Una **interfaz** en [Java](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)) es una colección de métodos abstractos y propiedades constantes.

En las interfaces se especifica qué se debe hacer pero no su implementación. Serán las clases que implementen estas interfaces las que describen la lógica del comportamiento de los métodos.

La principal diferencia entre *interface* y *abstract* es que un interface proporciona un mecanismo de encapsulación de los protocolos de los métodos sin forzar al usuario a utilizar la herencia.

El uso de las interfaces Java proporciona las siguientes ventajas:

* Organizar la programación.
* permiten declarar constantes que van a estar disponibles para todas las clases que queramos (implementando esa interfaz)
* Obligar a que ciertas [clases](https://es.wikipedia.org/wiki/Clase_(programaci%C3%B3n)) utilicen los mismos [métodos](https://es.wikipedia.org/wiki/M%C3%A9todo_(programaci%C3%B3n)) (nombres y parámetros).
* Establecer relaciones entre clases que no estén relacionadas.

Uso[[editar](https://es.wikipedia.org/w/index.php?title=Interfaz_(Java)&action=edit&section=2)]

[Java](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)) proporciona dos palabras reservadas para trabajar con interfaces: interface e implements.

Para declarar una interfaz se utiliza:

modificador\_acceso **interface** **NombreInterfaz** {

....

}

modificador\_acceso puede ser una clase de objetos que nos permite utilizar herencia en abstracción constante en las clases en las que se implemente.

Para implementarla en una clase, se utiliza la forma:

modificador\_acceso **class** **NombreClase** **implements** NombreInterfaz1 [, NombreInterfaz2]

Una clase puede implementar varias interfaces de los paquetes que se han importado dentro del programa, separando los nombres por comas.

**Ejemplo**

* Definición de una interfaz:

**interface** **Nave** {

**public** void moverPosicion (int x, int y);

**public** void disparar();

.....

}

* Uso de la interfaz definida:

**public** **class** **NaveJugador** **implements** Nave {

**public** void moverPosicion (int x, int y) {

*//Implementación del método*

posActualx = posActualx - x;

posActualy = posActualy - y;

}

**public** void disparar() {

*//Implementación del método*

}

...

}