



CIPFP Mislata
Centre Integrat Públic
Formació Professional Superior

Formador/a: j.puigcerveribanez@edu.gva.es

CFS/M: DAM

Mòdul DAM-ED

UD5

Grup 1S

2024-02-19: de 16:00 a 18:00

Contrasenya:

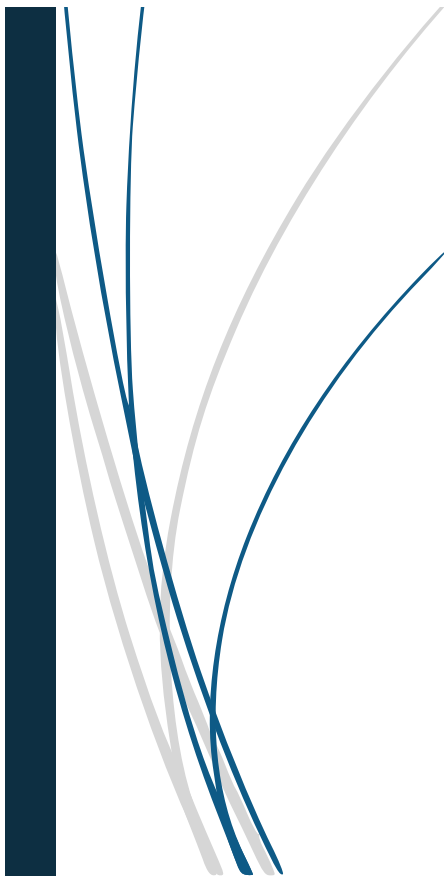
Submit

Development ONLY



CIPFP Mislata
Centre Integrat Públic
Formació Professional Superior

DAMS ED: 2a Avaluació



Autor: Joan Puigcerver Ibáñez

Correu electrònic: j.puigcerveribanez@edu.gva.es

Curs: 2023/2024

Llicència: BY-NC-SA



Entrega

Tots els exercicis han d'estar situats en el package corresponent.

- **Package: exam2**
- El títol de cada exercici ha de ser el nom de la Classe on comença l'execució del exercici.
- S'ha d'entregar a Aules/Moodle un fitxer **.zip** amb el contingut del package **exam2**
 - Renomenar el .zip perquè incloga el vostre nom.
- El codi ha d'estar pujat a GitHub en el vostre repositori de l'assignatura en el package corresponent. (*)
- Tag: **Exam2** ([StackOverflow: Create a tag in a GitHub repository](#)) (*)
- El codi ha de poder-se compliar.

(*) Aquesta part s'explicarà a partir de la Unitat Didàctica 2.

Si l'entrega no compleix aquests criteris, no és qualificarà.

Cal utilitzar el repositori creat en l'organització del mòdul: **CognomNom-ED**.

1. Exercici 1: Git Avançat (4 punts)

Es demana seguir els següents passos:

1. Crea un commit amb el missatge: **"Exam2.1.1: Creació ex1.md"**. on s'afegeix el fitxer **ex1.md** dins del package **exam2**.

```
# Exercici 1 - Nom Cognom
```



2. Crea les branques `exam2/branch1` i `exam2/branch2`.
3. Crea un commit a la branca `exam2/branch1` amb el missatge: **"Exam2.1.2: Canvi 1"**. on s'afegeix el següent contingut al fitxer **ex1.md**.

```
diff --git a/ex1.md b/ex1.md
index 98d41c3..cd34b45 100644
--- a/ex1.md
+++ b/ex1.md
@@ -1,2 @@
# Exercici 1 - Nom Cognom
+- Canvi 1
```

4. Crea l'etiqueta amb el nom **EX2-C1** al commit anterior.

5. Crea un commit a la branca **exam2/branch2** amb el missatge: "**Exam2.1.3: Canvi 2**". on s'afegeix el següent contingut al fitxer **ex1.md**.

```
diff --git a/ex1.md b/ex1.md
index 98d41c3..fa1e8e5 100644
--- a/ex1.md
+++ b/ex1.md
@@ -1,2 @@
# Exercici 1 - Nom Cognom
+- Canvi 2
```

6. Crea l'etiqueta **EX2-C2** al commit anterior.

7. A la branca **exam2/branch2** desfés el commit anterior sense modificar l'històric.

8. A la branca **exam2/branch2** aplica els canvis del commit etiquetat amb **EX2-C1**.

9. A la branca **exam2/branch2** aplica els canvis del commit etiquetat amb **EX2-C2**.

- Esborra les marques dels conflictes.

10. Fusiona la branca **exam2/branch2** a la branca **main** de manera que tots els canvis es fusionin en un sol commit. El missatge del commit ha de ser: "**Merge branch 'exam2/branch2' into main**".

1.1. Estat final

L'estat final del repositori ha de ser el següent:

```
* bf3da88 - (2 seconds ago) Merge branch 'exam2/branch2' into main - Joan Puigcerver (HEAD -> main)
| * e1aef31 - (2 minutes ago) Exam2.1.3: Canvi 2 - Joan Puigcerver (exam2/branch2)
| * 9017e50 - (2 minutes ago) Exam2.1.2: Canvi 1 - Joan Puigcerver
| * ea375dc - (2 minutes ago) Revert "Exam2.1.3: Canvi 2" - Joan Puigcerver
| * 180bbdb - (2 minutes ago) Exam2.1.3: Canvi 2 - Joan Puigcerver (tag: EX2-C2)
|/
| * 614e147 - (2 minutes ago) Exam2.1.2: Canvi 1 - Joan Puigcerver (exam2/branch1, tag: EX2-C1)
|/
* b420a17 - (2 minutes ago) Exam2.1.1: Creació ex1.md - Joan Puigcerver
```

1.2. Rúbrica

- **Total: 4 punts**

- (0.5 punts) Creació dels commits i les branques.
- (0.5 punts) Creació de les etiquetes.
- (1 punt) Desfer el commit a la branca **exam2/branch2** sense modificar l'històric.
- (0.5 punts) Aplicar els canvis del commit etiquetat amb **EX2-C1** a la branca **exam2/branch2**.
- (0.5 punts) Aplicar els canvis del commit etiquetat amb **EX2-C2** a la branca **exam2/branch2**.
- (1 punt) Fusionar la branca **exam2/branch2** a la branca **main** de manera que tots els canvis es fusionin en un sol commit.

2. Exercici 2: Lamp (6 punts)

- El codi ha d'estar en el package: **exam2**

Mitjançant l'estratègia TDD, implementa la classe `Lamp` que representa una làmpara que es pot encendre i apagar.

Consell

Recorda que el TDD consisteix en escriure els tests abans de programar la lògica de l'aplicació.

2.1. LampTest

Crea la classe `LampTest` per implementar les proves unitàries de la classe `Lamp`.

- Utilitza l'annotació `@Nested` per agrupar els tests.
- Utilitza l'annotació `@DisplayName` per donar un nom significatiu als tests i a les classes anidades.
- Utilitza l'annotacions necessàries per simplificar la codificació dels tests.
- Manteniu l'ordre de creació dels tests per veure el procés TDD seguit.
- Si has d'esborrar un test, no l'esborres, comenta'l.

2.2. Lamp

La classe `Lamp` representa una làmpara que es pot encendre i apagar.

Els objectes d'aquesta classe poden ser instanciats de dues maneres:

- Amb el constructor `Lamp(double consumption)`, que crea una làmpara apagada amb el consum indicat en kWh.
- Amb el constructor `Lamp(double consumption, boolean turnedOn)`, que crea una làmpara encesa o apagada segons el valor del paràmetre `turnedOn` amb el consum indicat en kWh.

La classe `Lamp` disposa dels següents mètodes:

- `isOn(): boolean`: retorna `true` si la làmpara està encesa, `false` en cas contrari.
 - (No cal provar aquest mètode)
- `getConsumption(): double`: retorna el consum de la làmpara en kWh.
 - (No cal provar aquest mètode)
- `turnOn()`: encén la làmpara.
- `turnOff()`: apaga la làmpara.
- `toggle()`: alterna l'estat de la làmpara; si està encesa l'apaga i si està apagada l'encén.
- `consume(double seconds): double`: retorna el consum de la làmpara en kW després de passar el temps indicat en segons.
 - Si la làmpara està apagada no consumeix res.
 - Aquest mètode s'ha de provar amb un test parametritzat.**

Consell

Els kilowatts consumits per una làmpara encesa es poden calcular com a:

$$kW = kWh \cdot \frac{seconds}{3600}$$

2.3. Calculadora de consum

Consumption (kWh) 0

Seconds 0

Calculate**Consumption (kW):** 0.00

2.4. Rúbrica

- **Total: 6 punts**
 - (1 punt) Prova i implementació dels constructors en una classe anomenada anidada.
 - (1 punt) Prova i implementació del mètode `turnOn` en una classe anomenada anidada.
 - (1 punt) Prova i implementació del mètode `turnOff` en una classe anomenada anidada.
 - (1 punt) Prova i implementació del mètode `toggle` en una classe anomenada anidada.
 - (2 punts) Prova i implementació del mètode `consume` amb un test parametritzat.