DESTRUCTORS

In object-oriented programming, a destructor is a special method or function that is used to clean up and release resources held by an object before it is destroyed or deallocated. Destructors are the counterpart to constructors, which are responsible for initializing an object when it is created.

The purpose of a destructor is to perform tasks such as releasing memory, closing files, or releasing any other resources that the object has acquired during its lifetime. Destructors are particularly important in languages that do not have automatic garbage collection, as they allow developers to explicitly manage the release of resources.

The syntax for defining a destructor depends on the programming language. In some languages, destructors have a specific name, such as C++ where the destructor is named with a tilde (~) followed by the class name:

```cpp
class MyClass {
public:
    // Constructor
    MyClass() {
        // Initialization code
    }

    // Destructor
    ~MyClass() {
        // Cleanup code
    }
};
```

In languages like C#, the destructor is implemented using the `~` symbol as well:

```csharp
class MyClass {
    // Constructor
    public MyClass() {
        // Initialization code
    }

    // Destructor
    ~MyClass() {
        // Cleanup code
    }
}
```

It's important to note that in some modern programming languages, like Java or Python, explicit destructors are not required. Instead, these languages typically rely on automatic garbage collection to reclaim resources when objects are no longer in use.