```c
#include <stdio.h>
#include <stdlib.h>

#include "author.h"
#include "utils.h"


author_t* author_new() {

    author_t* toto ;
    toto = malloc ( sizeof(author_t) );
    if ( toto == NULL ) {
        throw_error("malloc failed", 1 ) ;
        return NULL ;
    }
    toto->lastname = malloc ( STRING_SIZE ) ;
    if ( toto->lastname == NULL ) {
        throw_error("malloc failed", 1 ) ;
        return NULL ;
    }
    toto->firstname = malloc ( STRING_SIZE );
    if ( toto->firstname == NULL ) {
        throw_error("malloc failed", 1 ) ;
        return NULL ;
    }
    toto->email = malloc ( STRING_SIZE );
    if ( toto->email == NULL ) {
        throw_error("malloc failed", 1 ) ;
        return NULL ;
    }
    sprintf(toto->lastname,     "Doe") ;
    sprintf(toto->firstname, "John") ;
    sprintf(toto->email,    "lepape@vatican.org") ;
    return toto;
}

void author_del ( author_t* author_a_dell ) {
    free (author_a_dell->lastname) ;
    free (author_a_dell->firstname) ;
    free (author_a_dell->email) ;
    free (author_a_dell) ;
}

void author_show (author_t* author_affiche ) {

    printf("Auteur : %s %s \nEmail : %s\n", author_affiche->lastname,
                                     author_affiche->firstname,
                                     author_affiche->email);

}

/////////////NOM/////////////////
void author_set_lastname ( author_t* author_a_set, char* lastname) {
    sprintf(author_a_set->lastname,"%s",lastname); // Méthode pour éviter une
 fuite mémoire : je prend la chaine et la recopie dans le tableau déjà allouÃ
©
    // author_a_set->lastname = lastname; //Fuite mémoire : réécrase adresse m
émoire donc Malloc du début perdu, pas grave mais à  EVITER
}

char* author_get_lastname (author_t* author_a_get) {
    return author_a_get->lastname;
}
```

```c
/////////////PRENOM/////////////////
void author_set_firstname (author_t* author_a_set, char* firstname) {
    sprintf(author_a_set->firstname,"%s",firstname);
}

char* author_get_firstname (author_t* author_a_get) {
    return author_a_get->firstname;
}

/////////////EMAIL/////////////////
void author_set_email (author_t* author_a_set, char* email) {
    sprintf(author_a_set->email,"%s",email);
}

char* author_get_email (author_t* author_a_get) {
    return author_a_get->email;
}
```