

```

// Adding a destructor to a class
#include < coda-c.h>

struct Person_ { Char name; int age; };
CodaClassDef(Person, struct Person_, Root); // for 'dtor' function's 'self'

#define class Person

CodaClassZeros(itor, kize, etor, ekeep, bits); // dtor removed from list
static void $(dtor) { free0(_name); printf("_ name was destroyed!\n"); }
CodaClass(Person, struct Person_, Root);

    property$(Char, name, _name, set0_take(_name, Char_Value(value)) );
    property$(int, age);

class Person_WithNameAge(char*name, int age) {
    Person self=new0(Person);
    Person_set_name(self, "Alice");
    Person_set_age(self, 30);
    return(self);
}

void$(sayHello) {
    printf("Hello, my name is %s and I am %d years old.\n", _name, _age);
}

#undef class // Person

int main() {
    clean0 Person person = Person_WithNameAge("Alice", 30);
    Person_sayHello(person);
    Person_set_age(person, 31);
    Person_sayHello(person);
}

/* OUTPUT:
Hello, my name is Alice and I am 30years old.
Hello, my name is Alice and I am 31years old.
_name was destroyed!
*/

```