

ÍNDEX

ACTIVITAT 3.26 - Endevina el número.....	2
ACTIVITAT 3.27 - Contant matrícules.....	6

Mètode de treball per a totes les Activitats

- Cada activitat ha de ser resolta en el seu fitxer `ActivitatX.java` on X representa el número d'activitat, per exemple, per a la **Activitat 3.26** el nom del fitxer serà `Activitat26.java`). En cada fitxer has de tenir una classe amb el mateix nom, que resolga el problema plantejat.
- Totes les activitats estaran dins del projecte **PROG-UD3-CLASSWORK**. Per fer l'entrega has de pujar sols aquells fitxers que contenen el **codi font**, es a dir, **els fitxers** `xxxxxx.java` corresponents a **cada activitat**.
- **Has de lliurar només els fitxers java de cada activitat.**
- **No oblidis llegir amb especial atenció les normes concretes** de cada activitat i la rúbrica que encontraràs en la tasca corresponent d'aules.
- Recorda utilitzar constants i seguir estrictament els estàndards de codificació del llenguatge JAVA.
- No et quedes amb els valors dels exemples i prova el teu programa amb altres valors per a comprovar que el teu programa funciona per a TOTS ELS POSSIBLES CASOS tant vàlids com invàlids.
- **Es penalitzarà no complir amb les normes de codificació establides a classe** (nombrat de fitxers i classes, identificadors adequats de variables, respectar els indentats, creació de constants per les dades literals proposades als enunciats de cada problema i respectar l'espai entre instruccions (com a molt una línia en blanc).
- Recorda que el teu codi ha de passar les proves a la plataforma Repl.it.
- **MOLT IMPORTANT: La instrucció de creació de l'objecte `Scanner` ha de executar-se només una vegada en el teu codi. D'altra manera, les proves no passaran correctament i la pràctica serà suspesa.**

ACTIVITAT 3.26 - Endevina el número



Realitza un programa que ens permeta jugar amb l'ordinador a endevinar un número generat aleatoriament i què serà secret (en la secció de recursos del curs disposes de documentació sobre com generar números aleatoris). El **número ha de contindre com a màxim 2 dígits** (-99 - 99). Haurem de preguntar a l'usuari que introduísca números entre el **-99 i 99** amb la finalitat de endevinar-ho.

L'usuari disposarà d'un **número d'intents màxim** per a poder endevinar el número. Aquesta número **vindrà donat pel nivell de dificultat seleccionat** que se li **preguntarà a l'usuari abans de començar el joc**. El número d'intents ve donat per la següent taula:

Nivell	Número d'intents disponibles
fàcil	10 intents
intermedi	8 intents
difícil	5 intents

Si en cada intent, no s'endevina el número, l'ordinador ens donarà una pista per tal de saber si el número introduït s'acosta al número secret. Les possibles pistes seran les següents, i aquestes variaran en funció del nivell de dificultat:

- **Cremant:** el número introduït és un número que queda a menys de **10 unitats absolutes** (per dalt o per baix del número introduït). **[no es mostrarà en el nivell difícil; es mostrarà Calent en el seu lloc]**
- **Calent:** el número introduït és un número que queda a menys de **20 unitats absolutes**.
- **Templat:** el número introduït és un número que queda a menys de **30 unitats absolutes**. **[només disponible en nivell fàcil, en la resta de nivells la pista serà també Fret]**
- **Fret:** el número introduït queda a **30 o més unitats absolutes**.

Has de tindre en compte que has de mostrar els textos de les pistes en el color adequat (en el següent [enllaç](#) disposes de la documentació suficient per a poder mostrar textos en un color concret en la consola de Java).

Normes i exemples d'execució

- És obligatori utilitzar bucles `while` i/o `do-while` (utilitza aquell que supose més adequat en cada moment, es valorarà la tria de l'opció més indicada).
- Es valorarà la utilització adequada de estructures `if/else` en cada cas.
- En qualsevol moment, en cas que s'introduïska una dada que no corresponga amb el tipus de dades esperat, **el programa finalitzarà** mostrant el missatge d'error *"Error! El tipus de dades introduït és incorrecte"*.
- Si s'introdueix un nivell de dificultat no contemplat en qualsevol de les opcions indicades (fàcil, intermedi o difícil) es mostrarà el missatge d'error *"Error! Nivell incorrecte"*, i **es demanarà de nou la introducció del nivell de dificultat**. La selecció de nivell només pot ser introduïda en minúscules.
- En cas de introduir un número a endevinar fora de rang, el programa mostrarà el missatge d'error *"Número incorrecte. Introduïx-lo de nou"*, **demanant de nou la dada. No es contabilitzarà aquesta tirada com a intent**. Fixa't també en els exemples d'execució.
- Quan exportes el teu codi a la plataforma Repl.it, **hauràs de fixar el número secret al valor **-45** en lloc de generar-lo aleatòriament**.

Exemples d'execució (el número secret sempre es el – 45):

Exemple 1 (S'introdueix un nivell de dificultat erroni i després es gasten els intents en el nivell fàcil)

```
BENVINGUT AL JOC: ENDEVINA EL NÚMERO SECRET
-----
Selecciona el nivell [fàcil|intermedi|difícil]:-4
Error! Nivell incorrecte
Selecciona el nivell [fàcil|intermedi|difícil]:fàcil
Intent 1 [-99 - 99]: 50
Fret
Intent 2 [-99 - 99]: 9
Fret
Intent 3 [-99 - 99]: -10
Fret
Intent 4 [-99 - 99]: -20
Templat
Intent 5 [-99 - 99]: 48
Fret
Intent 6 [-99 - 99]: -30
Calent
Intent 7 [-99 - 99]: -40
Cremant
Intent 8 [-99 - 99]: -60
Calent
Intent 9 [-99 - 99]: -55
Calent
Intent 10 [-99 - 99]: -59

Ho senc. No l'has encertat
Fi
```

Exemple 2 (S'endevina el número en el nivell fàcil)

```
BENVINGUT AL JOC: ENDEVINA EL NÚMERO SECRET
-----
Selecciona el nivell [fàcil|intermedi|difícil]:fàcil
Intent 1 [-99 - 99]: -10
Fret
Intent 2 [-99 - 99]: -30
Calent
Intent 3 [-99 - 99]: -49
Cremant
Intent 4 [-99 - 99]: -56
Calent
Intent 5 [-99 - 99]: -45

Enhorabona! L'has encertat
Fi
```

Exemple 3 (S'endevina el número en el nivell fàcil, introduïnt una vegada un número incorrecte)

```
BENVINGUT AL JOC: ENDEVINA EL NÚMERO SECRET
-----
Selecciona el nivell [fàcil|intermedi|difícil]:fàcil
Intent 1 [-99 - 99]: 89
Fret
Intent 2 [-99 - 99]: 109
Número incorrecte. Introduïx-lo de nou
Intent 2 [-99 - 99]: -9
Fret
Intent 3 [-99 - 99]: -45

Enhorabona! L'has encertat
Fi
```

Exemple 4 (S'introdueix un tipus de dada no esperat)

```
BENVINGUT AL JOC: ENDEVINA EL NÚMERO SECRET
-----
Selecciona el nivell [fàcil|intermedi|difícil]:fàcil
Intent 1 [-99 - 99]: -19
Templat
Intent 2 [-99 - 99]: hola

Error! El tipus de dades introduït és incorrecte
```

Exemple 5 (S'endevina el número en nivell difícil)

```
BENVINGUT AL JOC: ENDEVINA EL NÚMERO SECRET
-----
Selecciona el nivell [fàcil|intermedi|difícil]:difícil
Intent 1 [-99 - 99]: -10
Fret
Intent 2 [-99 - 99]: -30
Calent
Intent 3 [-99 - 99]: -44
Calent
Intent 4 [-99 - 99]: -45

Enhorabona! L'has encertat
Fi
```

ACTIVITAT 3.27 - Contant matrícules



Alguna vegada t'has preguntat quantes matriculacions de cotxes s'han realitzat a Espanya just abans de la teua, la del teu pare o la de la teua amiga? Perquè ha arribat l'hora de respondre a eixa pregunta amb un programa que serà capaç d'obtindre eixa informació.

L'objectiu d'esta pràctica consisteix a obtindre tant **el nombre de cotxes que s'han matriculat just abans d'una matrícula donada** com mostrar el **llistat de totes i cadascuna d'eixes matrícules**. A més, ens proporcionarà la informació de **la flota a la qual pertany una matrícula concreta**, tenint en compte les primeres 15600 matrícules pertanyen a la flota 1, les 15600 a la flota 2, i així successivament.

Cal tindre en compte que el format de les matrícules a Espanya està format per 4 números seguit de 3 lletres, seguint les següents restriccions que es mostren en el següent article:

<https://theobjective.com/motor/2022-09-08/letras-prohibida-matriculas-dgt/>

Per tant, en executar-se el nostre programa se li preguntarà a l'usuari la matrícula de la qual es vol obtindre el nombre de cotxes matriculats amb anterioritat a ell i per a això se sol·licitaran les dades de la següent manera:

1. Es demanarà la introducció del número del cotxe (un nombre enter comprés entre 0 i 9999).
2. Posteriorment es demanarà la introducció de la primera lletra de la matrícula (qualsevol lletra majúscula que no siga vocal ni la ñ). No és necessari validar que l'usuari introdueix només un caràcter ja que en este cas ens quedarem només amb el primer. (per exemple, si s'introdueix BCD, la lletra obtinguda serà la B).
3. Ídem al pas 2, per a la segona lletra.
4. Ídem al pas 3, per a la tercera lletra.

Una vegada recopilats estes dades, el programa anirà mostrant totes i cadascuna de les matrícules i al seu costat, el número de flota a la qual pertany cadascuna,

però en lloc de mostrar esta informació l'una davall de l'altra, el que el programa ha de fer és substituir la matriculada mostrada per la seua matrícula posterior seguint un ordre (per a això, hauràs de fer ús del caràcter retorne de carro que es va veure en l'exemple [d'aquesta diapositiva](#) de la presentació de la unitat que ara estem treballant).

L'ordre d'increment de les matrícules serà començant pel número de matrícula i al arribar al final dels números (9999), s'incrementarien les lletres (començant per la lletra de la dreta). Per exemple, aquesta seria la seqüència des de la primera matrícula 0000 BBB fins la ultima possible:

```
0000 BBB, 0001 BBB, ..., 9999 BBB, 0000 BBC, 0001 BBC, ...,  
9999 BBC, 0000 BBD, ..., 9999 BBZ, 0000 BCA, ..., 9999 YZZ,  
0000 ZZZ, ..., 9999 ZZZ
```

No has d'oblidar mostrar al costat de la matrícula el número de flota al qual pertanyerà eixa matrícula així com mostrar al final del procés quants cotxes s'han matriculat prèviament a la matrícula introduïda.

Normes i exemples d'execució

1. És obligatori la utilització d'estructures `for` anidades per a realitzar la implementació de la solució al problema. Tot i això, ja que el més probable és que necessites recolzar-te, estructures `while/do-while` pots fer servir-les també.
2. El text de sol·licitud de número i lletra es mostrarà amb **un tabulador** en el seu començament.
3. En qualsevol moment, en cas que introduïska una dada que no correspon amb el tipus de dades esperat, el programa finalitzarà mostrant el missatge d'error "Error! El tipus de dades introduït és incorrecte" i s'acabarà l'execució del programa
4. Si s'intenta introduir un nombre de matrícula fora de rang o si s'introdueix una lletra no permesa es mostrarà el missatge "Matrícula no vàlida" justament després de la introducció de la dada errònea i es tornarà a preguntar a l'usuari totes les dades de nou.

Al [següent video](#) pots veure un **exemple d'execució** del programa.

Es mostren a continuació exemples d'execució (final) en altres situacions:

Exemples d'execució

Exemple 1 (*Introducció d'una dada de tipus incorrecte*)

```
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: hola  
Error! El tipus de dades introduït és incorrecte
```

Exemple 2 (*Introducció d'un nombre incorrecte*)

```
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: 99999  
Matrícula no vàlida  
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: ...
```

Exemple 3 (*Introducció d'una lletra no permesa*)


```
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: 1234  
-> Introdueix primera letra: Ñ  
Matrícula no vàlida  
Introdueix matrícula  
-> Introdueix número: ...
```

Exemple 4 (*Introducció d'una lletra minúscula*)

```
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: 1234  
-> Introdueix primera letra: B  
-> Introdueix segona letra: b  
Matrícula no valida  
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: ...
```

Exemple 5 (*Introducció d'una matrícula correcta*)

```
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: 0  
-> Introdueix primera letra: B  
-> Introdueix segona letra: B  
-> Introdueix tercera letra: B  
0000 BBB(Flota: 1)  
Cotxes anteriors a 0000 BBB: 0
```

Exemple 6 (*Introducció d'altra matrícula correcta*)

```
--INTRODUCCIÓ DE MATRÍCULA--  
-> Introdueix número: 1234  
-> Introdueix primera letra: C  
-> Introdueix segona letra: C  
-> Introdueix tercera letra: C  
1234 CCC(Flota: 270)  
Cotxes anteriors a 1234 CCC: 4211234
```

NOTES D'AJUDA

- Per a convertir un String a char, es necessari que utilitzes la següent instrucció: `char lletra = lletraString.charAt(0);` on `lletraString` es un String que pot estar compostat per més d'un caràcter, d'aquesta manera, `lletra` contindrà només la primera lletra del String.
- Has de tindre en compte que els caràcters permeten les mateixes operacions que un número enter, és a dir, un caràcter es pot sumar, restar, etc.
- **Per poder donar la sensació de que els comptador s'actualitzen**, has de tindre en compte que si imprimeixes per pantalla el caràcter "`\r`" i **no s'ha fet cap salt de línia**, el cursor **tornarà al principi de la línia**, per la qual cosa, qualsevol cosa que imprimeixes sobre-escriurà l'anterior.
- Pots parar l'execució d'un programa durant els milisegons que necessites fent ús de la sentència `Thread.sleep(milisegons)`. D'aquesta forma donarà la sensació que la actualització de les matrícules és més lenta.
- **El següent snippet de codi** mostra un comptador que va **des de 0 fins al 100** en una **única línia** i ficant sempre 3 digits, completant amb zeros per l'esquerra quan siga necessari. PROVA'L I TRACTA DE COMPRENDRE EL CODI ABANS D'IMPLEMENTAR EL PROGRAMA.

```
for (int i = 0; i <= 100 ; i++) {  
    System.out.printf("%03d", i);  
    System.out.print("\r");  
    try {  
        Thread.sleep(10);  
    } catch (InterruptedException e) {  
        throw new RuntimeException(e);  
    }  
}
```