



- Simplifica a manipulação de diversas variáveis de um mesmo tipo de dado
- Armazenam um conjunto de itens que tenham o mesmo tipo de informação
- Os dados são acessados através de uma única variável
- São diferenciados e referenciados por um índice numérico

- Estão presentes em praticamente todas as linguagens de programação
- Constituem um dos aspectos mais importantes e facilitadores no desenvolvimento de aplicações

Tipo da variável [] Nome da variável, Nome da variável ;

Ex.:

```
int [ ] notas ;
```

```
double [ ] [ ] vendasPorMes ;
```

- Embora a declaração de um vetor seja idêntico ao de qualquer variável, é necessário que seja reservado a quantidade de itens que poderão ser armazenados no vetor antes de sua utilização

Ex.:

```
int [ ] notas = new int [ 5 ] ;
```

```
double [ ] [ ] vendasPorMes = new double [3] [12] ;
```

- Para referenciar um vetor utilizamos o nome da variável, colchetes e uma literal ou variável que representa a posição dentro do vetor
- Esta posição não pode ultrapassar os limites inferior e posterior do vetor

```
int[ ] notas = new int[5];  
notas[0] = 5;
```

```
int vendas = 0;  
int mes = 0;  
double[ ][ ] vendasPorMes = new double[3][12];  
vendasPorMes[vendas][mes] = 345.93;
```

Estruturas de repetição

- Muitas vezes necessitamos de percorrer um vetor para efetuar uma operação
- Poderemos utilizar qualquer uma das três estruturas de repetição para efetuar qualquer operação.
- Normalmente utilizamos uma variável para controlar a posição no vetor
- Esta variável deve ser inicializada e incrementada a fim para que seja possível endereçar todos os itens do vetor

- A variável “i” representa o ponteiro do vetor

```
int[ ] notas = new int[5];  
  
int i = 0;  
while(i < notas.length) {  
    notas[i] = leInteiro("Informe a ", i+1, "ª nota");  
    i++;  
}
```

- Será necessário inverter o teste condicional ao se utilizar repita

```
int[ ] notas = new int[5];  
  
int i = 0;  
do {  
    notas[i] = leInteiro("Informe a ", i+1, "ª nota");  
    i++;  
} while(i < notas.length);
```


- A estrutura de repetição “for” é a mais simples de se utilizar para estas operações

```
int[ ] notas = new int[5];
```

```
for(int i = 0; i < notas.length; i++) {  
    notas[i] = leInteiro("Informe a ", i+1, "ª nota");  
}
```

Vetor bidimensional

- Num vetor bidimensional necessitamos de dois índices para referenciarmos um elemento do vetor
- Cada índice endereça uma dimensão no vetor

```
double[ ][ ] vendasPorMes = new double[3][12];

for (int i = 0; i < vendasPorMes.length; i++) {
    for (int j = 0; j < vendasPorMes[i].length; j++) {
        vendasPorMes[i][j] = leReal("Informe o ", i+1,
                                     "° valor de vendas para o ", j+1, "° mês");
    }
}
```