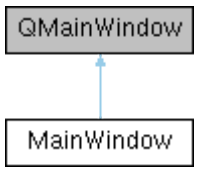
MainWindow Class Reference

The **MainWindow** class this is the main window of all UIs and carry most of the important functions to run the program. More...

`#include <mainwindow.h>`

Inheritance diagram for MainWindow:



Public Slots

void	**onVectorReceived** (QVector< **admin** > one_ad)
void	**recieveProfData** (QVector< **profsCLASS** * > one_prof) recieveProfData,setdata_in_profs_table these two functions recieves the data of the created prof, stores it in our database and sets his basic information in a table
void	**setdata_in_profs_table** ()
void	**recieveStudData** (QVector< **studentsClass** * > one_stud) recieveStudData,setdata_in_studs_table these two functions recieves the data of the created student, stores it in our database and sets his basic information in a table
void	**setdata_in_studs_table** ()
void	**recieveCourseData** (QVector< **course** * > vector) recieveCourseData,setdata_in_classes_table these two function recieves the data of the newly created course, stores it in our database and sets his basic information in a table
void	**setdata_in_classes_table** ()
void	**recieveListElements** (QVector< QString > listElements) recieveListElements this function recives list elements carrying the chosen courses to be added to a specefic student
void	**setCoursesToStudent** () setCoursesToStudent set chosen course in student data, the tables and set the student in the course
void	**recieve_new_myClasses** (int ind, QVector< QPair< QString, QString > > newC) recieve_new_myClasses,update_my_classes_table these functions recieve the classes that their grades have been added to them in the students vector and use these vectors to update the tables
void	**update_my_classes_table** (int indxxx)
void	**recieveListElementsP** (QVector< QString > listElementsProf) recieveListElementsP,setCoursesToProf these functions recieve list elements carrying the courses chosen to be assigned to the specefic professor
void	**setCoursesToProf** ()

Signals

void **sendCoursesData** (QVector< **course** * > all)
sendCoursesData when this signal is emmited it sends all courses to add new student to the add new course window

void **sendStudentName** (QString)
sendStudentName,sendStudentIndex these two signals when emmited send the chosen student name and index to assigna courses to them

void **sendStudentIndex** (int indx)

void **sendAllClasses** (QVector< **course** * > all)
sendAllClasses when this signal is emmited, the window to add courses to that class recieves the data of all classes

void **sendMyClasses** (QVector< QPair< QString, QString > > all)
sendMyClasses when emmited it sends all classes to the grades window

void **please_set_data** ()
please_set_data,please_set_dataP when emmited set the data in courses table in add course to student/prof window

void **please_set_dataP** ()

void **please_set_my_classes** (QVector< QPair< QString, QString > > my)
please_set_my_classes when emmited set the data in courses table in set grade window

void **sendAllProfs** (QVector< **profsCLASS** * > allProfs)
sendAllProfs,sendAllStds when emmited send aleardy found data of students/profs tp add new prof/student

void **sendAllStds** (QVector< **studentsClass** * > allStds)

void **sendPNameAndTitle** (QString name, QString tit)

Public Member Functions

MainWindow (QWidget *parent=nullptr)

Public Attributes

QVector< **course** * > **allClasses**
alldata structures data structues to store all what we have createdd in our program like a (database) access these vectors carry pointer to objects

QVector< **admin** > **allAdmins**

QVector< **profsCLASS** * > **allProfs**

QVector< **studentsClass** * > **allStudents**

QVector< QString > **listElement**
listElement list of classes added to the chosen student

QVector< QString > **listElementP**
listElementP lise of all classes added to a chosen prof

int **stddd_index**

stddd_index the index of the student that was double clicked

Detailed Description

The **MainWindow** class this is the main window of all UIs and carry most of the important functions to run the program.

Member Function Documentation

◆ please_set_my_classes

void MainWindow::please_set_my_classes (QVector< QPair< QString, QString > > my) signal

please_set_my_classes when emmited set the data in courses table in set grade window

Parameters

my

◆ recieve_new_myClasses

void MainWindow::recieve_new_myClasses (int ind,
QVector< QPair< QString, QString > > newC
) slot

recieve_new_myClasses,update_my_classes_table these functions recieve the classes that their grades have been added to them in the students vector and use these vectors to update the tables

Parameters

ind

newC

◆ recieveCourseData

void MainWindow::recieveCourseData (QVector< **course** * > vector) slot

recieveCourseData,setdata_in_classes_table these two function recieves the data of the newly created course, stores it in our database and sets his basic information in a table

Parameters

vector

◆ recieveListElements

void MainWindow::recieveListElements (QVector< QString > listElements) slot

recieveListElements this function recives list elements carrying the chosen courses to be added to a specefic student

Parameters

listElements

◆ recieveListElementsP

void MainWindow::recieveListElementsP (QVector< QString > listElementsProf) slot

recieveListElementsP,setCoursesToProf these functions recieve list elements carrying the courses chosen to be assigned to the specefic professor

Parameters

listElementsProf

◆ recieveProfData

void MainWindow::recieveProfData (QVector< **profsCLASS** * > one_prof) slot

recieveProfData,setdata_in_profs_table these two functions recieves the data of the created prof, stores it in our database and sets his basic information in a table

Parameters

one_prof

◆ recieveStudData

void MainWindow::recieveStudData (QVector< **studentsClass** * > one_stud) slot

recieveStudData,setdata_in_studs_table these two functions recieves the data of the created student, stores it in our database and sets his basic information in a table

Parameters

one_stud

◆ sendAllClasses

void MainWindow::sendAllClasses (QVector< **course** * > all) signal

sendAllClasses when this signal is emmited, the window to add courses to that class recieves the data of all classes

Parameters

all

◆ sendAllProfs

void MainWindow::sendAllProfs (QVector< **profsCLASS** * > allProfs) signal

sendAllProfs,sendAllStds when emmited send aleardy found data of students/profs tp add new prof/student

Parameters

allProfs

◆ sendCoursesData

void MainWindow::sendCoursesData (QVector< **course** * > all) signal

sendCoursesData when this signal is emmited it sends all courses to add new student to the add new course window

Parameters

all

◆ sendMyClasses

void MainWindow::sendMyClasses (QVector< QPair< QString, QString > > all) signal

sendMyClasses when emmited it sends all classes to the grades window

Parameters

all

The documentation for this class was generated from the following files:

- **mainwindow.h**
- mainwindow.cpp