

Anem a veure la forma més simple de fer sortida i entrada: la sortida per pantalla l'entrada per teclat.

Sortida per pantalla

Aquestes instruccions es troben a la llibreria `java.lang`. No ho indicarem, ja que es carrega per defecte.

```
public class Sortides { // Proves de diferents formes de fer sortides
    public static void main (String[] args) {
        int x=125;
        // Mostra per pantalla el valor de la variable x
        System.out.print(x);
        //Mostra el valor de la variable x i canvia de línia
        System.out.println(x);
        //Mostra una cadena
        System.out.println("Hola món!");
        //Exemple amb seqüències d'escapament
        System.out.println("Hola \n\t com \r anam");
    }
}
```

Ex. 1. Escriu el codi anterior i comprova com funciona.

Entrada per pantalla

Per poder fer això haurem d'utilitzar elements que encara no s'han explicat. Indicarem com es fa i ja ho anirem entenent. Per ara ho utilitzarem talment com apareix a l'exemple:

```
import java.util.Scanner; // Indicam la biblioteca on es troben les funcions
d'entrada
public class NombreQuadrat {

    public static void main (String[] args) {
        Scanner lector= new Scanner(System.in); //Cream una instància de lector
        System.out.println("Entra un nombre: "); // Mostram missatge a l'usuari
        int nombre=lector.nextInt();
        // En aquest moment l'usuari pot anar entrant caràcters
        // No continuarà fins que es pitgi retorn
        lector.nextLine(); // Buida el buffer d'entrada
        int quadrat=nombre*nombre; // Feim l'operació
        System.out.println("El quadrat és: "+quadrat); // Mostram el resultat
    }
}
```

Ex. 2. Escriu el codi anterior i comprova com funciona. Fes proves amb entrades diferents.

Les instruccions de lectura per teclat que tenim a disposició són:

```
lector.nextByte()
lector.nextShort()
lector.nextInt()
lector.nextLong()
lector.nextFloat()
lector.nextDouble()
lector.nextBoolean()
```

```
lector.next()
```

En aquest cas agafa un String fins al següent espai.

```
lector.nextLine()
```

En aquest cas agafa un String fins Intro.

Consideracions:

- D'aquesta manera no es poden llegir caràcters individuals
- Comprovau quin és el caràcter de decimal (. o ,) que utilitza la vostra configuració.

Ex. 3. Escriu un programa que demani la base i l'altura d'un triangle i en retorni l'àrea.

Ex. 4. Escriu un programa que demani la base i l'altura d'un quadrat i en retorni l'àrea i el perímetre.

Ex. 5. Escriu un programa que demani el radi i ens doni el volum d'una esfera. Utilitza una constant per π .

Ex. 6. Escriu un programa que demani el nom a l'usuari i el saludi.

Ex. 7. Escriu un programa que li demani a l'usuari una quantitat de diners (C), una taxa d'interès (x) i un nombre d'anys (n). Ha de mostrar com a resultat la quantitat de diners a pagar. El càlcul és:

$$C_n = C \cdot \left(1 + \frac{x}{100}\right)^n$$

Indicació: Hauràs de cercar com fer una potència.

Ex. 8. Escriu un programa que passi de graus Fahrenheit a graus Celsius i a temperatura absoluta (K).

INDICACIONS

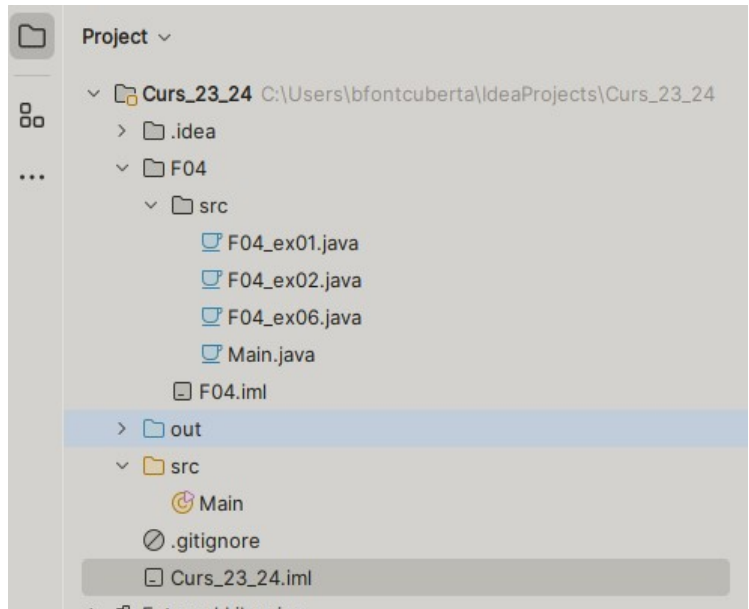
1. L'aspecte que ha de tenir l'IDE que utilitzam ha de ser semblant a la imatge.

Crearem un projecte nou anomenat Curs_23_24

Crearem un mòdul per a cada fitxa de treball.

Cada exercici serà una classe i el nom serà la fitxa i el nombre d'exercicis:

F04_ex01



Quan s'hagi d'entregar un exercici s'ha de penjar el fitxer .java. Aquest es troba a la carpeta:

(En Windows) C:\Users\nom_usuari\IdeaProjects\Curs_23_24\F04\src
(En Ubuntu) \home\nom_usuari\IdeaProjects\Curs_23_24\F04\src

2. Per fer alguns exercicis necessitarem algun mètode de la **classe Math**. Per saber quins mètodes té i quins tipus de dades esperen i retornen és important pegar una ullada als **JavaDocs**.