPool(4);
```

- La primera vegada al cap de 5 segons i posteriorment cada 6 segons.

```
// Programa Fil, s'inicia als 5 segons i després es va executant cada 6 segons
schExService.scheduleWithFixedDelay(ob, 5, 6, TimeUnit.SECONDS);
```

- Cal deixar temps suficient per que s'executin 5 tasques.

```
// Espera per acabar 34 segons
schExService.awaitTermination(34, TimeUnit.SECONDS);
```

```
Inici: 18:18:6
pool-1-thread-1
Hora execució tasca: 18:18:11
Tasca en execució
Execució acabada
pool-1-thread-1
Hora execució tasca: 18:18:17
Tasca en execució
Execució acabada
pool-1-thread-2
Hora execució tasca: 18:18:23
Tasca en execució
Execució acabada
pool-1-thread-1
Hora execució tasca: 18:18:29
Tasca en execució
Execució acabada
pool-1-thread-3
Hora execució tasca: 18:18:35
Tasca en execució
Execució acabada
Completat
|
```

En mi caso hace falta 34 segundos para que ejecuten 5 tareas



Arxiu	Document extern
Elaborat	Cap d'estudis

Codi	MO-CAP012			1 de 1
Versió	6	Data	19/12/2018	



```
package uf2;

import java.util.Calendar;
import java.util.GregorianCalendar;
import java.util.concurrent.ExecutionException;
import java.util.concurrent.Executors;
import java.util.concurrent.ScheduledExecutorService;
import java.util.concurrent.TimeUnit;

public class activitat2 {

    public static void main(final String... args) throws InterruptedException,
    ExecutionException {

        //mostrem hora actual abans d'execució

        Calendar calendario = new GregorianCalendar();

        System.out.println("Inici: " + calendario.get(Calendar.HOUR_OF_DAY) + ":" +
        calendario.get(Calendar.MINUTE) +

            ":" + calendario.get(Calendar.SECOND));

        // Crea un pool de 2 fils
        final ScheduledExecutorService schExService = Executors.newScheduledThreadPool(2);
        // Crea objecte Runnable.
        final Runnable ob = new activitat2().new ExecutaFil();
        // Programa Fil, s'inicia als 2 segons i després es va executant cada 3 segons
        schExService.scheduleWithFixedDelay(ob, 2, 3, TimeUnit.SECONDS);
        // Espera per acabar 10 segons
        schExService.awaitTermination(10, TimeUnit.SECONDS);
        // shutdown .
        schExService.shutdownNow();
        System.out.println("Completat");
    }

    // Fil Runnable
    class ExecutaFil implements Runnable {

        @Override
        public void run() {

            Calendar calendario = new GregorianCalendar();

            System.out.println("Hora execució tasca: " +
                calendario.get(Calendar.HOUR_OF_DAY) + ":" +
                calendario.get(Calendar.MINUTE) + ":" +
                calendario.get(Calendar.SECOND));

            System.out.println("Tasca en execució");
            System.out.println("Execució acabada");

        }

    }

}
```



Arxiu	Document extern
Elaborat	Cap d'estudis

Codi	MO-CAP012			2 de 1
Versió	6	Data	19/12/2018	