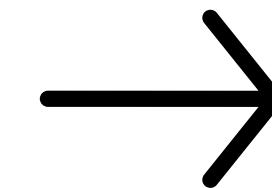


AEA1

Programació bàsica, condicionals i estructures de repetició



Prat

DAM

AEA1

Nom de l'AEA: Programació bàsica, condicionals i estructures de repetició	Hores totals AEA: 39 hores				
Descripció: L'alumnat adquirirà els fonaments de la programació, aprenent a utilitzar estructures bàsiques com les variables, operadors i funcions.					
Resultat d'Aprendentatge					
RA1: A1: Reconeix l'estructura d'un programa informàtic, identificant i relacionant els elements propis del llenguatge de programació utilitzat. RA2: Escriu i prova programes senzills, reconeixent i aplicant els fonaments de la programació orientada a objectes.					
Organitzadors previs: -					
Seqüència de l'activitat d'ensenyament-aprenentatge	Organització aula	Temporització	Metodologia	Instruments d'avaluació	Localització
AA1 - Multiplicació de 3 nombres reals 5%	Individual	4	ABP	Rúbrica	Aula
AA2 - Descompte 95 5%	Individual	4	Aprenentatge cooperatiu	Rúbrica	Aula
AA3 - Descompte 100 10%	Individual	4	ABP	Rúbrica	Aula
AA4 - Subscripció de videojoc 10%	Individual	4	ABP	Rúbrica	Aula

AEA1

Nom de l'AEA: Programació bàsica, condicionals i estructures de repetició			Hores totals AEA: 39 hores		
Seqüència de l'activitat d'ensenyament-aprenentatge	Organització aula	Temporització	Metodologia	Instruments d'avaluació	Localització
AA5 - Càcul d'àrees 10%	Individual	4	ABP	Rúbrica	Aula
AA6 - Sumar múltiples de tres 10%	Individual	4	Aprenentatge cooperatiu	Rúbrica	Aula
AA7 - Taula multiplicar enrere 15%	Individual	4	Aprenentatge cooperatiu	Rúbrica	Aula
AA8 - Endevina semàfor 15%	Individual	4	ABP	Rúbrica	Aula
AA9 - Menú interactiu 5%	Individual	3	ABP	Rúbrica	Aula
AA10 - Taula multiplicar v3 15%	Individual	4	Aprenentatge cooperatiu	Rúbrica	Aula

Programació

Introducció

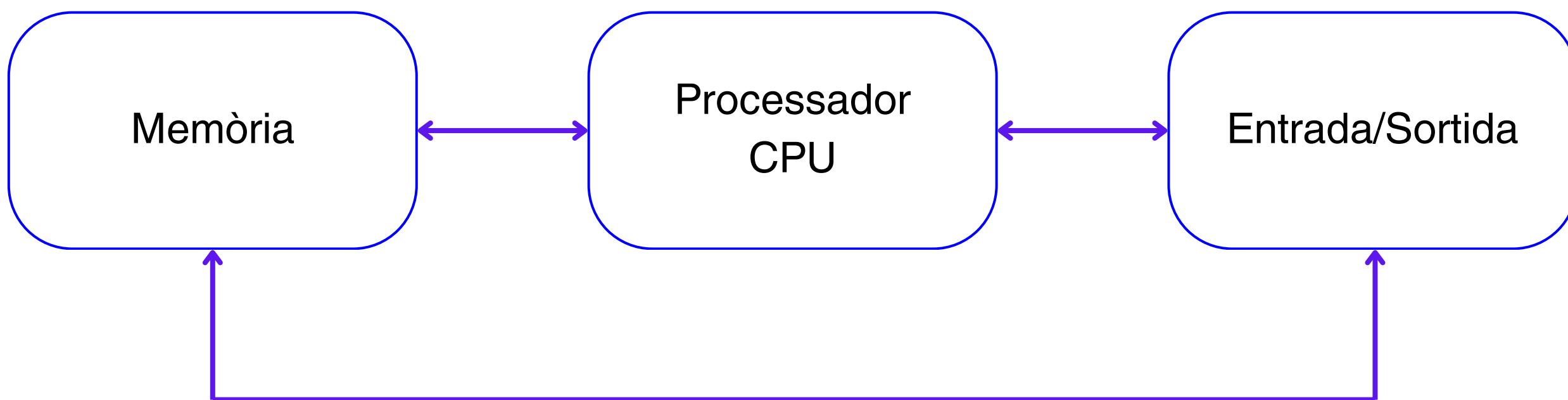
Un ordinador és incapaç de fer absolutament res per si mateix, sempre cal dir-li què ha de fer.

Això se li diu mitjançant l'execució de programes.

Introducció

L'estructura d'un ordinador bàsica es divideix en 3 components connectats entre ells.

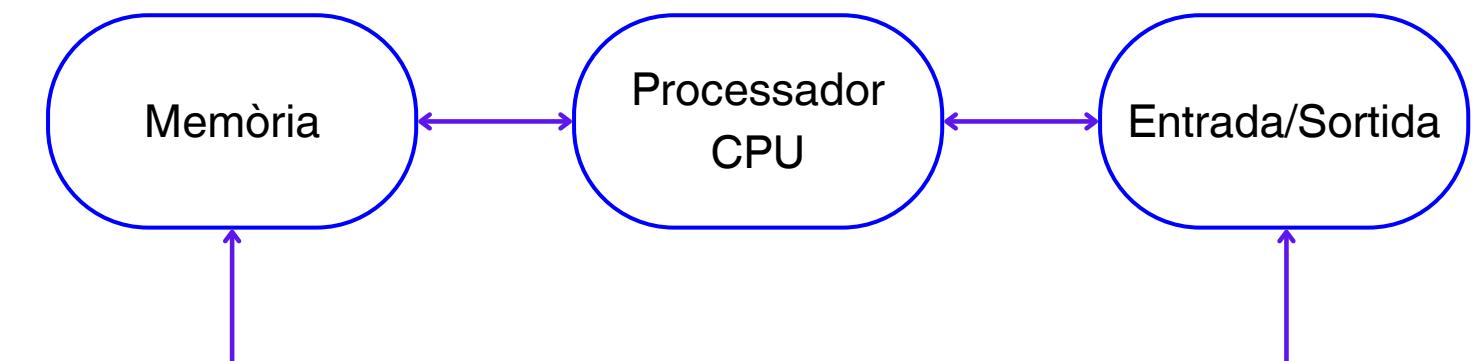
Què sabem de cadascun d'ells?



Introducció

Exemple

Anem a suposar que volem que el nostre pc multipliqui dos nombres.



Ordre per donar	Element que l'efectua
1. Llegeix un nombre del teclat.	E/S (teclat)
2. Desa el nombre a la memòria.	Memòria
3. Llegeix un altre nombre del teclat.	E/S (teclat)
4. Desa el nombre a la memòria.	Memòria
5. Recupera els nombres de la memòria i fes-ne la multiplicació.	Processador
6. Desa el resultat a la memòria.	Memòria
7. Mostra el resultat a la pantalla.	E/S (pantalla)

Introducció

Nivell alt

Programem amb dades textuais (de forma que nosaltres ho puguem escriure!)

```
import java.util.Scanner;  
  
public class HelloWorld {  
  
    public static void main(String[] args) {  
  
        // Creates a reader instance which takes  
        // input from standard input - keyboard  
        Scanner reader = new Scanner(System.in);  
        System.out.print("Enter a number: ");  
  
        // nextInt() reads the next integer from the keyboard  
        int number = reader.nextInt();  
  
        // println() prints the following line to the output screen  
        System.out.println("You entered: " + number);  
    }  
}
```

L'usuari entra un número i l'imprimim per la pantalla

Nivell baix

Llenguatge assemblador/màquina (no programem amb aquest llenguatge!)

```
# -----  
# Writes "Hello, World" to the console  
# To assemble and run:  
#  
#     gcc -c hello.s && ld hello.o && ./a.out  
#  
# or  
#  
#     gcc -nostdlib hello.s && ./a.out  
# -----  
  
.global _start  
  
.text  
  
_start:  
    # write(1, message, 13)  
    mov    $1, %rax           # system call 1 is write  
    mov    $1, %rdi           # file handle 1 is stdout  
    mov    $message, %rsi      # address of string to output  

```

Escrivim “Hello world!” per pantalla

Crear un programa executable

Com crear un programa executable? **⚠ Ho hem de fer en aquest ordre SEMPRE!**

1. Codi font

Arxiu **.java** on nosaltres (com a programadorxs) escriurem totes les ordres que volem que segueixi el PC quan executi el programa.

⚠ .java és l'extensió dels arxius que generarem

2. Compilació

Traducció del codi font a codi binari, per a que el PC pugui llegir el programa.

Compilem a través de d'un programa que es diu "compilador"

3. Codi objecte i programa executable

Un cop compilat el programa de forma correcta, el PC ens genera dos arxius:

- **Un codi objecte** on està la traducció del nostre programa a codi binari.
- Un arxiu amb extensió **.exe** que és el que ens deixarà veure el programa.

Crear un programa executable

Com crear un programa executable? **⚠ Ho hem de fer en aquest ordre SEMPRE!**



3. Codi objecte i programa executable

Un cop compilat el programa de forma correcta, el PC ens genera dos arxius:

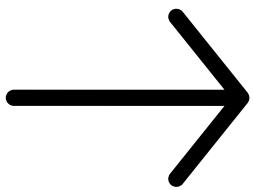
- **Un codi objecte** on està la traducció del nostre programa a codi binari.
- Un arxiu amb extensió **.exe** que és el que ens deixarà veure el programa.

JAVA

En aquest mòdul nosaltres ens centrarem en programar en JAVA.

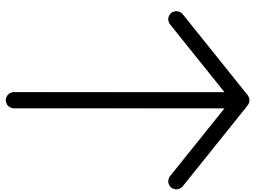
Us deixo links interessants per començar a introduir-nos a JAVA.

1. [Intro a JAVA](#)
2. [Tutorial per a començar de 0](#)
3. [Tutorial de JAVA bàsics](#)



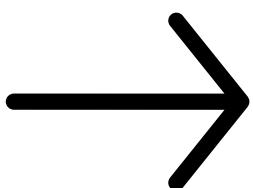
PROCÈS ITERATIU

És necessari que fem moltes proves per tal d'aconseguir entendre tot el que programem.



LLENGUATGE

Un cop entenem la dinàmica de la programació, ens serà molt fàcil canviar de llenguatge.



BASELINE

JAVA és la base de la majoria de llenguatges i frameworks actualment, així que 💪

La terminal

La terminal és un programa que permet comunicar-te amb l'ordinador escrivint ordres de text.

No té botons ni finestres visuals com altres programes; tot funciona amb línies de comandes.

En Java, la terminal és molt útil perquè és on:

- Compiles els teus programes (`javac NomPrograma.java`)
- Els executes (`java NomPrograma`)
- Veus els resultats i missatges del teu programa

La terminal

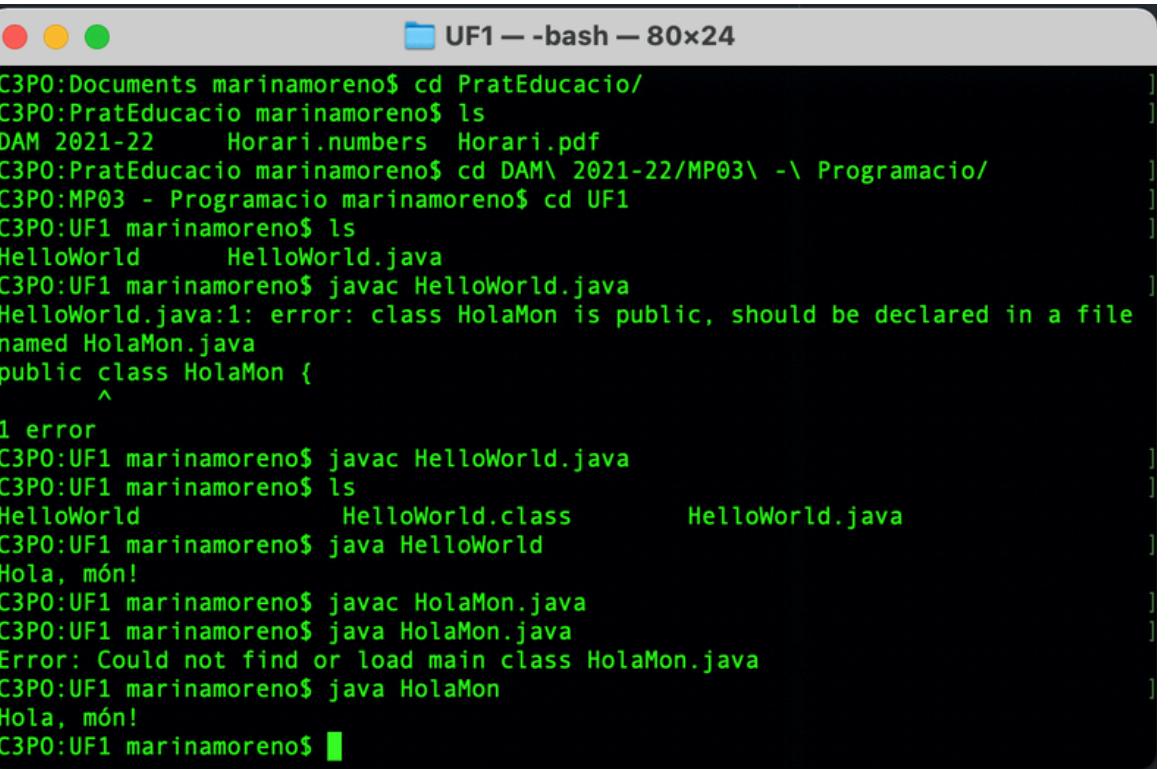
Per què la fem servir en Java?

Quan aprenes Java, sovint tens un programa com aquest:

```
public class HolaMundo {  
    public static void main(String[] args) {  
        System.out.println("Hola, món!");  
    }  
}
```

Per veure què fa el programa, has de:

- 1 Obrir la terminal
- 2 Compilar el programa
- 3 Executar el programa



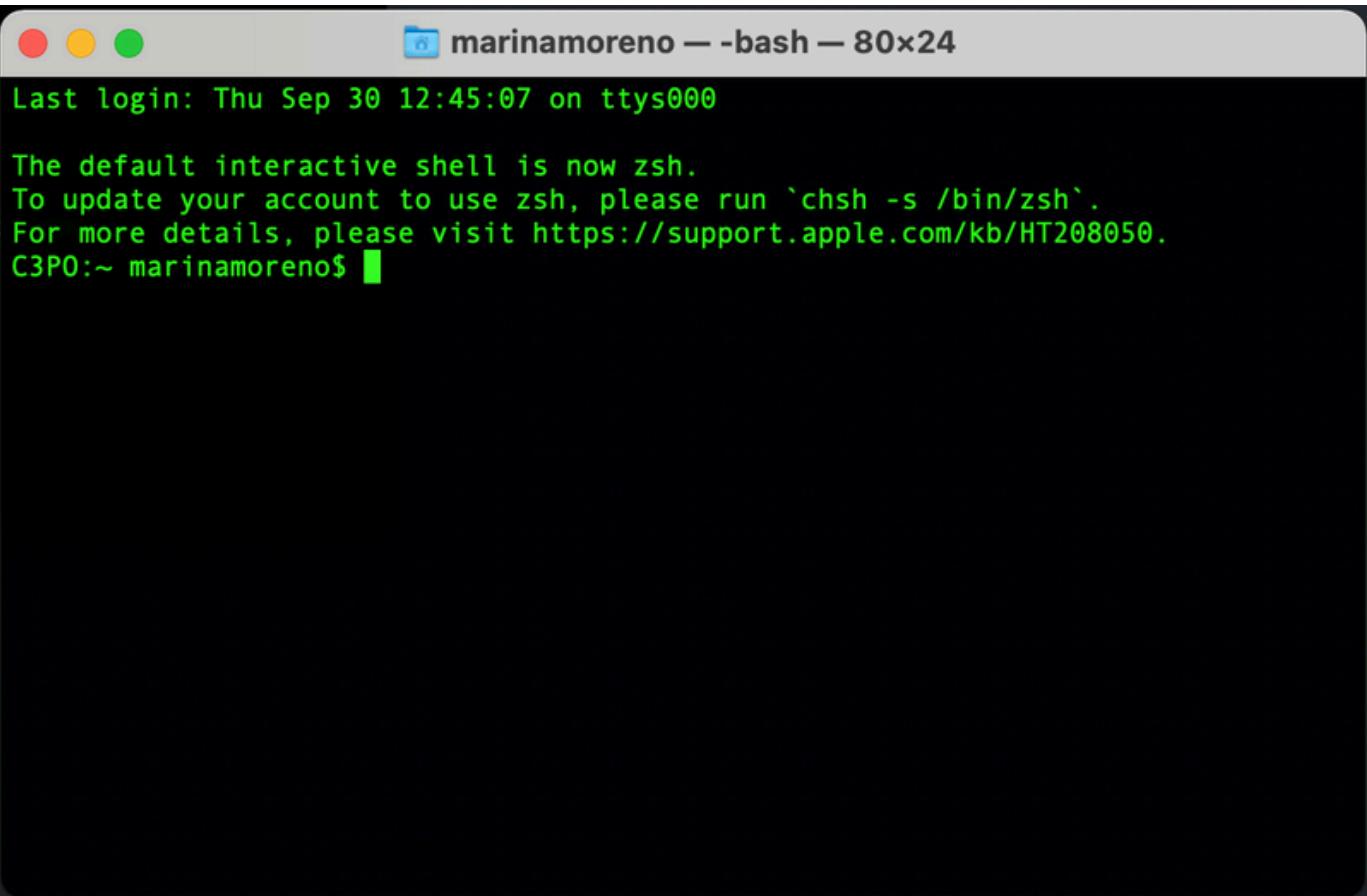
```
C3PO:Documents marinamoreno$ cd PratEducacio/  
C3PO:PratEducacio marinamoreno$ ls  
DAM 2021-22 Horari.numbers Horari.pdf  
C3PO:PratEducacio marinamoreno$ cd DAM\ 2021-22\MP03\ -\ Programacio/  
C3PO:MP03 - Programacio marinamoreno$ cd UF1  
C3PO:UF1 marinamoreno$ ls  
HelloWorld HelloWorld.java  
C3PO:UF1 marinamoreno$ javac HelloWorld.java  
HelloWorld.java:1: error: class HolaMon is public, should be declared in a file  
named HolaMon.java  
public class HolaMon {  
    ^  
1 error  
C3PO:UF1 marinamoreno$ javac HelloWorld.java  
C3PO:UF1 marinamoreno$ ls  
HelloWorld HelloWorld.class HelloWorld.java  
C3PO:UF1 marinamoreno$ java HelloWorld  
Hola, món!  
C3PO:UF1 marinamoreno$ javac HolaMon.java  
C3PO:UF1 marinamoreno$ java HolaMon.java  
Error: Could not find or load main class HolaMon.java  
C3PO:UF1 marinamoreno$ java HolaMon  
Hola, món!  
C3PO:UF1 marinamoreno$
```

La terminal

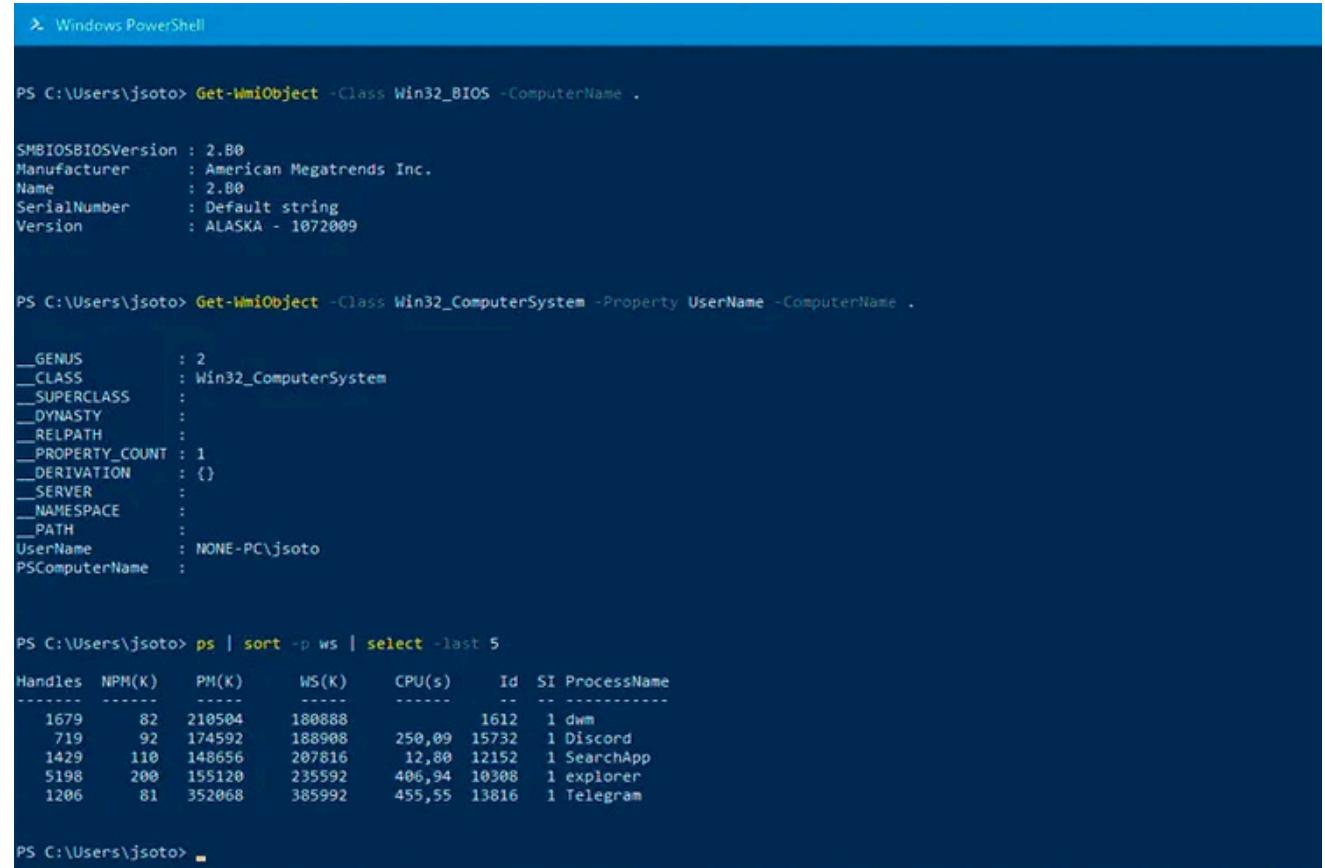
Exercici de classe - La terminal

Obrim la terminal

- MAC: busquem “terminal”
- WINDOWS: busquem “power shell”



```
Last login: Thu Sep 30 12:45:07 on ttys000
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
C3PO:~ marinamoreno$
```



```
PS C:\Users\jsoto> Get-WmiObject -Class Win32_BIOS -ComputerName .

SMBIOSBIOSVersion : 2.80
Manufacturer       : American Megatrends Inc.
Name               : 2.80
SerialNumber      : Default string
Version           : ALASKA - 1072009

PS C:\Users\jsoto> Get-WmiObject -Class Win32_ComputerSystem -Property UserName -ComputerName .

__GENUS          : 2
__CLASS          : Win32_ComputerSystem
__SUPERCLASS     :
__DYNASTY        :
__RELPATH        :
__PROPERTY_COUNT : 1
__DERIVATION     : {}
__SERVER         :
__NAMESPACE      :
__PATH           :
UserName         : NONE-PC\jsoto
PSComputerName   :

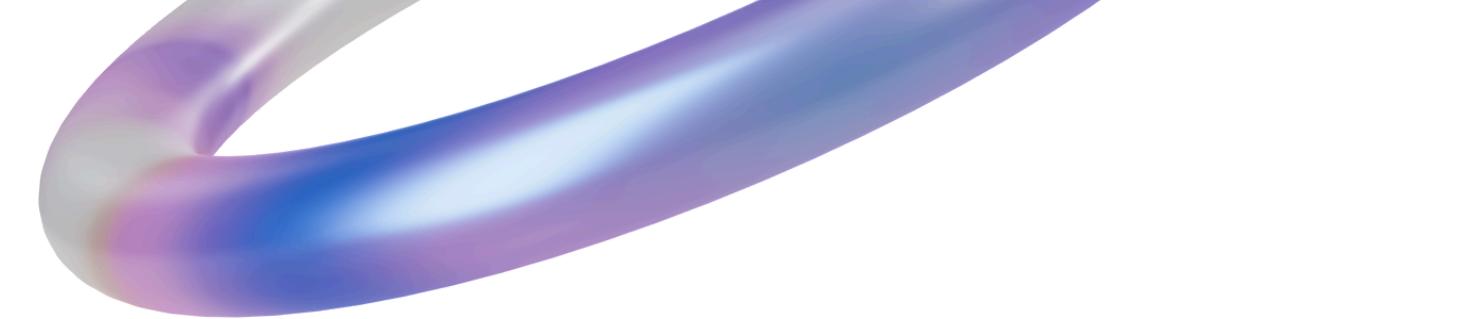
PS C:\Users\jsoto> ps | sort -p ws | select -last 5
Handles  NPM(K)    PM(K)    WS(K)    CPU(s)   Id  SI ProcessName
-----  -----    -----    -----    -----  --  --
1679     82    210504   180888    1612    1  dwm
719      92    174592   188908   250,09  15732   1 Discord
1429     110   148656   207816   12,88  12152   1 SearchApp
5198     200   155120   235592   406,94  10308   1 explorer
1206     81    352068   385992   455,55  13816   1 Telegram

PS C:\Users\jsoto>
```



Let's
play

La terminal - Comandaments



Per navegar per la terminal necessitem conèixer almenys els més bàsics

Accedir a carpetes

De la mateixa manera que anem fent clicks a les carpetes per entrar a elles, per fer-ho desde terminal haurem de fer....

cd *NomCarpeta*

```
C3PO:~ marinamoreno$ cd Applications/
```

Veure el que tenim en aquella carpeta del PC

Quan hem fet click en una carpeta, podem veure quines arxius tenim. Amb la terminal podrem veure'ls també si executem....

ls (és una L minúscula)

```
C3PO:~ marinamoreno$ ls
Applications          Downloads        Pictures
Creative Cloud Files Library        Public
Desktop              Movies          Sites
Documents            Music
```

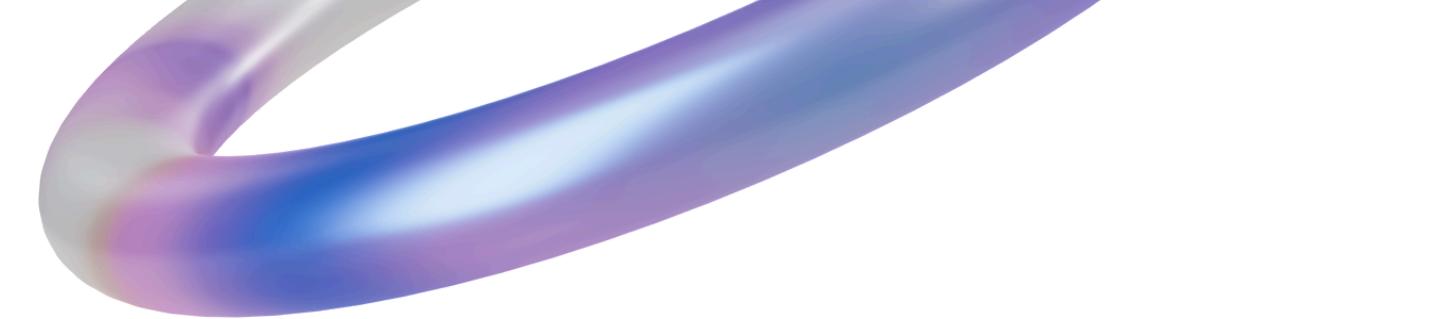
Crear carpeta

No ens caldrà fer botó dret o click amb dos dits per crear-ne a terminal, només haurem descriure...

mkdir *NomCarpeta*

```
C3PO:PratEducacio marinamoreno$ mkdir Test
C3PO:PratEducacio marinamoreno$ ls
DAM 2021-22      Horari.numbers   Horari.pdf      Test
C3PO:PratEducacio marinamoreno$
```

La terminal - Comandaments



Per navegar per la terminal necessitem conèixer almenys els més bàsics

Eliminar a carpetes

De la mateixa manera que anem fent clicks a les carpetes per eliminar-ne, per fer-ho desde terminal haurem de fer....

rmdir NomCarpeta

```
pc-252:Documents marinamoreno$ rmdir test
```

Fer “back” entre carpetes

Quan hem fet click en una carpeta i volem tornar enrere, mitjançant la terminal podrem fer-ho si executem....

cd ..

```
pc-252:Documents marinamoreno$ cd ..
pc-252:~ marinamoreno$
```

Tornar al path root inicial quan obrim la terminal

No ens caldrà anar una en una, amb un comandament a terminal podem fer-ho.

cd

```
pc-252:~ marinamoreno$ cd Documents/PratEducacio/
pc-252:PratEducacio marinamoreno$ cd ..
pc-252:~ marinamoreno$
```

Creació i execució d'un programa JAVA

Anem a començar a programar en JAVA 🎉. Què necessitem?

1. Un editor de text simple

Exemples:

Sublime text
Notepad
TextEdit (Apple)

⚠ Recordem que haurem de guardar el fitxer amb l'extensió **.java**

Descarregant-ho una vegada, ens descarreguem els compilador i l'intèpret.

2. Compilador JAVA

Java



3. Un intèpret de JAVA

Java ME Embedded
Java ME Embedded Client
Java ME SDK
Java Runtime Environment (JRE)
Java SE
Java SE Embedded
Java TV

Cal descarregar-lo a:
<http://www.oracle.com/downloads>

Cal descarregar-lo a:
<http://www.oracle.com/downloads>

Primer programa en JAVA



El primer programa de tot programador és: “Hola, món!”

1. Creem la carpeta de 0485

Anem a documents i creem la carpeta de “0485-Programacio”

⚠ Sense espais

⚠ Sense accents

2. Obrim l'editor de text i enganxem això

```
public class HolaMon {  
    public static void main(String[] args) {  
        System.out.println("Hola, món!");  
    }  
}
```

3. Guardem el fitxer

⚠ Anomenem el fitxer
“HolaMon.java”

⚠ El guardem a la carpeta amb
path “Documents/0485-
Programacio/AEA1/HolaMon”

Primer programa en JAVA



El primer programa de tot programador és: “Hola, món!”

4. Obrim la terminal

Se sol anomenar: consola, terminal,... hem de localitzar-la dins el nostre PC.

```
UF1 -- bash -- 80x24
[C3PO:Documents marinamoreno$ cd PratEducacio/
[C3PO:PratEducacio marinamoreno$ ls
DAM 2021-22 Horari.numbers Horari.pdf
[C3PO:PratEducacio marinamoreno$ cd DAM 2021-22/MP03\ -\ Programacio/
[C3PO:MP03 - Programacio marinamoreno$ cd UF1
[C3PO:UF1 marinamoreno$ ls
HelloWorld HelloWorld.java
[C3PO:UF1 marinamoreno$ javac HelloWorld.java
HelloWorld.java:1: error: class HolaMon is public, should be declared in a file
named HolaMon.java
public class HolaMon {
 ^
1 error
[C3PO:UF1 marinamoreno$ javac HelloWorld.java
[C3PO:UF1 marinamoreno$ ls
HelloWorld HelloWorld.class HelloWorld.java
[C3PO:UF1 marinamoreno$ java HelloWorld
Hola, món!
[C3PO:UF1 marinamoreno$ javac HolaMon.java
[C3PO:UF1 marinamoreno$ java HolaMon.java
Error: Could not find or load main class HolaMon.java
[C3PO:UF1 marinamoreno$ java HolaMon
Hola, món!
[C3PO:UF1 marinamoreno$ ]
```

5. Escrivim una comanda a terminal

javac HolaMon.java

⚠️javac: és la crida a programa que traduirà del nostre arxiu a codi binari.

⚠️ HolaMon.java és el nom de l'arxiu del programa que hem creat

Primer programa en JAVA

El primer programa de tot programador és: “Hola, món!”



6. Mirem que estigui ben compilat

Error:

```
C3PO:UF1 marinamoreno$ javac HelloWorld.java
HelloWorld.java:1: error: class HolaMon is public, should be declared in a file
named HolaMon.java
public class HolaMon {
^
1 error
```

```
C3PO:UF1 marinamoreno$ java HolaMon.java
Error: Could not find or load main class HolaMon.java
```

No error:

```
[C3PO:UF1 marinamoreno$ javac HolaMon.java
C3PO:UF1 marinamoreno$
```

Primer programa en JAVA

El primer programa de tot programador és: “Hola, món!”

7. Executem el programa i veiem què imprimim per pantalla

```
[C3P0:UF1 marinamoreno$ javac HolaMon.java
[C3P0:UF1 marinamoreno$ java HolaMon
Hola, món!
```

Primer programa en JAVA



Exercici de classe

Modifiquem el text que imprimim a pantalla.

- Obrim l'editor de text i modifiquem el text afegint el nostre nom
- Tornem a guardar el fitxer
- Obrim la terminal i escrivim
 - javac HolaMon.java
 - java HolaMon
- Comprovem que imprimim el nou text que hem escrit

```
public class HolaMon {  
    public static void main(String[] args) {  
        System.out.println("Hola, món!");  
    }  
}
```

Corred insensatos



Exercici de classe

1. Creem la carpeta “2-CorredInsensatos” al path “Documents/MP03-Programació/AEA1/”
2. Creem un document anomenat CorredInsensatos.java
3. Copiem el codi de l’arxiu HolaMon.java
4. Modifiquem tot el necessari per a que puguem compilar (nom de la public class, nom de l’arxiu,...) Recordeu que l’arxiu ha de tenir el nom de CorredInsensatos.java
5. Modifiquem el primer print escrivint “Corred! Corred! Corred!”
6. Afegim un segon print escrivint “JAAAAAA!”
7. Afegim un tercer print escrivint “Insensatos!”
8. Compilem el programa escrivint javac CorredInsensatos.java a terminal
9. Executem el programa escrivint java CorredInsensatos a terminal

Extra

Fem una versió canviada del “HolaMon” que es digui “CorredInsensatos”.

- Modifiquem el “public class...”
- Modifiquem el que imprimim a pantalla
- Afegim dos missatges més a la pantalla

Importar biblioteca



Una biblioteca és un conjunt d'extensions al conjunt d'instruccions disponibles quan genereu un programa.

```
import <nomBiblioteca>;
```

Exemple

```
import java.util.Scanner;

class MyClass {
    public static void main(String[] args) {
        Scanner myObj = new Scanner(System.in);
        System.out.println("Enter username");

        String userName = myObj.nextLine();
        System.out.println("Username is: " + userName);
    }
}
```

⚠ Cal que busquem per cada biblioteca que inclouem els mètodes/funcions que inclou.

Definició exacta de què inclou la [biblioteca/classe SCANNER de JAVA](#)

Manipulació bàsica de dades

Tipus de dades

El terme **dada** indica tota la informació que fa servir l'ordinador durant l'execució d'un programa.

Tipus de dades



El terme dada indica tota la informació que fa servir l'ordinador durant l'execució d'un programa.

Exemple

Quins tipus de dades podem tenir en un programa? Què ens estem imaginant?

Nombres enters

Paraules

Caràcters

Reals

Tipus de dades - Boolean

És un tipus de dada lògic que serveix per establir la certesa o falsedat d'una informació

- Valor: **true** o **false**

Exemple

```
public class ExempleBooleanCondicio {  
    public static void main(String[] args) {  
        boolean plou = false;  
  
        if (plou) {  
            System.out.println("Porta paraigua!");  
        } else {  
            System.out.println("Gaudeix del sol!");  
        }  
    }  
}
```

Tipus de dades - Enter

Representa una valor numèric, positiu o negatiu, sense cap decimal.

- Valor: **qualsevol número sense decimal**

Exemple

```
public class ExempleInt {  
    public static void main(String[] args) {  
        int numero1 = 10;  
        int numero2 = 5;  
  
        int suma = numero1 + numero2;  
  
        System.out.println("La suma és: " + suma);  
    }  
}
```

Tipus de dades - Real

Representa una valor numèric, positiu o negatiu, amb decimals.

- Valor: **qualsevol número amb decimals**

Exemple

```
public class ExempleDouble {  
    public static void main(String[] args) {  
        double numero1 = 5.75;  
        double numero2 = 2.25;  
  
        double suma = numero1 + numero2;  
  
        System.out.println("La suma és: " + suma);  
    }  
}
```

Tipus de dades - Caràcter

Representa una unitat bàsica de l'alfabet, ja sigui una lletra, número o signes de puntuació.

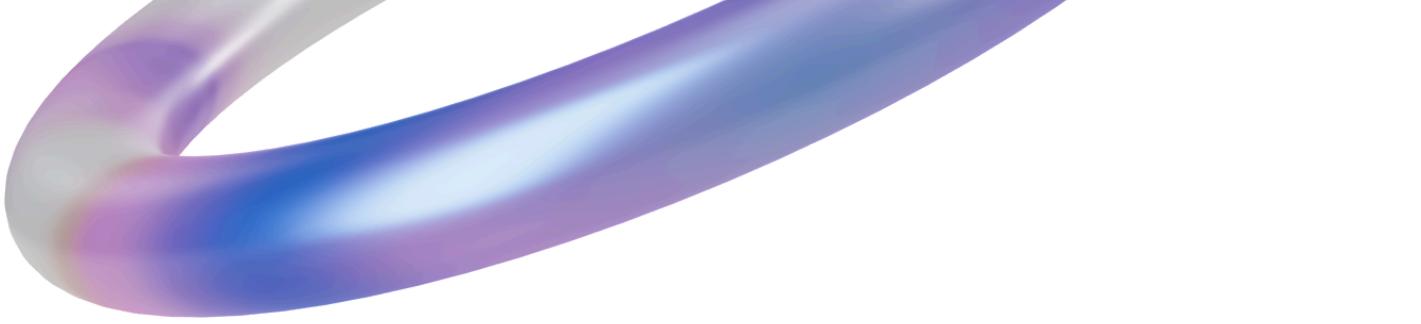
- Valor: **qualsevol element bàsic de l'alfabet**

[Documentació chars: taula ASCII](#)

Exemple

```
public class ExempleChar {  
    public static void main(String[] args) {  
        char lletra = 'A';  
        char numero = '7';  
        char simbol = '#';  
  
        System.out.println("Lletra: " + lletra);  
        System.out.println("Número: " + numero);  
        System.out.println("Símbol: " + simbol);  
    }  
}
```

Tipus de dades



Exercici de classe

Classifiqueu aquestes dades en les categories corresponents.

1. “DAM”

Booleà

2. false

Enter

3. 23

Real

4. 399,05

Caràcter

5. true

6. ‘2349’

A large, abstract graphic on the left side of the slide features several translucent, flowing shapes in shades of blue, purple, and white. One large, teardrop-shaped blob is prominent in the center, with smaller circular and irregular shapes to its left.

Transformació de les dades

Transformació de les dades - Boolean

Les **taules de la veritat** són el que utilitzem per saber les operacions que podem fer amb els booleans i el seu resultat.

Transformació de les dades - Boolean Negació !

Operació que dona com a resultat el valor contrari del que ha entrat.

Exemple

Operador	Operand
A	!A
TRUE	FALSE
FALSE	TRUE

Transformació de les dades - Boolean AND &&

Operació que necessita que ambdós dades siguin certes per a donar un resultat true.

⚠ Un valor false → operació false

Exemple

Operadors		Operand
A	B	A&B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	FALSE

Transformació de les dades - Boolean OR II

Operació que necessita que només una de les dues dades sigui certa per a donar un resultat true.

⚠ Un valor true → operació true

Exemple

Operadors		Operand
A	B	A B
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

Transformació de les dades - Boolean

Operadors		Operand		
A	B	!A	A&B	A B
TRUE	TRUE	FALSE	TRUE	TRUE
TRUE	FALSE	FALSE	FALSE	TRUE
FALSE	TRUE	TRUE	FALSE	TRUE
FALSE	FALSE	TRUE	FALSE	FALSE

Transformació de les dades - Boolean igual ==

Operació en què volem saber si les dades són iguals.

⚠️HELP: Pensem en...
“És cert que les dues dades estan en el mateix estat?”

Exemple

Operadors		Operand
A	B	A==B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	TRUE

Transformació de les dades - Boolean diferent !=

Operació en què volem saber si les dades són diferents.

⚠️HELP: Pensem en...
“És cert que les dues dades estan en estats diferents?”

Exemple

Operadors		Operand
A	B	$A \neq B$
TRUE	TRUE	FALSE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

Transformació de les dades - Boolean

Operadors		Operand	
A	B	$A==B$	$A!=B$
TRUE	TRUE	TRUE	FALSE
TRUE	FALSE	FALSE	TRUE
FALSE	TRUE	FALSE	TRUE
FALSE	FALSE	TRUE	FALSE

Transformació de les dades - Boolean

El terme dada indica tota la informació que fa servir l'ordinador durant l'execució d'un programa.

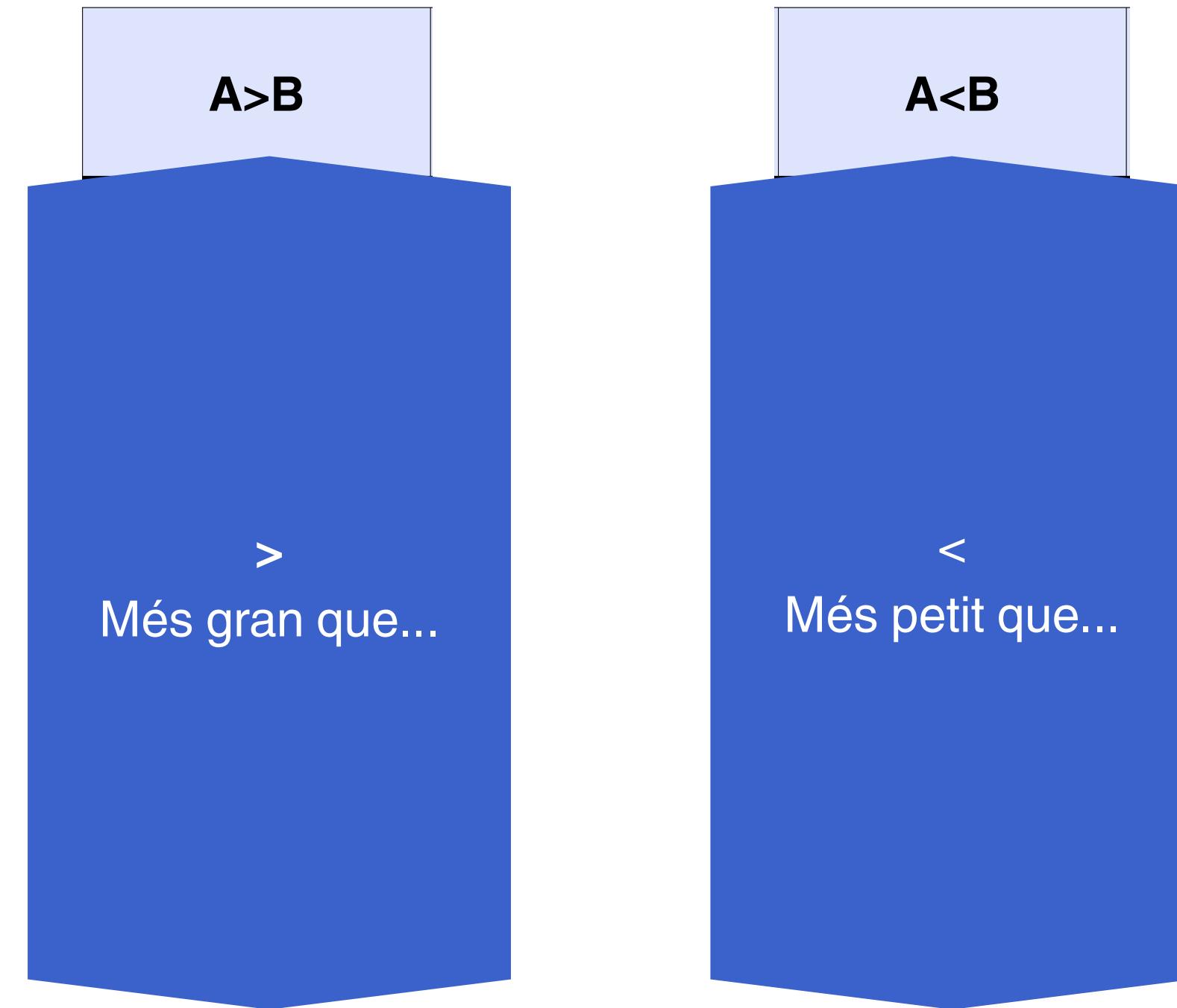
Exemple

```
public class ExempleBooleans {
    public static void main(String[] args) {
        boolean A = true;
        boolean B = false;

        System.out.println("A = " + A);
        System.out.println("B = " + B);

        System.out.println("A && B = " + (A && B));
        System.out.println("A || B = " + (A || B));
        System.out.println("!A = " + (!A));
        System.out.println("!B = " + (!B));
    }
}
```

Transformació de les dades - Boolean



Transformació de les dades - Boolean

A	B	A==B	A>B	A>=B	A<B	A<=B
900	45					
-67	89					
55	55					
3	20					

Transformació de les dades - Boolean

A	B	A==B	A>B	A>=B	A<B	A<=B
900	45	FALSE	TRUE	TRUE	FALSE	FALSE
-67	89	FALSE	FALSE	FALSE	TRUE	TRUE
55	55	TRUE	FALSE	TRUE	FALSE	TRUE
3	20	FALSE	FALSE	FALSE	TRUE	TRUE

Transformació de les dades

Tipus	Mida (bits)	Exemple de valor	Operacions principals
byte	8	<code>byte b = 100;</code>	+, -, *, /, % (operacions enteres)
short	16	<code>short s = 3000;</code>	+, -, *, /, %
int	32	<code>int x = 42;</code>	+, -, *, /, %, comparacions (<, >, ==, etc.)
long	64	<code>long l = 100000L;</code>	+, -, *, /, %, comparacions
float	32	<code>float f = 3.14f;</code>	+, -, *, /, %, comparacions (atenció a la precisió)
double	64	<code>double d = 2.718;</code>	+, -, *, /, %, comparacions (alta precisió)
char	16 (1 caràcter Unicode)	<code>char c = 'A';</code>	Comparacions (==, <, >), sumes amb enters (per codi Unicode)
boolean	1 (lògic)	<code>boolean b = true;</code>	Operacions lògiques: && (AND), `



Expressions

Expressions

Una expressió és una combinació d'operadors (*variables o valors*) i operands (*operació a fer*).

Una expressió és una combinació d'operadors i operands.

- **Qualsevol operació ha de ser sempre entre dades del mateix tipus.**
- **L'operació ha d'existir per al tipus de dada.**

Exemple

- true && false
- 4 + 2
- !true
- 3.9 / 3.0
- 'a' == 'b'
- costatQuadrat * 4
 - costatQuadrat és una variable entera

Expressions

Exercici a classe

Les següents expressions són sintàcticament correctes, però no ho són semànticament. No són expressions correctes, per què?

- $5.3 == '4'$



Expressions

Exercici a classe

Les següents expressions són sintàcticament correctes, però no ho són semànticament. No són expressions correctes, per què?

- **5.3 == '4'** Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- **true + false**



Expressions

Exercici a classe

Les següents expressions són sintàcticament correctes, però no ho són semànticament. No són expressions correctes, per què?

- $5.3 == '4'$ Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- `true + false` L'operació no existeix per als booleans
- `-'g'`



Expressions

Exercici a classe

Les següents expressions són sintàcticament correctes, però no ho són semànticament. No són expressions correctes, per què?

- $5.3 == '4'$ Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- $true + false$ L'operació no existeix per als booleans
- $-'g'$ L'operació no existeix per als caràcters
- $5 == \text{false}$



Expressions

Exercici a classe

Les següents expressions són sintàcticament correctes, però no ho són semànticament. No són expressions correctes, per què?

- $5.3 == '4'$ Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- $true + false$ L'operació no existeix per als booleans
- $-'g'$ L'operació no existeix per als caràcters
- $5 == \text{false}$ Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- $5 \parallel 4.0$



Expressions

Exercici a classe

Les següents expressions són sintàcticament correctes, però no ho són semànticament. No són expressions correctes, per què?

- $5.3 == '4'$ Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- $true + false$ L'operació no existeix per als booleans
- $-'g'$ L'operació no existeix per als caràcters
- $5 == \text{false}$ Tot i que reals i caràcters disposen de l'operació d'igualtat, els dos literals són de tipus de dades diferents.
- $5 \parallel 4.0$ Es fa una operació entre tipus de dades diferents, a part la OR no existeix per números enters ni números per reals.



Avaluació d'expressions

L'**ordre de precedència** d'un conjunt d'operadors és la **regla** usada per establir de manera no ambigua l'ordre **com s'han de resoldre les operacions dins d'una expressió**.

Ordre	Operació	Operador
1	Canvi de signe	-(unari)
2	Producte, divisió i mòdul	* / %
3	Suma i resta	+ -
4	Relacionals de comparació	> < <= >=
5	Relacionals d'igualtat	== !=
6	Negació	! (unari)
7	Conjunció	&&
8	Disjunció	

Avaluació d'expressions

Exercici a classe

Què imprimirà la terminal?

```
//Comprova una expressió complexa
public class ProvaExpressio {
    public static void main(String[] args) {
        System.out.println(((3 + 4) == 7 )&&!(12.3 > 2.11)||('a' == 'b'));
    }
}
```

Avaluació d'expressions

Exercici a classe

Què imprimirà la terminal?

Fixeu-vos que en aquesta expressió apareixen literals de tipus de dades totalment diferents. Tot i així, les operacions són amb tipus de dades iguals.

1) Parèntesis més interns

```
//Comprova una expressió complexa
public class ProvaExpressio {
    public static void main(String[] args) {
        System.out.println((3 + 4) == 7 )&&(12.3 > 2.11)||('a' == 'b'));
    }
}
```

((3+4)==7) && ! (12.3>2.11) || ('a'=='b')



((7)==7) && ! (12.3>2.11) || ('a'=='b')

Avaluació d'expressions

Exercici a classe

Què imprimirà la terminal?

Fixeu-vos que en aquesta expressió apareixen literals de tipus de dades totalment diferents. Tot i així, les operacions són amb tipus de dades iguals.

```
//Comprova una expressió complexa
public class ProvaExpressio {
    public static void main(String[] args) {
        System.out.println((3 + 4) == 7 )&&(12.3 > 2.11)||('a' == 'b'));
    }
}
```

1) Parèntesis més interns

((3+4)==7) && !(12.3>2.11) || ('a'=='b')

2) Parèntesis més externs

((7)==7) && !(12.3>2.11) || ('a'=='b')

(true) && ! (true) || (false)

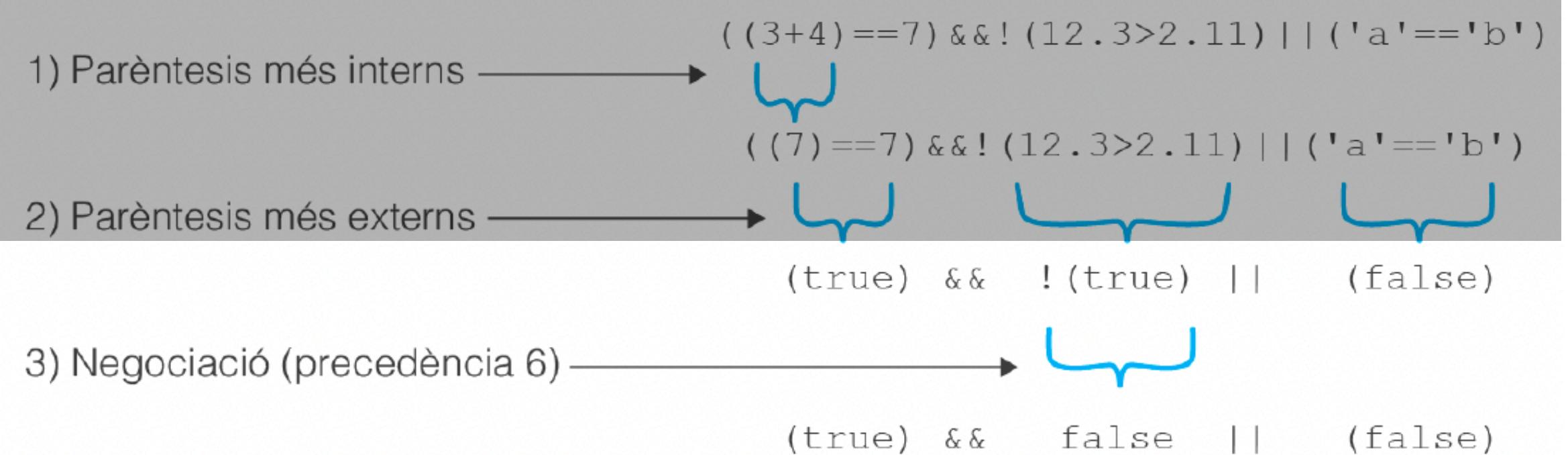
Avaluació d'expressions

Exercici a classe

Què imprimirà la terminal?

Fixeu-vos que en aquesta expressió apareixen literals de tipus de dades totalment diferents. Tot i així, les operacions són amb tipus de dades iguals.

```
//Comprova una expressió complexa
public class ProvaExpressio {
    public static void main(String[] args) {
        System.out.println((3 + 4) == 7 )&&(12.3 > 2.11)||('a' == 'b'));
    }
}
```



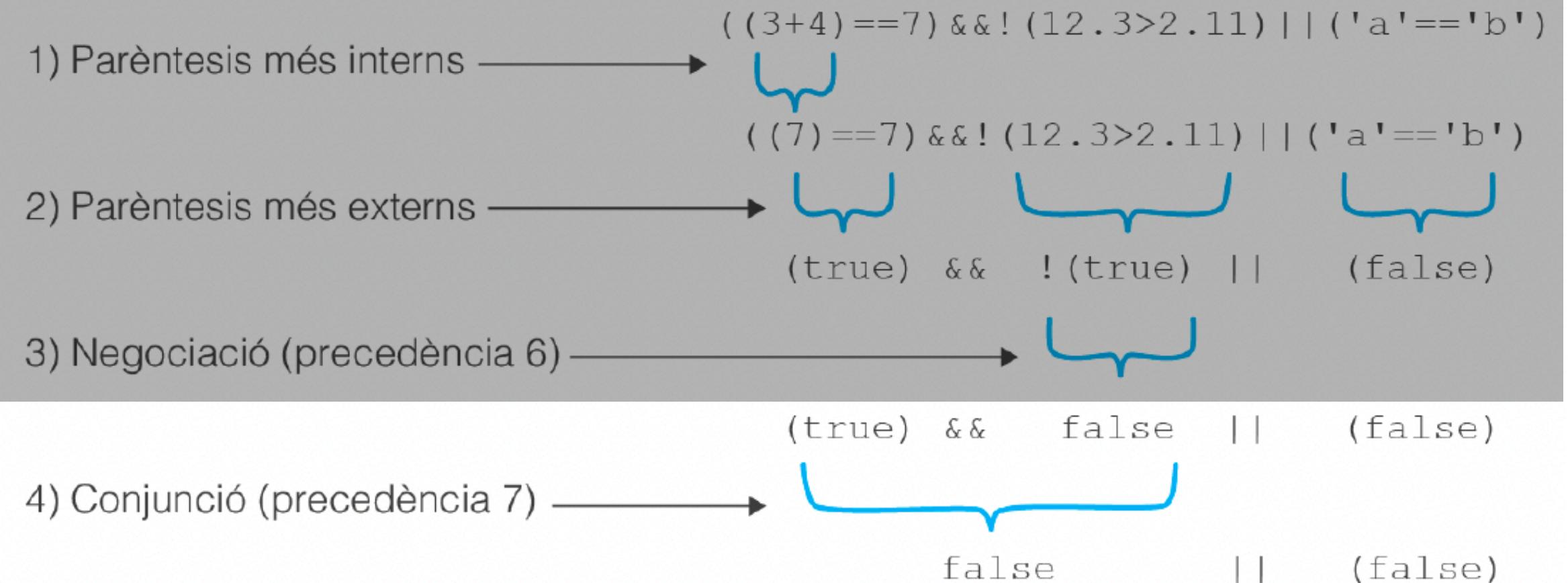
Avaluació d'expressions

Exercici a classe

Què imprimirà la terminal?

Fixeu-vos que en aquesta expressió apareixen literals de tipus de dades totalment diferents. Tot i així, les operacions són amb tipus de dades iguals.

```
//Comprova una expressió complexa
public class ProvaExpressio {
    public static void main(String[] args) {
        System.out.println((3 + 4) == 7 )&&!(12.3 > 2.11)||('a' == 'b'));
    }
}
```



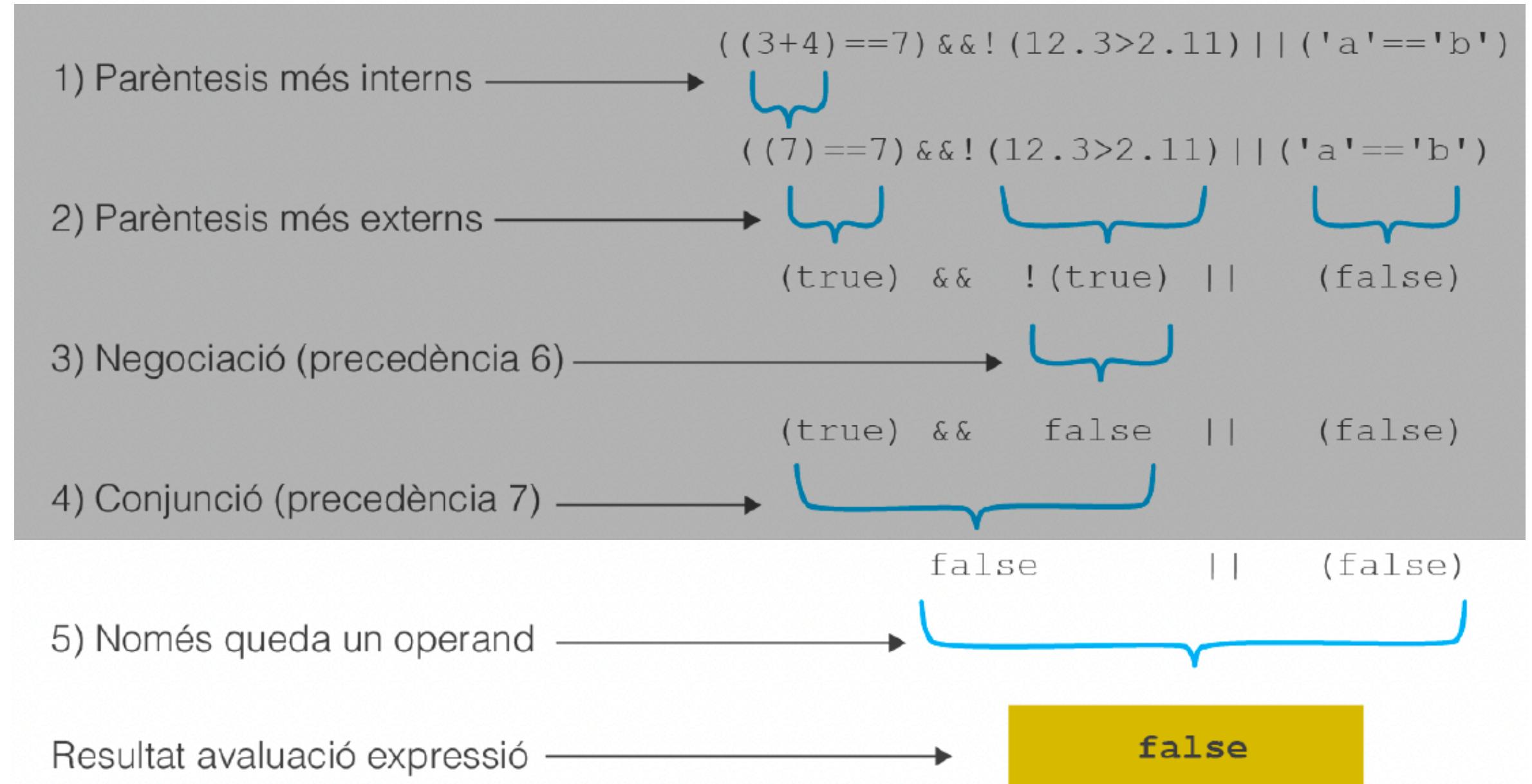
Avaluació d'expressions

Exercici a classe

Què imprimirà la terminal?

Fixeu-vos que en aquesta expressió apareixen literals de tipus de dades totalment diferents. Tot i així, les operacions són amb tipus de dades iguals.

```
//Comprova una expressió complexa
public class ProvaExpressio {
    public static void main(String[] args) {
        System.out.println((3 + 4) == 7 )&&!(12.3 > 2.11)||('a' == 'b'));
    }
}
```



A large, abstract graphic on the left side of the slide features several translucent, rounded, and flowing shapes in shades of blue, purple, and white. One large, teardrop-shaped blob occupies the center-left, with smaller, more spherical shapes above and to the left of it.

Desbordaments i errors de precisió

Desbordaments i errors de precisió

Un fet molt important que heu de tenir en compte **quan avalueu expressions aritmètiques entre dades de tipus numèric (enters i reals)**, és que aquests tipus tenen una limitació en els valors que poden representar.

L'ordinador no pot representar la infinitat de números que podem crear.

> Per tant, els llenguatges de programació només poden representar un rang concret en tots dos casos.

⚠ Si mai se supera aquest rang, es diu que heu patit un desbordament (overflow) i el resultat de qualsevol expressió serà sempre incorrecte.



Desbordaments i errors de precisió



Tipus	Paraula clau Java	Mida (bits)	Rang
byte	byte	8	-128 a 127
enter curt	short	16	-32768 a 32767
enter simple	int	32	-2147483648 a 2147483648
enter llarg	long	64	-9,223,372,036,854,775,808 a 9,223,372,036,854,775,808
real de simple precisió	float	32	-3.40292347*10^38 a 3.40292347*10^38
real de doble precisió	double	64	-4.94065645841246544*10^24 a 1.79769313486231570*10^308

A large, abstract graphic on the left side of the slide features several translucent, flowing shapes in shades of blue, purple, and white. One large, teardrop-shaped blob is prominent in the center, with smaller, spherical shapes above and to the left.

Operacions amb diferents tipus de dades

Operacions amb diferents tipus de dades

Sempre que s'avalua una **operació binària** en què apareixen **dades de tipus different però compatibles**, el **tipus del resultat serà el mateix que l'operand amb més rang**.

Exemple

$$(2+5) * 40L = 280L$$

Enter long



true * 3.9

boolean real

 operació no possible
perquè son tipus de dades
diferents no compatibles.

A large, abstract, organic shape composed of several translucent, overlapping blobs in shades of blue, purple, and white. It has a glossy, liquid-like texture and a organic, flowing form.

Variables

Variable

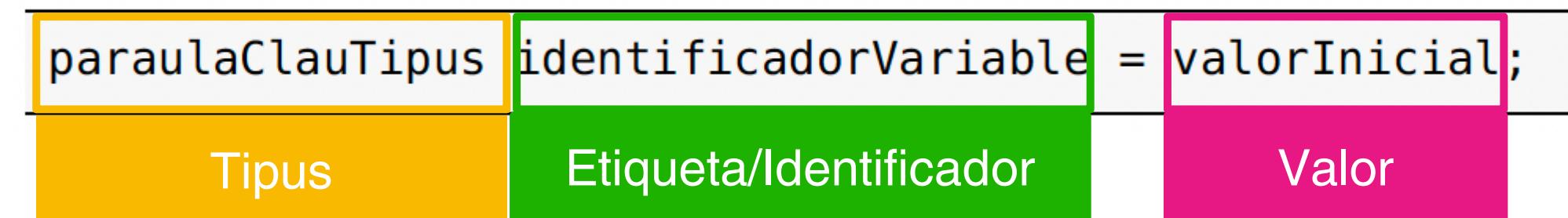
Una **variable** és una **dada emmagatzemada a la memòria** que pot veure modificat el seu valor en qualsevol moment durant l'execució del programa.

Tota variable dins del codi font d'un programa ha d'haver estat **declarada** prèviament pel programador abans de poder fer-la servir.

⚠ Variable: Tipus de dada que volem tractar/gestionar/executar/...

Declaració de variables

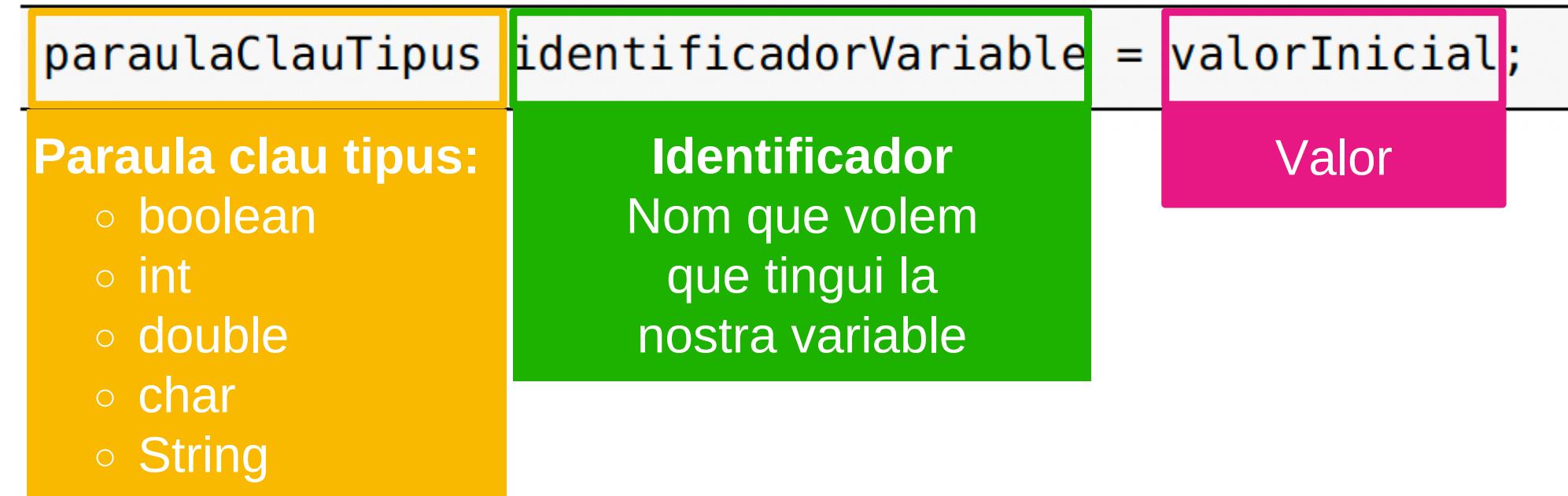
Per declarar una variable dins el vostre programa, només cal especificar tres coses: el **tipus**, un **identificador** o etiqueta únic i un **valor inicial**.



1. Tot i que la sintaxi per declarar una variable pot variar segons el llenguatge de programació, gairebé sempre cal definir aquestes tres coses.
2. L'**identificador** ens permet diferenciar les diferents variables que hi hagi dins del codi font del programa, el **tipus** delimita quins valors pot emmagatzemar i quines operacions són aplicables, i el **valor inicial** el contingut que hi haurà tan bon punt s'ha declarat.

⚠ Evidentment, el valor inicial ha de ser una dada del mateix tipus que la variable.

Declaració de variables



Exemple

```
//El programa que només declara una variable de tipus enter
public class DeclararEnter {
    public static void main(String[] args) {
        int elMeuEnter = 2;
        //Ara hi ha una variable en memòria que conté el valor enter 2.
    }
}
```

Declaració de variables

Exercici de classe

- Quin tipus de variable és?
- Com es diu la variable?
- Quin valor té?

paraulaClauTipus identificadorVariable = valorInicial;

```
//El programa que declara de cop dues variables de tipus enter
public class DeclararEnter {
    public static void main(String[] args) {
        int elMeuEnter = 2, elMeuAltreEnter = 4;
        //Ara hi ha dues variables a la memòria
        //Contenen els valors enters 2 i 4, respectivament.
    }
}
```

Declaració de variables

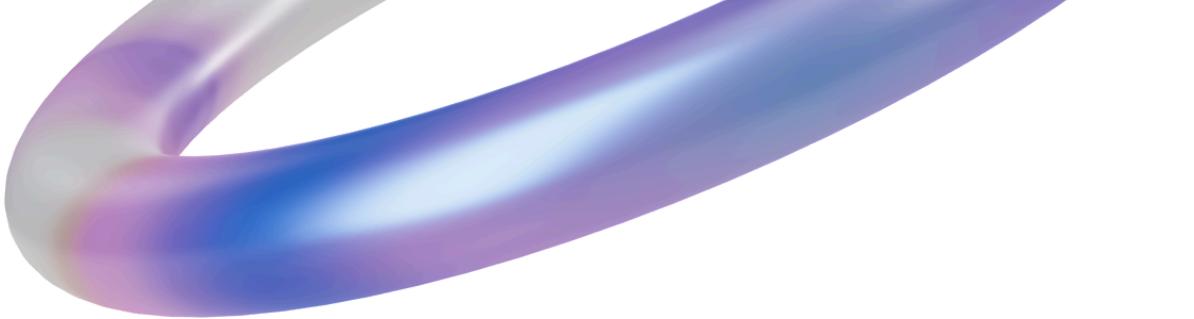
Exercici de classe

- Quin tipus de variable és?
- Com es diu la variable?
- Quin valor té?

paraulaClauTipus identificadorVariable = valorInicial;

```
//El programa que declara de cop dues variables de tipus enter
public class DeclararEnter {
    public static void main(String[] args) {
        int elMeuEnter = 2, elMeuAltreEnter = 4;
        //Ara hi ha dues variables a la memòria
        //Contenen els valors enters 2 i 4, respectivament.
    }
}
```

Declaració de variables



Exercici de classe

Creeu un nou document .java i feu la creació de les següents variables:

- Declareu un enter amb valor 5
- Declareu un booleà amb valor true
- Declareu un real amb valor 9.99
- Declareu un caràcter amb valor ‘a’
- Declareu una paraula amb valor “barcelona”

```
paraulaClauTipus identificadorVariable = valorInicial;
```

Què hem fet en cada cas?

Declaració de variables

Exercici de classe

Creeu un nou document .java i feu la creació de les següents variables:

- Declareu un enter amb valor 5 —> **int divisor = 5;**
- Declareu un booleà amb valor true —> **boolean estudiant = true;**
- Declareu un real amb valor 9.99 —> **double radi = 9.99;**
- Declareu un caràcter amb valor ‘a’ —> **char lletra = ‘a’;**
- Declareu una paraula amb valor “barcelona” —> **String ciutat = “barcelona”**

```
paraulaClauTipus identificadorVariable = valorInicial;
```

Què hem fet en cada cas?

Ara feu un print per terminal per cadascuna de les variables que heu creat

Identificador: nom de la variable

Un identificador/etiqueta pren la forma d'un text arbitrari. Podeu triar el que més us agradi.

De totes formes, **és molt i molt recomanable que sempre utilitzeu algun text que us ajudi a entendre fàcilment què s'està emmagatzemant a cada variable**, ja que això ajuda a entendre què fa el codi font del vostre programa.

Exemple

```
//Programa A. Un codi poc entenedor.  
public class DivideixISuma {  
    public static void main(String[] args) {  
        double x = 20.0;  
        double y = 6.0;  
        double z = 3.0;  
        //Ara vindria la resta del codi  
        ...  
    }  
}
```

```
//Programa B. Un codi més entenedor.  
public class DivideixISuma {  
    public static void main(String[] args) {  
        double dividend = 20.0;  
        double divisor = 6.0;  
        double sumarAlFinal = 3.0;  
        //Ara vindria la resta del codi  
        ...  
    }  
}
```

Tot i que la declaració és la mateixa si ens fixem en el tipus de dada, l'identificador/etiqueta ens facilita entendre què representa aquell valor.

Identificador: nom de la variable

Regles dels identificadors/etiquetes

- No pot contenir espais
- No pot començar amb un número
- Millor no utilitzar accents
- No pot ser una paraula clau del llenguatge. *Una paraula clau (o reservada) d'un llenguatge de programació és aquella que té un significat especial dins la seva sintaxi i s'usa per compondre certes parts o instruccions en el codi font d'un programa. Les paraules reservades son:*

abstract, continue, for, new, switch, assert, default, goto,
package, synchronized, boolean, do, if, private, this, break,
double, implements, protected, throw, byte, else, import,
public, throws, case, enum, instanceof, return, transient,
catch, extends, int, short, try, char, final, interface,
static, void, class, finally, long, strictfp, volatile, const,
float, native, super, while

Declaració de variables



Exercici de classe

Fem dos exercicis separats. Per cadascun d'ells, creeu un document .java diferent

1. Multipliquem nombres entrats per terminal —> Multiplicació dinàmica.java
2. Calculem l'àera d'un triangle. —> AreaTriangle.java
3. Feu un programa que imprimeixi el següent text: “ Bona tarda! Estem a classe de DAM!”
 - a. Utilitzant 3 variables diferents:
 - i. Bona tarda
 - ii. Estem a classe de DAM
 - iii. !

A large, abstract graphic on the left side of the slide features several translucent, flowing shapes in shades of blue, purple, and white. One large, teardrop-shaped blob is prominent in the center, with smaller, rounded shapes above and to the left.

Seqüències d'escapament

Seqüències d'escapament

En alguns llenguatges de programació, com en Java, hi ha un conjunt de **literals** dins dels caràcters que són especialment **útils dins de les cadenes de text**, ja que permeten **representar símbols o accions amb un significat especial**.

S'anomenen seqüències d'escapament.

Exemple

```
//Mostra text usant caràcters de control.
public class CaractersControl{
    public static void main(String[] args) {
        System.out.println("Línia 1\n\tLínia2\nLínia 3");
    }
}
```

Sortida

Línia 1
Línia 2
Línia 3

Seqüències d'escapament

Exercici de classe

1. Feu un programa on la terminal imprimeixi això:

```
[c3po-1:6 - Java dia 2 marinamoreno$ java SequenciaEscapament
    Fútbol
Club
    Barcelona
"1899"
```

2. Feu un programa que mostri en pantalla de manera tabulada la taula de veritat d'una expressió d'AND entre dues variables booleanes.

Exemple taula OR

```
[c3po-1:6 - Java dia 2 marinamoreno$ java TaulaBooleanaOR
A      B      (A || B)
false  false  false
true   false  true
false  true   true
true   true   true
```

Seqüència d'escapament	Acció o símbol representat
\t	Una tabulació.
\n	Un salt de línia i retorn de carro.
\`	El caràcter “cometa simple”.
\\"	El caràcter “cometes dobles” ("").
\\\	El caràcter “contrabarra” (\).



Algoritmes

Algoritme



Un algoritme és un **conjunt ordenat i finit de passos** o instruccions que serveixen per resoldre un problema o per aconseguir un objectiu concret.

Característiques

Finit

Clar i precís

Ordenat

Eficaç

Ha de tenir un nombre limitat de passos.

Cada pas s'ha d'entendre sense ambigüïtats.

Les instruccions segueixen una seqüència lògica.

Ha de conduir al resultat correcte quan s'aplica correctament.

Algoritme



Exemple

1

Un algoritme per sumar dos nombres:

1. Llegir el primer nombre.
2. Llegir el segon nombre.
3. Sumar els dos nombres.
4. Mostrar el resultat.

2

Un algoritme per fer un entrepà podria ser:

1. Agafar dues llesques de pa.
2. Untar mantega en una de les llesques.
3. Afegir-hi formatge i pernil.
4. Tancar l'entrepà amb l'altra llesca.

Algoritme

Un algoritme és un conjunt finit d'instruccions o passos que serveixen per a executar una tasca o resoldre un problema.

1. Definició del problema

2. Disseny de l'algoritme

3. Implementació del programa

4. Verificació i proves

5. Posada en marxa i manteniment

Exercici per entregar



Multiplicació de 3 nombres reals

- > Feu un programa que mostri per pantalla la multiplicació de tres nombres reals entrats per teclat.

Sortida:

```
Introduceix el primer nombre: 2.5
Introduceix el segon nombre: 3
Introduceix el tercer nombre: 4
El resultat de la multiplicació és: 30.0
```

A large, abstract, organic shape composed of several translucent, overlapping blobs in shades of blue, purple, and white. It has a glossy, liquid-like texture and a somewhat organic, brain-like form.

Condicionals

Condicionals



Un condicional serveix per prendre decisions en un programa.

L'ordinador comprova una condició (si és certa o falsa) i, segons el resultat, executa unes instruccions o unes altres.

Exemple

Imaginem un videojoc on un personatge té punts de vida.

- Si la vida arriba a 0 o menys, el jugador mor.
- Si la vida és menor de 30, mostrarem que està en perill.
- En qualsevol altre cas, direm que està sa.

Condicionals - IF



S'utilitza quan volem fer una acció només si la condició és certa.

Exemple

Tenim aquest codi:

```
int edat = 20;

if (edat >= 18) {
    System.out.println("Ets major d'edat.");
}
```

L'edat és 20, per tant la sortida per terminal és “Ets major d'edat”.

Condicionals - IF/ELSE



Serveix per triar entre dues opcions:

- Una si la condició és certa.
- Una altra si la condició és falsa.

Exemple

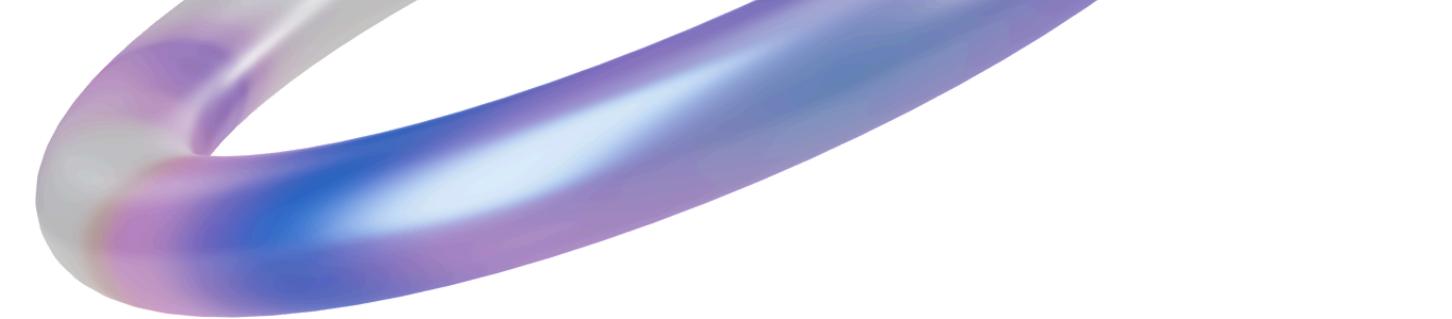
Tenim aquest codi:

```
int edat = 15;

if (edat >= 18) {
    System.out.println("Ets major d'edat.");
} else {
    System.out.println("Ets menor d'edat.");
}
```

L'edat és 15, per tant la sortida per terminal és “Ets menor d'edat”.

Exercici per entregar



Descompte 95

> Ara volem aplicar un descompte a un preu depenent del seu valor. Decidim que el descompte és del 15% i el preu mínim per a aplicar-lo és de 95€.

Sortida:

Introduceix el preu del producte: **120**
S'ha aplicat el descompte.
El preu final és: **102.0** €

Introduceix el preu del producte: **80**
No s'ha aplicat el descompte.
El preu final és: **80.0** €

Condicionals - IF/ELSE IF/ ELSE

S'utilitza quan hi ha més d'una condició.

Exemple

Tenim aquest codi:

```
int nota = 7;

if (nota >= 9) {
    System.out.println("Excel·lent!");
} else if (nota >= 5) {
    System.out.println("Has aprovat.");
} else {
    System.out.println("Has suspès.");
}
```

La nota és 7, per tant la sortida per terminal és “Has aprovat”.

Exercici per entregar



Descompte 100

> Creeu un programa per a que, en lloc de fer un descompte del 8% si la compra és de 100€ o més, apliqui una penalització de 2€ si el preu és inferior a 30€.

Sortida:

Introduceix el preu de la compra: **150**
S'ha aplicat un descompte del **8%**.
El preu **final** és: **138.0 €**

Introduceix el preu de la compra: **25**
S'ha aplicat una penalització de **2€**.
El preu **final** és: **23.0 €**

Introduceix el preu de la compra: **60**
No hi ha descompte ni penalització.
El preu final és: **60.0 €**

Condicionals - SWITCH

Quan hi ha moltes opcions basades en el valor d'una variable, és més còmode usar switch.

Exemple

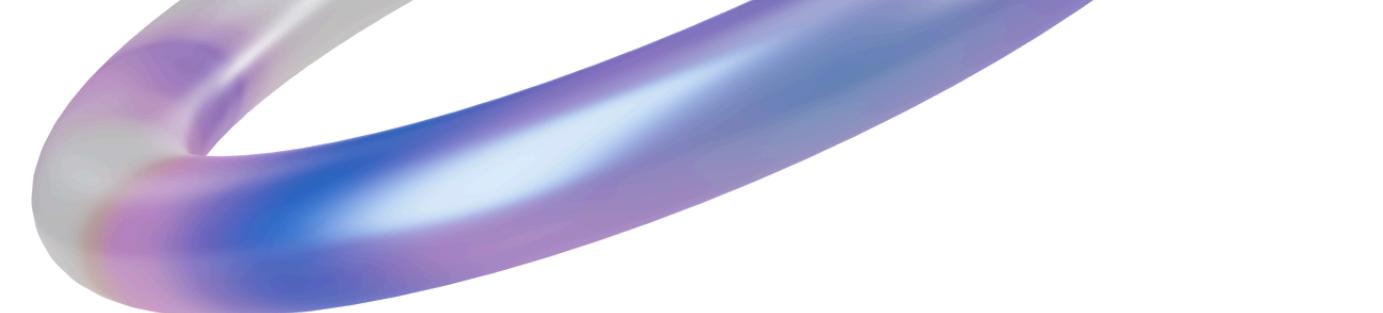
Tenim aquest codi:

El dia és 3, per tant s'imprimirà “Dimecres”

```
int dia = 3;

switch (dia) {
    case 1:
        System.out.println("Dilluns");
        break;
    case 2:
        System.out.println("Dimarts");
        break;
    case 3:
        System.out.println("Dimecres");
        break;
    case 4:
        System.out.println("Dijous");
        break;
    case 5:
        System.out.println("Divendres");
        break;
    case 6:
        System.out.println("Dissabte");
        break;
    case 7:
        System.out.println("Diumenge");
        break;
    default:
        System.out.println("Número de dia incorrecte");
}
```

Exercici per entregar



Subscripció de videojoc

> Creeu un programa que demani a l'usuari un número del 1 al 4 i mostri quin tipus de subscripció de videojoc té:

- 1 → Bàsica
- 2 → Estàndard
- 3 → Premium
- 4 → VIP

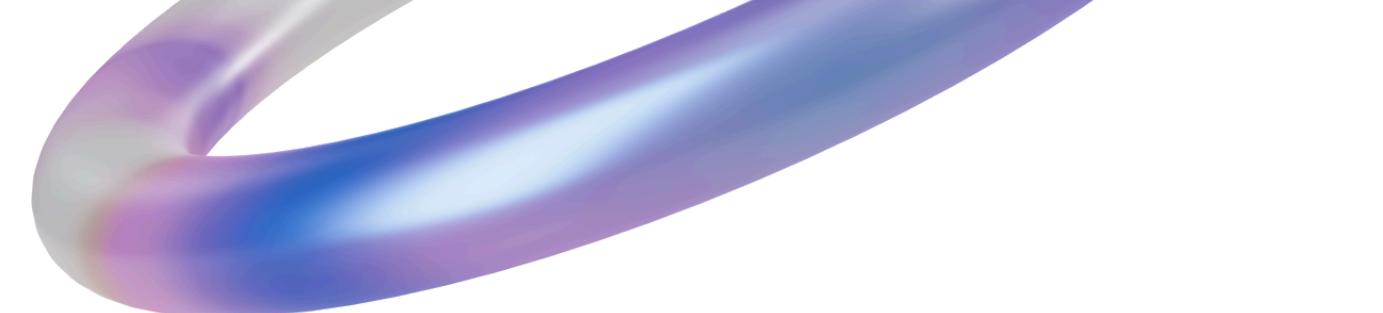
Si l'usuari introduceix un número diferent, s'ha de mostrar "Opció no vàlida".

Sortida:

```
Introduceix el tipus de subscripció (1-4): 3
Has escollit la subscripció Premium.
```

```
Introduceix el tipus de subscripció (1-4): 5
Opció no vàlida.
```

Exercici per entregar



Càlcul d'àrees

- > Creeu un programa que depenen de l'input que l'usuari escrigui per terminal, faci un càlcul o un altre.
- Si l'usuari escriu “triangle”, calcularem l'àrea del triangle.
 - Si l'usuari escriu “cercle”, calcularem l'àrea del cercle.
 - Si l'usuari escriu “trapezi”, calcularem l'àrea del trapezi.
 - Si l'usuari escriu “rombe”, calcularem l'àrea del rombe.
 - Si l'usuari escriu “paral·lelogram”, calcularem l'àrea del paral·lelogram.

Sortida:

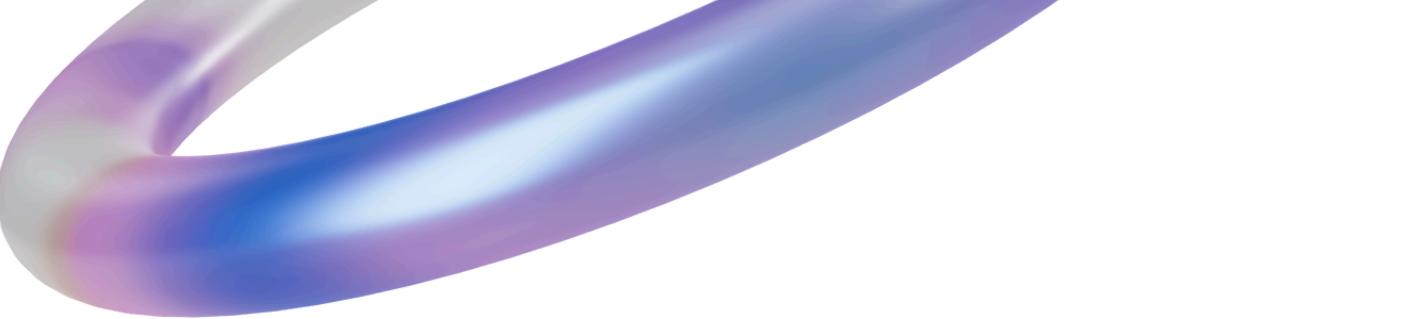
```
Escriu una figura (triangle, cercle, trapezi, rombe, paral·lelogram): trapezi
Introduceix la base major: 8
Introduceix la base menor: 4
Introduceix l'altura: 3
Àrea del trapezi: 18.0
```

```
Escriu una figura (triangle, cercle, trapezi, rombe, paral·lelogram): rombe
Introduceix la diagonal major: 6
Introduceix la diagonal menor: 4
Àrea del rombe: 12.0
```

A large, abstract graphic on the left side of the slide features several translucent, flowing shapes in shades of blue, purple, and white. One large, teardrop-shaped blob occupies most of the center-left, with smaller circular and irregular shapes to its left and bottom.

Àmbit de les variables

Àmbit de les variables



Exercici de classe

Què passa si modifiquem el `System.out.println()` i volem imprimir per pantalla la variable `descompteFet`?

Com ho solucionem?

```
import java.util.Scanner;
//Un programa que calcula descomptes.
public class DescompteControlErrors {

    public static void main (String[] args) {
        Scanner lector = new Scanner(System.in);
        //PASOS 2 i 3
        System.out.print("Quin és el preu del producte, en euros? ");
        float preu = lector.nextFloat();
        lector.nextLine();

        //declarem variables
        //Es fa un descompte del 8%.
        float DESCOMPTE = 8;
        //Es fa descompte per compres de 100 euros o més.
        float COMPRA_MIN = 100;
        //Valor del descompte màxim: 10 euros.
        float DESC_MAXIM = 10;

        //PAS 4. El preu és positiu?
        if (preu > 0) {
            //PAS I
            if (preu >= COMPRA_MIN) {
                //PAS a
                float descompteFet = preu * DESCOMPTE / 100;
                //PAS b. Si el descompte supera el màxim, cal reduir-lo.
                if (descompteFet > DESC_MAXIM) {
                    //PAS alfa. Reduir descompte
                    descompteFet = DESC_MAXIM;
                }
                //PAS c
                preu = preu - descompteFet;
            }
            //PAS III. Es mostra el preu final.
            System.out.println("El preu final per pagar és de " + preu + " euros.");
        } else {
            //PAS III. El preu era negatiu. Cal avisar l'usuari.
            System.out.println("Preu incorrecte. És negatiu.");
        }
    }
}
```

Àmbit de les variables

Exercici de classe

Què passa si modifiquem el `System.out.println()` i volem imprimir per pantalla la variable `descompteFet`?

Com ho solucionem?

```
import java.util.Scanner;
//Un programa que calcula descomptes.
public class DescompteControlErrors {

    public static void main (String[] args) {
        Scanner lector = new Scanner(System.in);
        //PASSOS 2 i 3
        System.out.print("Quin és el preu del producte, en euros? ");
        float preu = lector.nextFloat();
        lector.nextLine();

        //declarem variables
        //Es fa un descompte del 8%.
        float DESCOMPTE = 8;
        //Es fa descompte per compres de 100 euros o més.
        float COMPRA_MIN = 100;
        //Valor del descompte màxim: 10 euros.
        float DESC_MAXIM = 10;
        //variable descompteFet
        float descompteFet = 0;
        //PAS 4. El preu és positiu.
        if (preu > 0) {
            //PAS I
            if (preu >= COMPRA_MIN) {
                //PAS a
                descompteFet = preu * DESCOMPTE / 100;
                //PAS b. Si el descompte supera el màxim, cal reduir-lo.
                if (descompteFet > DESC_MAXIM) {
                    //PAS alfa. Reduir descompte
                    descompteFet = DESC_MAXIM;
                }
                //PAS c
                preu = preu - descompteFet;
            }
            //PAS II. Es mostra el preu final.
            //System.out.println("El preu final per pagar és de " + preu + " euros.");
            System.out.println("El descompte aplicat és de " + descompteFet + " euros.");
        }
        else {
            //PAS III. El preu era negatiu. Cal avisar l'usuari.
            System.out.println("Preu incorrecte. És negatiu.");
        }
    }
}
```

Declarar la variable en un àmbit superior, per a que fora del IF es conegui l'existència de la variable.

Exercicis per repassar



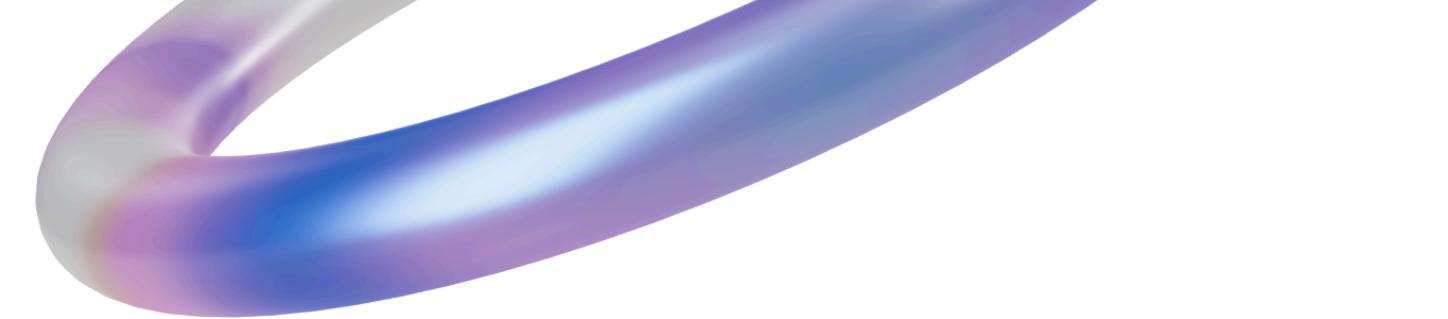
Exercicis que no s'entreguen

1. Calcular quants dies té un mes. A grans trets, el programa faria el següent:

1. Preguntar el número de mes (un valor de tipus enter).
2. Llegir-lo.
3. Segons el valor del mes:
 - (a) Si és 2 cal dir que el nombre de dies és 28 o 29.
 - (b) Si és 4, 6, 9 o 11 cal dir que el nombre de dies és 30.
 - (c) Si és 1, 3, 5, 7, 8, 10 o 12 cal dir que el nombre de dies és 31.
 - (d) Si és cap altre, cal dir que s'ha introduït un número de mes incorrecte.

2. Escriu un programa Java que llegeixi un nombre enter i imprimeix "zero" si el nombre és zero. En cas contrari, escriviu "positiu" o "negatiu". Afegiu "petit" si el valor absolut del nombre és inferior a 10, o "gran" si supera 1.000.000.

Exercicis per repassar



Exercicis que no s'entreguen

3. Escriu un programa Java que accepti tres nombres i imprimeix "Tots els números són iguals" si els tres són iguals, "Tots els números són diferents" si els tres números són diferents i "Ni tots són iguals ni diferents" en cas contrari. Poden ser nombres enters o reals.
4. Escriu un programa que accepti tres números de l'usuari i imprimeix "creixent" si els números estan en ordre creixent, "decreixent" si els números estan en ordre decreixent i "Ni ordre creixent ni decreixent" en cas contrari.
5. Escriu un programa Java que reculli 2 caràcters per terminal i digui quin caràcter és més gran. Si són iguals, cal dir-ho.
6. Escriu un programa Java per calcular la suma de dos nombres enters i retorna cert si la suma és igual a un tercer enter.
7. Escriu un programa Java que depenen de la paraula que l'usuari escrigui per terminal, imprimeixi un missatge específic.
 - Inputs acceptades pel programa: samsung, xiaomi, iphone
 - En cas que l'usuari escrigui ...
 - samsung → Output: "Buy a Samsung phone"
 - xiaomi → Output: "Buy a Xiaomi phone"
 - iphone → Output: "Buy an iPhone"
8. Escriu un programa Java que demani dos enters per terminal. A més a més, l'usuari ha d'indicar quina operació vol fer. Depenent del signe que l'usuari entri per terminal (+, -, /, *) faci l'operació que pertoca. Gestioneu els errors en cas que l'usuari entri una paraula i si aquest char és o no un dels símbols acceptats (+, -, /, *).



Gràcies

mmoreno@pratfp.com

Desenvolupament d'aplicacions multiplataforma

Prat