

# New (Operador)

29/05/2022

Francisco Mendoza

El operador "New" sirve para reservar memoria dinámica.

30604878

Sintaxis : [::] new [<emplazamiento>] <Tipo> [<inicializacion>]

[::] new [<emplazamiento>](<Tipos>) [<inicializacion>]

[::] new [<emplazamiento>] <Tipo> [<numero\_elementos>]

[::] new [<emplazamiento>](<Tipos>) [<numero\_elementos>]

El operador opcional :: esta relacionada con la sobrecarga de operadores,

la inicializa, si aparece, se usara para asignar valores iniciales a la memoria reservada con new, pero no puede ser usada con arrays

la Tercera y Cuarta forma, se usan para reservar espacio en memoria para arrays dinamicos.

La memoria reservada con new sera valida hasta que se libere con delete o hasta el fin del programa, aunque es aconsejable liberar siempre la memoria reservada con new usando delete.

Si la reserva de memoria no tuvo éxito, new devuelve un puntero nulo, NULL.

## Delete (Operador)

El operador delete se utiliza para liberar espacio en la memoria dinámica reservada con el new.

Sintaxis.

delete [<expresión>] [::]

delete [::] [<expresión>]

La expresión sera normalmente un puntero el operador delete[] se usa para liberar memoria de arrays dinámicos.

Es importante liberar siempre usando delete la memoria reservada con new. Existe el peligro de perdida de memoria si se ignora estas reglas.

Cuando se utiliza el operador delete con un puntero nulo, no se realiza ninguna acción. Esto permite utilizar el operador delete con punteros sin necesidad de preguntar si es nulo antes.

```
#include <iostream>
```

```
using namespace std;
```

```
int main () {
```

```
    int suma = 0;
```

```
    int n;
```

```
    int *p;
```

```
    cout << "Introduzca el numero de elementos:";
```

```
    cin >> n;
```

```
    p = new int [n];
```

```
    for (int i = 0; i < n; i++) {
```

```
        cout << "Introduzca elemento" << i + 1 << ":";
```

```
        cin >> p[i];
```

```
        suma += p[i];
```

```
}
```

```
    cout << "Elementos introducidos: ";
```

```
    for (int i = 0; i < n; i++)
```

```
        cout << p[i] << ";"
```

```
    cout << "Total:" << suma << endl;
```

```
    cout << "Media:" << (double) suma / n << endl;
```

```
    delete [] p;
```

```
    return 0;
```

```
}
```