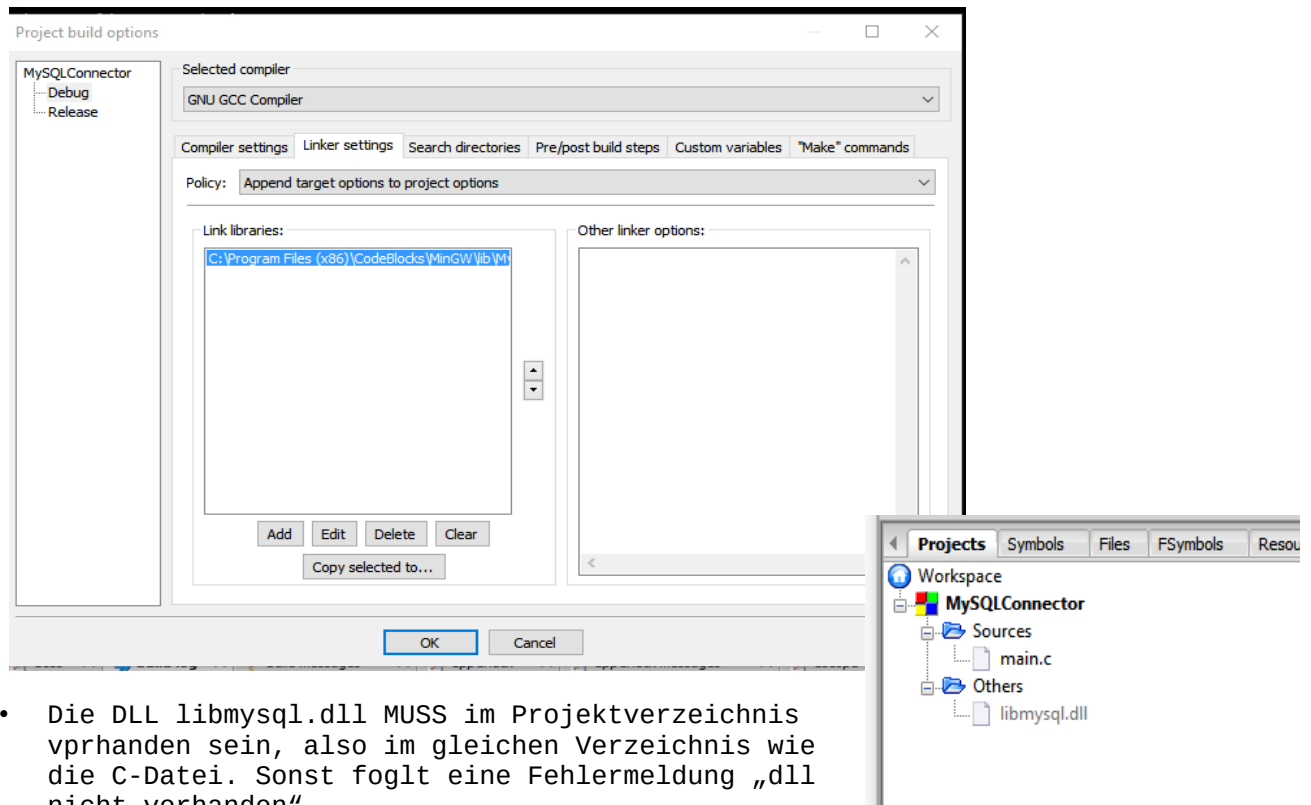


## Installation des MySQL Connectors für C und die CodeBlocks-IDE

- MySQL Connector Libraries downloaden unter:  
<http://dev.mysql.com/downloads/file/?id=456189>
- Der Connector wird unter C:\Program Files (x86)\MySQL .. installiert
- Den Inhalt der Verzeichnisse bin, include, lib in die entsprechenden Verzeichnisse von CodeBlocks kopieren
- In CodeBlocks im Menü Projekt unter Build Options im Reiter Linker settings den Pfad zu den Linker Libraries angeben -> hier z.B. C:\Program Files (x86)\MySQL



- Die DLL libmysql.dll MUSS im Projektverzeichnis vorhanden sein, also im gleichen Verzeichnis wie die C-Datei. Sonst folgt eine Fehlermeldung „dll nicht vorhanden“

## Beispielprojekt (quick an dirty): Siehe auch Rheinwerk Computing Tutorial

```
#include <stdio.h>
#include <mysql.h>
// globale Variable (dirty)
MYSQL *con;
// Funktionsprototypen
void check_error(void);
int main()
{
    // lokale Variablen
    int num_fields = 0, i = 0;
    con = mysql_init(NULL);
    if(con == NULL)
    {
        fprintf(stderr, "Initialisierung fehlgeschlagen\n");
        exit (EXIT_FAILURE);
    }
    /* MYSQL *mysql_real_connect( MYSQL *mysql,
                                   const char *host,
                                   const char *user,
                                   const char *password,
                                   const char *db,
                                   unsigned int port,
                                   const char *unix_socket,
                                   unsigned int client_flag );
    */
    mysql_real_connect(con,"localhost","root",NULL,"meineDB",0,NULL,0);
    check_error();
    printf("Client version: %s",mysql_get_client_info());
    printf("\nServer version: %s",mysql_get_server_info(con));
    if (mysql_query(con, "SELECT * FROM irgendwas"))
        check_error();
    MYSQL_RES *result = mysql_store_result(con);
    check_error();
    num_fields = mysql_num_fields(result);

    MYSQL_ROW row;

    // Die gefundenen Datensätze ausgeben
    while ((row = mysql_fetch_row(result)))
    {
        for(i = 0; i < num_fields; i++)
        {
            printf("\n%s ", row[i] ? row[i] : "NULL");
        }
        printf("\n");
    }
    mysql_free_result(result);
    mysql_close(con);
    return 1;
}
// Funktionsdefinitionen
/* Bricht bei Fehler (mysql_error != 0) das Programm ab. */
void check_error(void)
{
    if (mysql_errno(con) != 0)
    {
        fprintf(stderr, "Fehler: %s\n", mysql_error(con));
        // printf("Es ist ein Fehler aufgetreten!!\n");
        exit(EXIT_FAILURE);
    }
}
```