

Un tipus de dada compost és aquell que permet emmagatzemar més d'un valor dins una única variable. En aquest cas es tracta d'una estructura en forma de caseller on s'hi poden guardar dades pritives (o bé altres vectors, com veurem). Totes les posicions han de ser del mateix tipus.

Exemple: Vector amb dotze posicions per contitendir valors d'un simulador de dau. Observa que les posicions es comencen a numerar per 0. Per accedir a les diferents posicions ho farem amb el nom del vector i l' índex de la posició: dau[4] → posició 5a del vector "dau".	<table border="1"><tr><td>0</td><td>6</td></tr><tr><td>1</td><td>1</td></tr><tr><td>2</td><td>2</td></tr><tr><td>3</td><td>4</td></tr><tr><td>4</td><td>6</td></tr><tr><td>5</td><td>6</td></tr><tr><td>6</td><td>5</td></tr><tr><td>7</td><td>3</td></tr><tr><td>8</td><td>2</td></tr><tr><td>9</td><td>6</td></tr><tr><td>10</td><td>5</td></tr><tr><td>11</td><td>1</td></tr></table>	0	6	1	1	2	2	3	4	4	6	5	6	6	5	7	3	8	2	9	6	10	5	11	1
0	6																								
1	1																								
2	2																								
3	4																								
4	6																								
5	6																								
6	5																								
7	3																								
8	2																								
9	6																								
10	5																								
11	1																								

Declaració i inicialització

Per poder ho fer hem de:

- indicar quantes posicions tindrà. Aquesta no pot canviar durant l'execució.
- indicar el tipus d'aquestes posicions i inicialitzar-les.

Per **inicialitzar-ho a un valor concret** farem:

```
<tipus>[] nomVariable={valor1, valor2, ... , valorN};
```

Exemple: Si volem declara l'exemple anterior posaríem

```
int[] dau = {6, 1, 2, ... , 5, 1};
```

Si sols el volem declarar sense posar-hi cap valor inicial la forma seria:

```
<tipus>[] nomVariable=new <tipus>[mida];
```

Exemple: Seguint l'exemple anterior quedaria

```
int[] dau = new int[12];
```

En aquest cas la inicialització seria totes les posicions a 0. Seria equivalent a:

```
int[] dau = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
```

Si intentam accedim a un índex que no existeix donarà un error i s'aturarà el programa.

Llargada d'un vector

Per recuperar la llargada d'un array, en Java, teniu un mètode anomenat `length`.

Exemple: Si posam `int llargada= dau.length;` dins `llargada` s'hi posarà 12.

A continuació posarem dos exemples de processos d'entrada molt habituals. Es tracta de posar una gran quantitat d'entrades de teclat dins un array. Fins ara ho havíem fet de forma individual. Recorda que per eliminar les dades que no interessin posàvem `lector.nextLine()`;, això s'anomena "buidar el *buffer* de lectura".

Demanarem que les dades s'introdueixin seguides i separades per, al manco, un espai.

Aquí també plantejarem la possibilitat de comprovar el tipus abans de fer la lectura per evitar errors d'assignació amb les instruccions `lector.hasNext<tipus>`.

Entrada d'una quantitat coneguda des de teclat

Llegeix i analitza el codi que hi ha a continuació.

```
import java.util.Scanner;
public class LecturaSenseValidacio {
    private static final int NUM_VALORS = 10;
    public static void main(String[] args) {
        Scanner lector = new Scanner(System.in);
        System.out.println("Escriu " + NUM_VALORS + " enters. Es pot fer en
diferents línies.");
        //Es llegeixen exactament NUM_VALORS valors.
        int numValorsLlegits = 0;
        while (numValorsLlegits < NUM_VALORS) {
            //Abans de llegir, comprovem si realment hi ha un enter.
            if (lector.hasNextInt()) {
                int valor = lector.nextInt();
                System.out.println("Valor " + numValorsLlegits + " llegit:" +
valor);
                numValorsLlegits++;
            } else {
                //Si el valor no és enter, es llegeix però s'ignora.
                //No s'avança tampoc el comptador.
                lector.next();
            }
        }
        //Els valors que sobrin a la darrera línia escrita es descarten.
        lector.nextLine();
        System.out.println("Ja s'han llegit " + NUM_VALORS + "valors.");
    }
}
```

Entrada d'una quantitat desconeguda des de teclat

Un altre cas, també habitual, és haver de fer una entrada d'un nombre de dades indeterminat.

Llegeix i analitza el codi que hi ha a continuació. Observa que no ho podem assignar a un array ja que aquest tipus de dades ha de tenir una llargada determinada.

```
import java.util.Scanner;

public class LectorValorsDesconeguts {
    public static final int MARCA_FI = -1;
    public static void main (String[] args) {
        Scanner lector = new Scanner(System.in);
        System.out.print("Escriu diferents valors enters.");
        System.out.println("Després del darrer valor escriu un " + MARCA_FI);
        //Es llegeixen exactament NUM_VALORS valors.
        boolean marcaTrobada = false;
        while (!marcaTrobada) {
            //Abans de llegir, comprovem si realment hi ha un enter.
            if (lector.hasNextInt()) {
                int valor = lector.nextInt();
                //És la marca de fi?
                if (valor == MARCA_FI) {
                    //Sí que ho és.
                    marcaTrobada = true;
                } else {
                    //No. És un valor que ha de ser tractat.
                    System.out.println("Valor llegit: " + valor);
                }
            } else {
                //Si el valor no és enter, es llegeix però s'ignora.
                lector.next();
            }
        }
        //Els valors que sobrin a la darrera línia escrita es descarten.
        lector.nextLine();
        System.out.println("Ja s'han llegit tots els valors.");
    }
}
```

Ex. 1. Fes un programa que demani 10 nombres enters. Després demanarà un altre enter i respondrà si aquest valor està dins el vector o no.

Ex. 2. Fes un programa que demani 10 nombres enters. Només hi ha d'haver nombres entre 0 i 50.

Ex. 3. Fes un programa que demani 10 nombres enters. Després demanarà un altre enter i respondrà quants valors estan per damunt i quants estan per davall.

Ex. 4. Fes un programa que demani 10 nombres enters. Respondrà mostrant els valors de les posicions múltiples de 3.

Ex. 5. Fes un programa que ompli un vector de 100 llocs amb nombres aleatoris entre 1 i 6. Ha de calcular la freqüència de cada nombre i mostrar-ho per pantalla.

Ex. 6. Fes un programa que ompli un vector de 12 posicions amb nombres aleatoris entre 1 i 50 sense repeticions.

Ex. 7. Fes un programa que mostri per pantalla els múltiples de 3 que hi hagi. Defineix el vector dins el programa.

Exemples: `int vector[] = { 1, 12, 4, 5, 7, 3, 2, 9, 6, 7};`
`int vector[] = { 2, 13, 4, 5, 7, 6, 2, 8, 5, 7};`

Ex. 8. Fes un programa que agafi dos vectors (poden ser els de l'exercici anterior) i en generi un tercer intercalant els valors de cada posició.

Ex. 9. Fes un programa que demani un nombre enter i que crei un vector d'aquests posicions amb nombres parells. L'ha de mostrar per pantalla.

Ex. 10. Fes un programa que ompli un vector de 15 posicions amb les qualificacions dels alumnes. Has de comprovar que estiguin entre 0 i 10 i ignorar els que no compleixin. En tenir els 15 valors els ha de mostrar per pantalla.

Ex. 11. Fes un programa que comprovi la lletra del DNI. Per calcular-la s'ha de fer el mòdul de 23 i consultar la taula:

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
T	R	W	A	G	H	Y	F	P	D	X	B	N	J	Z	S	Q	V	H	L	C	K	E

Ex. 12. Fes un programa que implementi la llei d'Hondt. El programa ha de demanar el nombre d'escons, el nombre de partits, i els vots obtinguts. Ha de retornar l'assignació d'escons de cada partit.

Per a que tots tinguem el mateix nom anomenau els vectors com:

- `esc[]`. El nombre d'escons
- `vots[]`. Nombre de vots obtinguts.

La llei d'Hondt serveix per assignar escons en unes eleccions. El sistema consisteix en:

- El primer escó s'assigna al partit més votat.
- El següent escó s'assigna al partit que maximitzi l'expressió `vots[]/(esc[]+1)`.
- En cas que dos partits tinguin el mateix valor de vots anterior, l'escó s'assigna al més votat.

Important: Fes unes simulacions prèvies per entendre el mètode i testejar el programa. Per exemple prova-ho amb 3 partits, 7 escons i els vots 90, 40 i 15 respectivament.