SmartCalc

Generated by Doxygen 1.9.1

1 Namespace Index 1

1.1 Namespace List . . . 1

2 Hierarchical Index 2

2.1 Class Hierarchy . . . 2

3 Class Index 2

3.1 Class List . . . 2

4 File Index 3

4.1 File List . . . 3

5 Namespace Documentation 4

5.1 s21 Namespace Reference . . . 4

5.1.1 Detailed Description . . . 4

6 Class Documentation 4

6.1 s21::deposit::arg Struct Reference . . . 4

6.1.1 Detailed Description . . . 5

6.2 s21::model::arg Struct Reference . . . 5

6.2.1 Detailed Description . . . 5

6.3 s21::Calculator Class Reference . . . 6

6.3.1 Detailed Description . . . 8

6.3.2 Constructor & Destructor Documentation . . . 8

6.3.3 Member Function Documentation . . . 8

6.4 s21::controller Class Reference . . . 13

6.4.1 Detailed Description . . . 14

6.4.2 Constructor & Destructor Documentation . . . 14

6.4.3 Member Function Documentation . . . 15

6.5 s21::credit Class Reference . . . 17

6.5.1 Detailed Description . . . 18

6.5.2 Constructor & Destructor Documentation . . . 18

6.5.3 Member Function Documentation . . . 19

6.6 s21::model::DateHolder Struct Reference . . . 19

6.6.1 Detailed Description . . . 20

6.6.2 Constructor & Destructor Documentation . . . 20

6.6.3 Member Function Documentation . . . 21

6.7 s21::deposit Class Reference . . . 24

6.7.1 Detailed Description . . . 26

6.7.2 Constructor & Destructor Documentation . . . 26

6.7.3 Member Function Documentation . . . 26

6.8 s21::Graph Class Reference . . . 29

6.8.1 Detailed Description . . . 30

6.8.2 Constructor & Destructor Documentation . . . 30

6.8.3 Member Function Documentation . . . 31
6.9 s21::deposit::lists Struct Reference . . . 32
6.9.1 Detailed Description . . . 32
6.10 s21::model::lists Struct Reference . . . 32
6.10.1 Detailed Description . . . 33
6.11 s21::model Class Reference . . . 33
6.11.1 Detailed Description . . . 35
6.11.2 Member Function Documentation . . . 35
6.12 s21::deposit::result Struct Reference . . . 39
6.12.1 Detailed Description . . . 40
6.13 s21::model::result Struct Reference . . . 40
6.13.1 Detailed Description . . . 40
6.14 s21::model::unit Struct Reference . . . 40
6.14.1 Detailed Description . . . 41
6.14.2 Member Function Documentation . . . 41

7 File Documentation **42**
7.1 sources/calc-insertion.cc File Reference . . . 42
7.1.1 Detailed Description . . . 42
7.2 sources/calc.cc File Reference . . . 43
7.2.1 Detailed Description . . . 43
7.3 sources/controller.cc File Reference . . . 43
7.3.1 Detailed Description . . . 44
7.4 sources/credit.cc File Reference . . . 44
7.4.1 Detailed Description . . . 44
7.5 sources/deposit.cc File Reference . . . 44
7.5.1 Detailed Description . . . 45
7.6 sources/graph.cc File Reference . . . 45
7.6.1 Detailed Description . . . 45
7.7 sources/model-financial.cc File Reference . . . 46
7.7.1 Detailed Description . . . 46
7.8 sources/model.cc File Reference . . . 46
7.8.1 Detailed Description . . . 47

Index **49**

1 Namespace Index

1.1 Namespace List

Here is a list of all documented namespaces with brief descriptions:

s21
Standard namespace for school project **4**

2 Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

s21::deposit::arg 4

s21::model::arg 5

s21::model::DateHolder 19

s21::deposit::lists 32

s21::model::lists 32

s21::model 33
QMainWindow

- **s21::Calculator** 6
- **s21::credit** 17
- **s21::deposit** 24
  QObject
- **s21::controller** 13
  QWidget
- **s21::Graph** 29

s21::deposit::result 39

s21::model::result 40

s21::model::unit 40

3 Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

s21::deposit::arg
Struct to keep all info for deposit calculation 4

s21::model::arg
Keeps all information to provide deposit calculation 5

s21::Calculator
Main window class 6

s21::controller
Implements the MVC pattern 13

s21::credit
Class for credit window widget 17

s21::model::DateHolder
Class that keeps dates, simple version of QDate 19

s21::deposit 24

s21::Graph
Class for graph window widget 29

s21::deposit::lists
Struct to keep date and amount of money 32

s21::model::lists
Stores date and amount of money 32

s21::model
Main class for calculations 33

s21::deposit::result
Struct to keep result info from deposit calculation 39

s21::model::result
Keeps result information after deposit calculation 40

s21::model::unit
Internal struct for rpn and calculation 40

4 File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

include/calc.h ??

include/controller.h ??

include/credit.h ??

include/deposit.h ??

include/graph.h ??

include/model.h ??

sources/calc-insertion.cc 42

sources/calc.cc 43

sources/controller.cc 43

sources/credit.cc 44

sources/deposit.cc 44

sources/graph.cc 45

sources/main.cc **??**

sources/model-financial.cc **46**

sources/model.cc **46**

5 Namespace Documentation

5.1 s21 Namespace Reference

standard namespace for school project

Classes

- class Calculator
  Main window class.
- class controller
  implements the MVC pattern
- class credit
  Class for credit window widget.
- class deposit
- class Graph
  Class for graph window widget.
- class model
  main class for calculations

5.1.1 Detailed Description

standard namespace for school project

6 Class Documentation

6.1 s21::deposit::arg Struct Reference

Struct to keep all info for deposit calculation.

`#include <deposit.h>`

Public Attributes

- QDate **start_date**
- double **deposit_amount** {}
- long long **term** {}
- double **rate** {}
- double **tax_rate** {}
- int **periodicity** {}
- bool **capitalization** {}
- vector **replenishments** {}
- vector **withdrawals** {}

6.1.1 Detailed Description

Struct to keep all info for deposit calculation.

Definition at line 46 of file deposit.h.

The documentation for this struct was generated from the following file:

- include/deposit.h

6.2 s21::model::arg Struct Reference

Keeps all information to provide deposit calculation.

`#include <model.h>`

Public Attributes

- std::tm **date** {}
- double **deposit_amount** {}
- long long **term** {}
- double **rate** {}
- double **tax_rate** {}
- int **periodicity** {}
- bool **capitalization** {}
- lst_vector **replenishments**
- lst_vector **withdrawals**

6.2.1 Detailed Description

Keeps all information to provide deposit calculation.

Definition at line 151 of file model.h.

The documentation for this struct was generated from the following file:

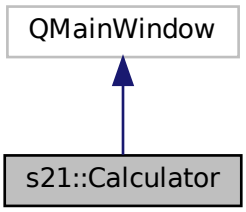
- include/model.h

6.3 s21::Calculator Class Reference

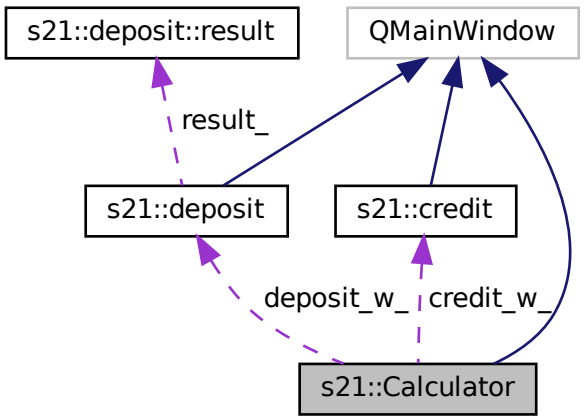
Main window class.

`#include <calc.h>`

Inheritance diagram for s21::Calculator:



Collaboration diagram for s21::Calculator:



Signals

- void **EvaluateExpression** (const QString &expression, const double &x)
- void **GraphCreated** (Graph &graph)

Public Member Functions

- Calculator (QWidget *parent=nullptr)
- ~Calculator () override
- void SetResult (double &)
- void SetError (const QString &message)
- credit * GetCredit ()
- deposit * GetDeposit ()

Private Slots

- void **on_Appended_buttonClicked** (QAbstractButton *button)
- void **on_Button_c_clicked** ()
- void **on_Button_backspace_clicked** ()
- void **on_Button_eq_clicked** ()
- void **on_Button_xc_clicked** ()
- void **on_Button_xs_clicked** ()
- void **on_Button_graph_clicked** ()
- void **on_actionCredit_triggered** ()
- void **on_actionDeposit_triggered** ()
- void **Close** ()

Private Member Functions

- void ComplementInsertion (const QString &button_text)
- QString ProvideInsertionLogic (QString &new_button_text) const
- bool ShouldSkipOperatorInsertion () const
- bool CheckIfAlreadyContainPoint () const
- bool ShouldAddMultSymbol (const QString &new_button_text) const
- void DeleteOneToken ()
- qsizetype FindPositionToTerminate () const
- QString ConvertExponentToDec () const
- bool Evaluate (const double &x=0.0)
- void SetStatusOK ()
- void ClearIfNeeded ()

Static Private Member Functions

- static bool IsDigit (const QChar &ch) noexcept

Private Attributes

- Ui::Calculator * **ui** {}
- double **x_** = NAN
- double **result_** = NAN
- bool **clear_screen_** = true
- bool **error** = false
- credit * **credit_w_**
- deposit * **deposit_w_**

6.3.1 Detailed Description

Main window class.

Definition at line 32 of file calc.h.

6.3.2 Constructor & Destructor Documentation

6.3.2.1 Calculator() `s21::Calculator::Calculator ( QWidget * parent = nullptr ) [explicit]`

Base constructor with corresponding connects to child windows

Parameters

parent	- QObject dependency

Definition at line 7 of file calc.cc.

6.3.2.2 ~Calculator() `s21::Calculator::~Calculator ( ) [override]`

Destructor

Definition at line 22 of file calc.cc.

6.3.3 Member Function Documentation

6.3.3.1 CheckIfAlreadyContainPoint() `bool s21::Calculator::CheckIfAlreadyContainPoint ( ) const [private]`

Checks if last number in expression already contains a point

Returns

bool

Definition at line 70 of file calc-insertion.cc.

6.3.3.2 ClearIfNeeded() `void s21::Calculator::ClearIfNeeded ( ) [private]`

clears expression if needed and drops the flag

Definition at line 29 of file calc.cc.

6.3.3.3 ComplementInsertion() `void s21::Calculator::ComplementInsertion ( const QString & button_text ) [private]`

Main function to provide insertion

Parameters

button_text	- text on a button

Definition at line 9 of file calc-insertion.cc.

6.3.3.4 ConvertExponentToDec() `QString s21::Calculator::ConvertExponentToDec ( ) const [private]`

Convert scientific exponent to dec representation

Returns

transformed string

Definition at line 92 of file calc.cc.

6.3.3.5 DeleteOneToken() `void s21::Calculator::DeleteOneToken ( ) [private]`

Provides pne token deletion

Definition at line 59 of file calc-insertion.cc.

6.3.3.6 Evaluate() `bool s21::Calculator::Evaluate ( const double & x = 0.0 ) [private]`

Provides evaluation signal and error's handling

Parameters

x	

Returns

Definition at line 106 of file calc.cc.

6.3.3.7 FindPositionToTerminate() `qsizetype s21::Calculator::FindPositionToTerminate ( ) const [private]`

Finds position to cut off a string

Returns

qsizetype

Definition at line 99 of file calc-insertion.cc.

6.3.3.8 GetCredit() `credit * s21::Calculator::GetCredit ( )`

Public getter for controller to get child window

Returns

pointer to child window

Definition at line 123 of file calc.cc.

6.3.3.9 GetDeposit() `deposit * s21::Calculator::GetDeposit ( )`

Public getter for controller to get child window

Returns

pointer to child window

Definition at line 125 of file calc.cc.

6.3.3.10 IsDigit() `bool s21::Calculator::IsDigit ( const QChar & ch ) [static], [private], [noexcept]`

Checks if symbol is Digit

Parameters

ch	- symbol to check

Returns

bool

Definition at line 95 of file calc-insertion.cc.

6.3.3.11 ProvideInsertionLogic() `QString s21::Calculator::ProvideInsertionLogic ( QString & new_button_text ) const [private]`

Function that provide base insertion logic

Parameters

new_button_text	- modifying text from a button

Definition at line 18 of file calc-insertion.cc.

6.3.3.12 SetError() `void s21::Calculator::SetError ( const QString & message )`

Public method for controller to set error

Parameters

message	exception message

Definition at line 113 of file calc.cc.

6.3.3.13 SetResult() `void s21::Calculator::SetResult ( double & result )`

Public method for controller to set result

Definition at line 104 of file calc.cc.

6.3.3.14 SetStatusOK() `void s21::Calculator::SetStatusOK ( ) [private]`

Set error to false

Definition at line 118 of file calc.cc.

6.3.3.15 ShouldAddMultSymbol() `bool s21::Calculator::ShouldAddMultSymbol ( const QString & new_button_text ) const [private]`

Checks if multiplication symbol should be added

Returns

bool

Definition at line 86 of file calc-insertion.cc.

6.3.3.16 ShouldSkipOperatorInsertion() `bool s21::Calculator::ShouldSkipOperatorInsertion ( ) const [private]`

Checks if operator insertion should be skipped

Returns

bool

Definition at line 79 of file calc-insertion.cc.

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

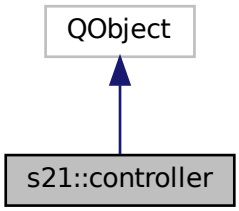
- include/calc.h
- sources/calc-insertion.cc
- sources/calc.cc

6.4 s21::controller Class Reference

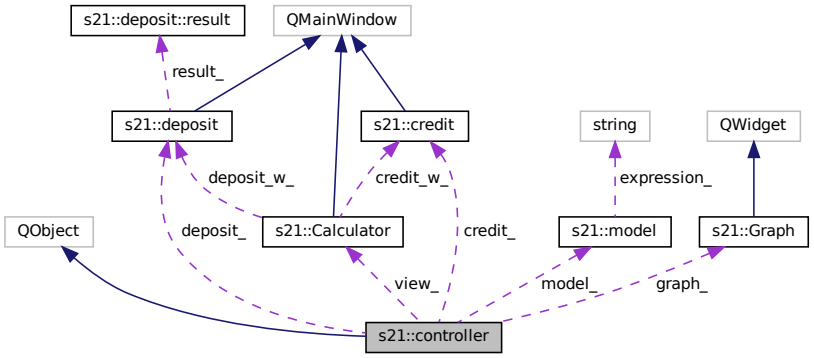
implements the MVC pattern

`#include <controller.h>`

Inheritance diagram for s21::controller:



Collaboration diagram for s21::controller:



Public Member Functions

- controller (model ∗model, Calculator ∗view, QObject ∗parent=nullptr)

Private Slots

- void ConnectGraph (Graph &graph)
- void CalculateExpression (const QString &expression, const double &x) const
- void PlotCalculation (const double &x) const
- void SetPrecision (const double &precision)
- void CalculateCredit (const double &loan_amount, const unsigned &loan_term, const double &interest_rate, const bool &annuity) const
- void CalculateDeposit (const deposit::arg &argument) const
- static model::lst_vector ConvertArguments (const deposit::vector &vector)

Private Attributes

- model ∗ **model_**
- Calculator ∗ **view_**
- Graph ∗ **graph_** = nullptr
- credit ∗ **credit_**
- deposit ∗ **deposit_**

6.4.1 Detailed Description

implements the MVC pattern

Definition at line 23 of file controller.h.

6.4.2 Constructor & Destructor Documentation

6.4.2.1 controller()

```
s21::controller::controller (
        model * model,
        Calculator * view,
        QObject * parent = nullptr )
```

Constructor. Get pointers to model and view(MVC). Connects their signals to self slots.

Parameters

model	- pointer to model
view	- pointer to ciew
parent	- parent fot QObject

Definition at line 10 of file controller.cc.

6.4.3 Member Function Documentation

6.4.3.1 CalculateCredit

```
void s21::controller::CalculateCredit (
        const double & loan_amount,
        const unsigned & loan_term,
        const double & interest_rate,
        const bool & annuity ) const  [private], [slot]
```

Slot connected to credit calculation signal

Parameters

loan_amount	
loan_term	
interest_rate	
annuity	

Definition at line 48 of file controller.cc.

6.4.3.2 CalculateDeposit

```
void s21::controller::CalculateDeposit (
        const deposit::arg & argument ) const  [private], [slot]
```

Slot connected to deposit calculation signal

Parameters

argument	

Definition at line 57 of file controller.cc.

6.4.3.3 CalculateExpression

```
void s21::controller::CalculateExpression (
        const QString & expression,
        const double & x ) const  [private], [slot]
```

Slot connected to expression calculation signal from main window

Parameters

expression	- to calculate
x	

Definition at line 26 of file controller.cc.

6.4.3.4 ConnectGraph

`void s21::controller::ConnectGraph ( Graph & graph ) [private], [slot]`

slot used to connect graph's signal to calculate an expression with corresponding X and calculation slot. Initializes graph_ pointer

Parameters

graph	- graph to get

Definition at line 37 of file controller.cc.

6.4.3.5 ConvertArguments

`model::lst_vector s21::controller::ConvertArguments ( const deposit::vector & vector ) [static], [private], [slot]`

Method to convert view::repl/with lists to model::repl/with lists

Parameters

vector	- view::repl/with

Returns

model::repl/with lists

Definition at line 80 of file controller.cc.

6.4.3.6 PlotCalculation

`void s21::controller::PlotCalculation ( const double & x ) const [private], [slot]`

Slot connected to plot calculation signal

Parameters

x	

Definition at line 44 of file controller.cc.

6.4.3.7 SetPrecision

`void s21::controller::SetPrecision ( const double & precision ) [private], [slot]`

Slot connected to precision set signal

Parameters

precision	

Definition at line 91 of file controller.cc.

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

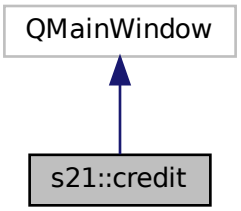
- include/controller.h
- sources/controller.cc

6.5 s21::credit Class Reference

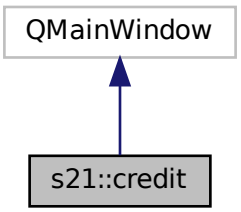
Class for credit window widget.

`#include <credit.h>`

Inheritance diagram for s21::credit:



Collaboration diagram for s21::credit:



Public Types

- using **vector** = QVector< double >

Signals

- void **ShowMain** ()
- void **ShowSecondWindow** ()
- void **Close** ()
- void **CalculateCredit** (const double &loan_amount, const unsigned &loan_term, const double &interest_rate, const bool &annuity)

Public Member Functions

- credit (QMainWindow *parent=nullptr)
- ~credit () override
- void SetCredit (vector &&result)

Private Slots

- void **on_actionBasic_triggered** ()
- void **on_actionDeposit_triggered** ()
- void **on_calculate_clicked** ()

Private Member Functions

- double CalculateTotalPayment ()

Private Attributes

- Ui::credit * **ui**
- vector **credit_**

6.5.1 Detailed Description

Class for credit window widget.

Definition at line 23 of file credit.h.

6.5.2 Constructor & Destructor Documentation

6.5.2.1 credit() `s21::credit::credit (QMainWindow * parent = nullptr ) [explicit]`

Constructor. Set ui and connects.

Parameters

parent	

Definition at line 13 of file credit.cc.

6.5.2.2 ~credit() `s21::credit::~credit ( ) [override]`

Destructor

Definition at line 19 of file credit.cc.

6.5.3 Member Function Documentation

6.5.3.1 CalculateTotalPayment() `double s21::credit::CalculateTotalPayment ( ) [private]`

Summarize payments from months

Returns

sum

Definition at line 46 of file credit.cc.

6.5.3.2 SetCredit() `void s21::credit::SetCredit ( credit::vector && result )`

Public method for controller to set result

Parameters

result	

Definition at line 59 of file credit.cc.

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

- include/credit.h
- sources/credit.cc

6.6 s21::model::DateHolder Struct Reference

Class that keeps dates, simple version of QDate.

`#include <model.h>`

Public Member Functions

- DateHolder () noexcept
  construct current date
- DateHolder (int day, int month, int year) noexcept
- DateHolder (std::tm other) noexcept
- bool Compare (const DateHolder &other) const noexcept
- size_t DaysInMonth () const noexcept
- size_t DaysInYear () const noexcept
- size_t ToDay (const DateHolder &other) const noexcept
- DateHolder & AddMonth () noexcept
- DateHolder & AddMonth (int n) noexcept
- DateHolder & AddDay () noexcept
- std::tm Tm () const noexcept
- size_t CalculateNumberOfDays (const int &periodicity) const noexcept
- void CheckIfReplenishmentWithdrawalToday (const lst_vector &lst_vector, double &deposit, bool sub=false) const

Public Attributes

- time_t **date**

6.6.1 Detailed Description

Class that keeps dates, simple version of QDate.

Definition at line 47 of file model.h.

6.6.2 Constructor & Destructor Documentation

6.6.2.1 DateHolder() [1/2]

```
s21::model::DateHolder::DateHolder (
        int day,
        int month,
        int year )  [noexcept]
```

Construct from explicit date

Parameters

day	
month	
year	

Definition at line 91 of file model-financial.cc.

6.6.2.2 DateHolder() [2/2] s21::model::DateHolder::DateHolder (std::tm *other*) [explicit], [noexcept]

Construct from std::tm

Parameters

other	- from where to construct

Definition at line 164 of file model-financial.cc.

6.6.3 Member Function Documentation

6.6.3.1 AddDay() model::DateHolder & s21::model::DateHolder::AddDay ( ) [noexcept]

Increase stored date by day

Returns

stored date

Definition at line 145 of file model-financial.cc.

6.6.3.2 AddMonth() [1/2] model::DateHolder & s21::model::DateHolder::AddMonth ( ) [noexcept]

Increase stored date by month

Returns

stored date

Definition at line 133 of file model-financial.cc.

6.6.3.3 AddMonth() [2/2] model::DateHolder & s21::model::DateHolder::AddMonth (int *n*) [noexcept]

Increase stored date by

Parameters

n	months

Returns

stored date

Definition at line 159 of file model-financial.cc.

6.6.3.4 CalculateNumberOfDays()

```
size_t s21::model::DateHolder::CalculateNumberOfDays (
        const int & periodicity ) const  [noexcept]
```

Transform periodicity to day's number

Parameters

periodicity	- periodicity from enum

Returns

number of days

Definition at line 178 of file model-financial.cc.

6.6.3.5 CheckIfReplenishmentWithdrawalToday()

```
void s21::model::DateHolder::CheckIfReplenishment↩
WithdrawalToday (
        const lst_vector & lst_vector,
        double & deposit,
        bool sub = false ) const
```

Checks if there was a deposit or withdrawal today and performs the corresponding operations

Parameters

lst_vector	- deposit or withdrawal
deposit	current amount of deposit
sub	- deposit and withdrawal diffs

Definition at line 202 of file model-financial.cc.

6.6.3.6 Compare()

```
bool s21::model::DateHolder::Compare (
        const DateHolder & other ) const  [noexcept]
```

Date comparison

Parameters

other	- date to compare with

Returns

true, if equal

Definition at line 87 of file model-financial.cc.

6.6.3.7 DaysInMonth() `size_t s21::model::DateHolder::DaysInMonth ( ) const [noexcept]`

Days in stored date's month

Returns

number of days

Definition at line 101 of file model-financial.cc.

6.6.3.8 DaysInYear() `size_t s21::model::DateHolder::DaysInYear ( ) const [noexcept]`

Days in stored date's year

Returns

number of days

Definition at line 118 of file model-financial.cc.

6.6.3.9 Tm() `std::tm s21::model::DateHolder::Tm ( ) const [noexcept]`

Represent stored date as std::tm

Returns

std::tm

Definition at line 173 of file model-financial.cc.

6.6.3.10 ToDay() `size_t s21::model::DateHolder::ToDay ( const DateHolder & other ) const [noexcept]`

Days from stored date till the

Parameters

other	date

Returns

number of days

Definition at line 169 of file model-financial.cc.

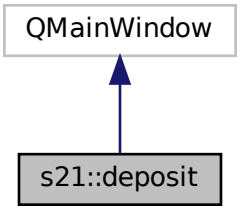
The documentation for this struct was generated from the following files:

- include/model.h
- sources/model-financial.cc

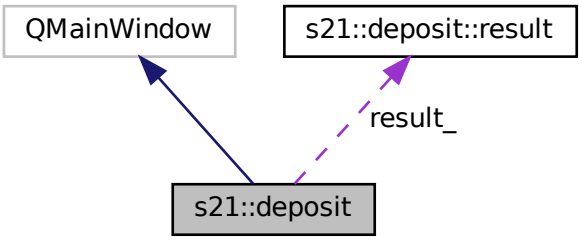
6.7 s21::deposit Class Reference

`#include <deposit.h>`

Inheritance diagram for s21::deposit:



Collaboration diagram for s21::deposit:



Classes

- struct arg

  Struct to keep all info for deposit calculation.
- struct lists

  struct to keep date and amount of money
- struct result

  Struct to keep result info from deposit calculation.

Public Types

- using **vector** = QVector< lists >

Signals

- void **ShowMain** ()
- void **ShowSecondWindow** ()
- void **Close** ()
- void **CalculateDeposit** (const arg &argument)

Public Member Functions

- deposit (QMainWindow *parent=nullptr)
- ~deposit () override
- void SetResult (result &&result)
- void SetError ()

Private Types

- enum { **top** , **bot** , **add_** , **remove_** }

Private Slots

- void **on_actionBasic_triggered** ()
- void **on_actionCredit_triggered** ()
- void **on_replenishments_clicked** ()
- void **on_withdrawals_clicked** ()
- void **on_add_buttonClicked** (QAbstractButton *button)
- void delete_row_from_top_table ()
- void delete_row_from_bot_table ()
- void **on_calculate_clicked** ()

Private Member Functions

- void ManageBox (bool where, int action)
- long long int CalculateNumberOfDays ()

Static Private Member Functions

- static void SetResize (QTableWidget ∗widget)
- static void InitDate (QDateEdit ∗date)
- static vector GetListFromTable (QTableWidget ∗table)

Private Attributes

- Ui::deposit ∗ **ui**
- result **result_** {}
- bool **error_** = false

6.7.1 Detailed Description

Class for deposit widget window

Definition at line 28 of file deposit.h.

6.7.2 Constructor & Destructor Documentation

6.7.2.1 deposit() `s21::deposit::deposit (`

`QMainWindow * parent = nullptr ) [explicit]`

Ctor. Set ui and connects

Parameters

parent	

Definition at line 13 of file deposit.cc.

6.7.2.2 ∼deposit() `s21::deposit::~deposit ( ) [override]`

Dtor.

Definition at line 26 of file deposit.cc.

6.7.3 Member Function Documentation

6.7.3.1 CalculateNumberOfDays() long long int s21::deposit::CalculateNumberOfDays ( ) [private]

Convert input term to days number

Returns

days number

Definition at line 173 of file deposit.cc.

6.7.3.2 delete_row_from_bot_table void s21::deposit::delete_row_from_bot_table ( ) [private], [slot]

Slot that deletes corresponding row from bot table

Definition at line 94 of file deposit.cc.

6.7.3.3 delete_row_from_top_table void s21::deposit::delete_row_from_top_table ( ) [private], [slot]

Slot that deletes corresponding row from top table

Definition at line 83 of file deposit.cc.

6.7.3.4 GetListFromTable() deposit::vector s21::deposit::GetListFromTable (QTableWidget * *table*) [static], [private]

Converts table info to vector

Parameters

table	

Returns

Definition at line 162 of file deposit.cc.

6.7.3.5 InitDate() void s21::deposit::InitDate (QDateEdit * *date*) [static], [private]

Init dates with current date

Parameters

date	- date to init

Definition at line 81 of file deposit.cc.

6.7.3.6 ManageBox() `void s21::deposit::ManageBox (`

```
bool where,
int action )  [private]
```

Managing tables with corresponding params from enum

Parameters

where	
action	

Definition at line 57 of file deposit.cc.

6.7.3.7 SetError() `void s21::deposit::SetError ( )`

Public method for controller to set error

Definition at line 191 of file deposit.cc.

6.7.3.8 SetResize() `void s21::deposit::SetResize (`

```
QTableWidget * widget )  [static], [private]
```

Fixes tables sizes

Parameters

date	

Definition at line 75 of file deposit.cc.

6.7.3.9 SetResult() `void s21::deposit::SetResult (`

```
deposit::result && result )
```

Public method for controller to set result

Parameters

result	

Definition at line 189 of file deposit.cc.

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

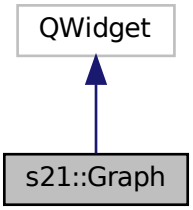
- include/deposit.h
- sources/deposit.cc

6.8 s21::Graph Class Reference

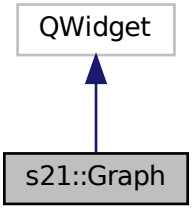
Class for graph window widget.

`#include <graph.h>`

Inheritance diagram for s21::Graph:



Collaboration diagram for s21::Graph:



Signals

- void **RequestCalculation** (const double &x)
- void **SetPrecision** (const double &precision=1e-7)

Public Member Functions

- Graph (const QString &func, QWidget *parent=nullptr)
- ~Graph () override
- void SetResult (const double &result)

Private Slots

- void **on_pushButton_redraw_clicked** ()
- void **on_domain_min_spin_valueChanged** ()
- void **on_domain_max_spin_valueChanged** ()
- void **on_codomain_max_spin_valueChanged** ()
- void **on_codomain_min_spin_valueChanged** ()
- void **on_domain_check_stateChanged** ()
- void **on_codomain_check_stateChanged** ()

Private Member Functions

- void Plot ()
- void PollSpins ()
- void showEvent (QShowEvent *event) override

Private Attributes

- double **result_** = NAN
- Ui::Graph * **ui**
- double **x_min_** = 0
- double **x_max_** = 0
- double **y_min_** = 0
- double **y_max_** = 0

6.8.1 Detailed Description

Class for graph window widget.

Definition at line 25 of file graph.h.

6.8.2 Constructor & Destructor Documentation

6.8.2.1 Graph()

```
s21::Graph::Graph (
        const QString & func,
        QWidget * parent = nullptr )  [explicit]
```

Constructor. Set ui and connects. Get expression to display

Parameters

func	
parent	

Definition at line 12 of file graph.cc.

6.8.2.2 ~Graph() `s21::Graph::~Graph ( ) [override]`

Destructor

Definition at line 18 of file graph.cc.

6.8.3 Member Function Documentation

6.8.3.1 Plot() `void s21::Graph::Plot ( ) [private]`

Method to plot the expression

Definition at line 25 of file graph.cc.

6.8.3.2 PollSpins() `void s21::Graph::PollSpins ( ) [private]`

Polls spins values from gui

Definition at line 52 of file graph.cc.

6.8.3.3 SetResult() `void s21::Graph::SetResult ( const double & result )`

Public method for controller to set result

Parameters

result	

Definition at line 111 of file graph.cc.

6.8.3.4 showEvent() `void s21::Graph::showEvent ( QShowEvent * event ) [override], [private]`

Overrides show method to plot expression before show

Parameters

event	

Definition at line 20 of file graph.cc.

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

- include/graph.h
- sources/graph.cc

6.9 s21::deposit::lists Struct Reference

struct to keep date and amount of money

`#include <deposit.h>`

Public Attributes

- QDate **date**
- double **amount** {}

6.9.1 Detailed Description

struct to keep date and amount of money

Definition at line 35 of file deposit.h.

The documentation for this struct was generated from the following file:

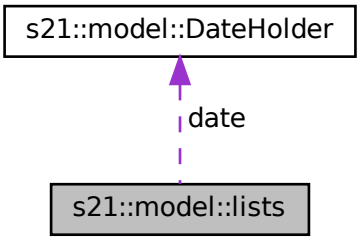
- include/deposit.h

6.10 s21::model::lists Struct Reference

Stores date and amount of money.

`#include <model.h>`

Collaboration diagram for s21::model::lists:



Public Attributes

- model::DateHolder **date**
- double **amount** {}

6.10.1 Detailed Description

Stores date and amount of money.

Definition at line 142 of file model.h.

The documentation for this struct was generated from the following file:

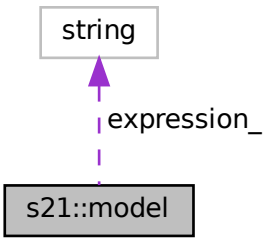
- include/model.h

6.11 s21::model Class Reference

main class for calculations

```
#include <model.h>
```

Collaboration diagram for s21::model:



Classes

- struct arg

  Keeps all information to provide deposit calculation.
- struct DateHolder

  Class that keeps dates, simple version of QDate.
- struct lists

  Stores date and amount of money.
- struct result

  Keeps result information after deposit calculation.
- struct unit

  Internal struct for rpn and calculation.

Public Types

- enum {
  day , **week** , **month** , **quarter** ,
  half_year , **year** }
- using **lst_vector** = std::vector< lists >
- using **vector** = std::vector< double >

Public Member Functions

- double Evaluate (double x)
- void SetExpression (const string &string)
- void SetPrecision (const double &precision) noexcept

Static Public Member Functions

- static vector CalculateCredit (double loan_amount, const unsigned &loan_term, const double &interest_rate, const bool &annuity)
- static model::result CalculateDeposit (const arg &argument)

Private Types

- using **rpn** = std::queue< unit >
- using **stack** = std::stack< unit >

Private Member Functions

- void TransformStringBeforeParsing ()
- void ReplaceFuncNameWithSymbol ()
- void ReplaceUnaryMinusWithTilde () noexcept
- void Validate () const
- void ParseStringToRPN ()
- double Calculate (double x)
- void ReplaceSinglePointWithZero () noexcept
- void DeleteUnaryPlus () noexcept
- void PushOperand (size_t &pos)

Static Private Member Functions

- static bool IsOperand (const char &ch) noexcept
- static bool IsDigit (const char &ch) noexcept
- static int Prioritize (const char &c) noexcept

Private Attributes

- string **expression_**
- rpn **string_in_rpn_** {}
- double **precision_** = 1e-7
- bool **modified_** = false

6.11.1 Detailed Description

main class for calculations

Definition at line 31 of file model.h.

6.11.2 Member Function Documentation

6.11.2.1 Calculate() `double s21::model::Calculate ( double x ) [private]`

calculate result with given params from rpn

Parameters

x	

Returns

result

Definition at line 75 of file model.cc.

6.11.2.2 CalculateCredit() `model::vector s21::model::CalculateCredit ( double loan_amount, const unsigned & loan_term, const double & interest_rate, const bool & annuity ) [static]`

Calculate credit info

Parameters

loan_amount	
loan_term	
interest_rate	
annuity	

Returns

std::vector of <doubles>, that contains month's payments. Only one record, if annuity == true

Definition at line 12 of file model-financial.cc.

6.11.2.3 CalculateDeposit() `model::result s21::model::CalculateDeposit ( const arg & argument ) [static]`

Calculate deposit info from given

Parameters

argument	

Returns

struct result

Definition at line 35 of file model-financial.cc.

6.11.2.4 DeleteUnaryPlus() `void s21::model::DeleteUnaryPlus ( ) [private], [noexcept]`

Just deletes all unary pluses

Definition at line 183 of file model.cc.

6.11.2.5 Evaluate() `double s21::model::Evaluate ( double x )`

Calculates expression_ with

Parameters

x	

Returns

calculation result

Definition at line 14 of file model.cc.

6.11.2.6 IsDigit() `bool s21::model::IsDigit ( const char & ch ) [static], [private], [noexcept]`

Checks if symbol is digit (== digit, . or X).

Parameters

ch	- symbol to check

Returns

- true, if yes

Definition at line 136 of file model.cc.

6.11.2.7 IsOperand() `bool s21::model::IsOperand ( const char & ch ) [static], [private], [noexcept]`

Checks if operand

Parameters

ch	- symbol to check

Returns

true, if yes

Definition at line 132 of file model.cc.

6.11.2.8 ParseStringToRPN() `void s21::model::ParseStringToRPN ( ) [private]`

Convert string to rpn representation

Definition at line 25 of file model.cc.

6.11.2.9 Prioritize() `int s21::model::Prioritize ( const char & c ) [static], [private], [noexcept]`

Returns corresponding priority

Parameters

c	- operator to get prority

Returns

number

Definition at line 115 of file model.cc.

6.11.2.10 PushOperand() `void s21::model::PushOperand ( size_t & pos ) [private]`

Operand smart push depending on X

Parameters

pos	- position of a symbol

Definition at line 58 of file model.cc.

6.11.2.11 ReplaceFuncNameWithSymbol() `void s21::model::ReplaceFuncNameWithSymbol ( ) [private]`

Replaces function names with corresponding symbols

Definition at line 147 of file model.cc.

6.11.2.12 ReplaceSinglePointWithZero() `void s21::model::ReplaceSinglePointWithZero ( ) [private], [noexcept]`

Replaces '.' with 0

Definition at line 172 of file model.cc.

6.11.2.13 ReplaceUnaryMinusWithTilde() `void s21::model::ReplaceUnaryMinusWithTilde ( ) [private], [noexcept]`

Replaces unary minuses with tilda symbol

Definition at line 162 of file model.cc.

6.11.2.14 SetExpression() `void s21::model::SetExpression ( const string & string )`

Sets expression_, that will be calculated

Parameters

string	- expression_

Definition at line 194 of file model.cc.

6.11.2.15 SetPrecision() `void s21::model::SetPrecision ( const double & precision ) [noexcept]`

Sets precision_

Parameters

precision	- precision↩ _

Definition at line 201 of file model.cc.

6.11.2.16 TransformStringBeforeParsing() `void s21::model::TransformStringBeforeParsing ( ) [private]`

Modifies string before parsing, Deleting unary pluses, transforms unary minuses to tilda and so on.

Definition at line 140 of file model.cc.

6.11.2.17 Validate() `void s21::model::Validate ( ) const [private]`

String still needs to be checked if it is incorrect

Definition at line 99 of file model.cc.

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

- include/model.h
- sources/model-financial.cc
- sources/model.cc

6.12 s21::deposit::result Struct Reference

Struct to keep result info from deposit calculation.

`#include <deposit.h>`

Public Attributes

- double **accrued_interest**
- double **tax**
- double **deposit_amount**

6.12.1 Detailed Description

Struct to keep result info from deposit calculation.

Definition at line 61 of file deposit.h.

The documentation for this struct was generated from the following file:

- include/deposit.h

6.13 s21::model::result Struct Reference

Keeps result information after deposit calculation.

`#include <model.h>`

Public Attributes

- double **accrued_interest**
- double **tax**
- double **deposit_amount**

6.13.1 Detailed Description

Keeps result information after deposit calculation.

Definition at line 166 of file model.h.

The documentation for this struct was generated from the following file:

- include/model.h

6.14 s21::model::unit Struct Reference

Internal struct for rpn and calculation.

Public Types

- enum { **operator_** = 1 , **operand_** = 2 , **operand_x** = 3 }

Public Member Functions

- void CalculateWithOneOperand (unit &value, const double &precision) const noexcept
- void CalculateWithTwoOperands (stack &stack, const double &precision) const
- void RoundToZero (const double &precision) noexcept

Public Attributes

- union {
  double **db**
  char **ch**
  };
- enum s21::model::unit:: { ... } **type**

6.14.1 Detailed Description

Internal struct for rpn and calculation.

Definition at line 223 of file model.h.

6.14.2 Member Function Documentation

6.14.2.1 CalculateWithOneOperand()

```
void s21::model::unit::CalculateWithOneOperand (
        unit & value,
        const double & precision ) const  [noexcept]
```

Calculates operation with one operand and places a result to stack

Parameters

value	- stack's top elem
precision	- precision to round a value

Definition at line 205 of file model.cc.

6.14.2.2 CalculateWithTwoOperands()

```
void s21::model::unit::CalculateWithTwoOperands (
        stack & stack,
        const double & precision ) const
```

Calculates operation with two operands and places a result to stack

Parameters

stack	- stack
precision	- precision to round a value

Definition at line 226 of file model.cc.

6.14.2.3 RoundToZero() `void s21::model::unit::RoundToZero ( const double & precision ) [noexcept]`

Rounds the value to zero if abs(value) $<$ eps

Parameters

precision	- precision to round a value

Definition at line 250 of file model.cc.

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

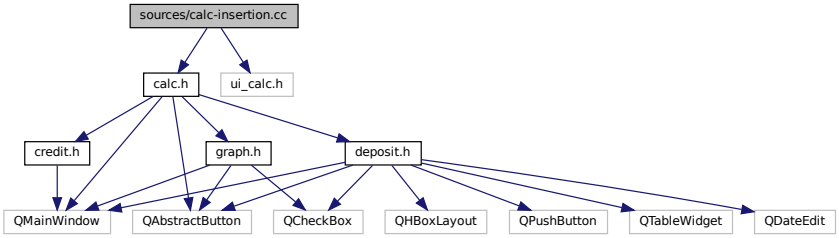
- include/model.h
- sources/model.cc

7 File Documentation

7.1 sources/calc-insertion.cc File Reference

```
#include "calc.h"
#include "ui_calc.h"
```

Include dependency graph for calc-insertion.cc:



Namespaces

- s21

  standard namespace for school project

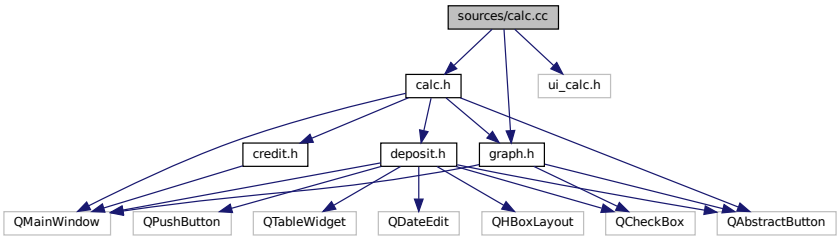
7.1.1 Detailed Description

- insertion logic methods definitions

7.2 sources/calc.cc File Reference

```
#include "calc.h"
#include "graph.h"
#include "ui_calc.h"
```

Include dependency graph for calc.cc:



Namespaces

- s21

  standard namespace for school project

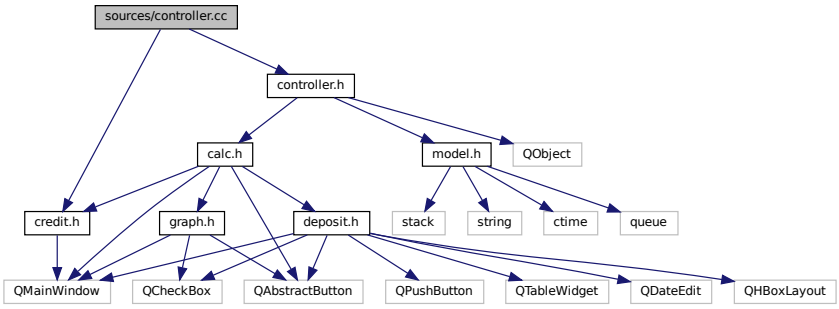
7.2.1 Detailed Description

- base widget methods definitions

7.3 sources/controller.cc File Reference

```
#include "controller.h"
#include "credit.h"
```

Include dependency graph for controller.cc:



Namespaces

- s21

  standard namespace for school project

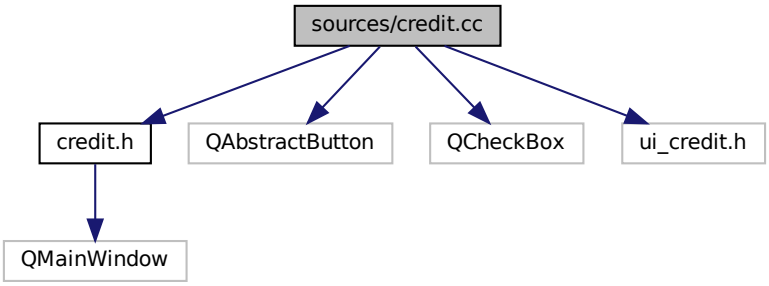
7.3.1 Detailed Description

controller class definitions

7.4 sources/credit.cc File Reference

```
#include "credit.h"
#include <QAbstractButton>
#include <QCheckBox>
#include "ui_credit.h"
```

Include dependency graph for credit.cc:



Namespaces

- s21

  standard namespace for school project

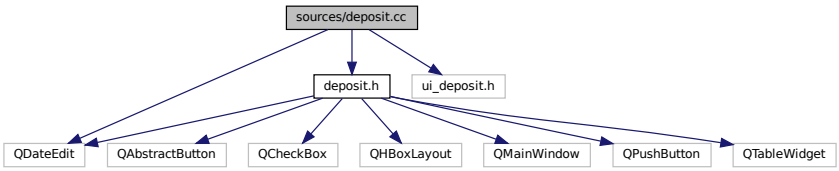
7.4.1 Detailed Description

- credit class definition

7.5 sources/deposit.cc File Reference

```
#include "deposit.h"
#include <QDateEdit>
#include "ui_deposit.h"
```

Include dependency graph for deposit.cc:



Namespaces

- s21

  standard namespace for school project

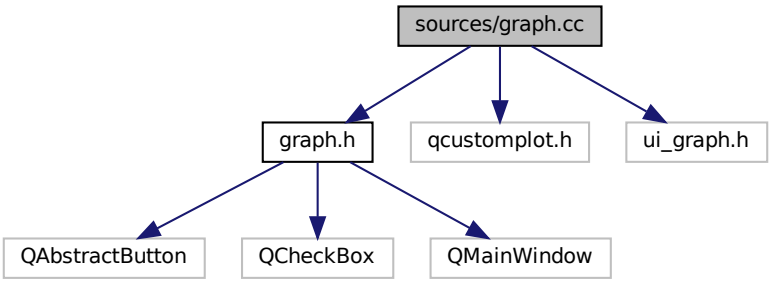
7.5.1 Detailed Description

- deposit class definitions

7.6 sources/graph.cc File Reference

```
#include "graph.h"
#include "qcustomplot.h"
#include "ui_graph.h"
```

Include dependency graph for graph.cc:



Namespaces

- s21

  standard namespace for school project

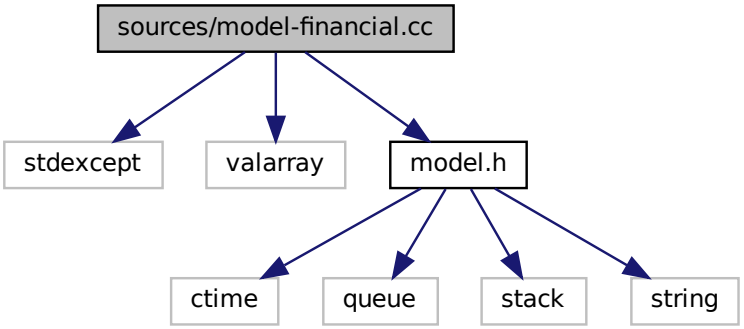
7.6.1 Detailed Description

- graph class definitions

7.7 sources/model-financial.cc File Reference

```
#include <stdexcept>
#include <valarray>
#include "model.h"
```

Include dependency graph for model-financial.cc:



Namespaces

- s21

  standard namespace for school project

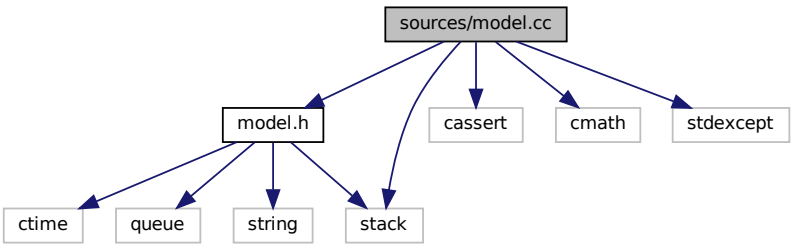
7.7.1 Detailed Description

Bonus part with credit and deposit calculators

7.8 sources/model.cc File Reference

```
#include "model.h"
#include <cassert>
#include <cmath>
#include <stack>
#include <stdexcept>
```

Include dependency graph for model.cc:



Namespaces

- s21

  standard namespace for school project

7.8.1 Detailed Description

All methods for calculations are here

Index

~Calculator
s21::Calculator, 8
~Graph
s21::Graph, 31
~credit
s21::credit, 19
~deposit
s21::deposit, 26

AddDay
s21::model::DateHolder, 21
AddMonth
s21::model::DateHolder, 21

Calculate
s21::model, 35
CalculateCredit
s21::controller, 15
s21::model, 35
CalculateDeposit
s21::controller, 15
s21::model, 35
CalculateExpression
s21::controller, 15
CalculateNumberOfDays
s21::deposit, 26
s21::model::DateHolder, 22
CalculateTotalPayment
s21::credit, 19
CalculateWithOneOperand
s21::model::unit, 41
CalculateWithTwoOperands
s21::model::unit, 41
Calculator
s21::Calculator, 8
CheckIfAlreadyContainPoint
s21::Calculator, 8
CheckIfReplenishmentWithdrawalToday
s21::model::DateHolder, 22
ClearIfNeeded
s21::Calculator, 8
Compare
s21::model::DateHolder, 22
ComplementInsertion
s21::Calculator, 9
ConnectGraph
s21::controller, 15
controller
s21::controller, 14
ConvertArguments
s21::controller, 16
ConvertExponentToDec
s21::Calculator, 10
credit
s21::credit, 18

DateHolder
s21::model::DateHolder, 20
DaysInMonth
s21::model::DateHolder, 23
DaysInYear
s21::model::DateHolder, 23
delete_row_from_bot_table
s21::deposit, 27
delete_row_from_top_table
s21::deposit, 27
DeleteOneToken
s21::Calculator, 10
DeleteUnaryPlus
s21::model, 36
deposit
s21::deposit, 26

Evaluate
s21::Calculator, 10
s21::model, 36

FindPositionToTerminate
s21::Calculator, 10

GetCredit
s21::Calculator, 11
GetDeposit
s21::Calculator, 11
GetListFromTable
s21::deposit, 27
Graph
s21::Graph, 30

InitDate
s21::deposit, 27
IsDigit
s21::Calculator, 11
s21::model, 36
IsOperand
s21::model, 37

ManageBox
s21::deposit, 28

ParseStringToRPN
s21::model, 37
Plot
s21::Graph, 31
PlotCalculation
s21::controller, 16
PollSpins
s21::Graph, 31
Prioritize
s21::model, 37
ProvideInsertionLogic
s21::Calculator, 11
PushOperand

s21::model, 37

ReplaceFuncNameWithSymbol
s21::model, 38
ReplaceSinglePointWithZero
s21::model, 38
ReplaceUnaryMinusWithTilde
s21::model, 38
RoundToZero
s21::model::unit, 41

s21, 4
s21::Calculator, 6
∼Calculator, 8
Calculator, 8
CheckIfAlreadyContainPoint, 8
ClearIfNeeded, 8
ComplementInsertion, 9
ConvertExponentToDec, 10
DeleteOneToken, 10
Evaluate, 10
FindPositionToTerminate, 10
GetCredit, 11
GetDeposit, 11
IsDigit, 11
ProvideInsertionLogic, 11
SetError, 12
SetResult, 12
SetStatusOK, 12
ShouldAddMultSymbol, 12
ShouldSkipOperatorInsertion, 12
s21::controller, 13
CalculateCredit, 15
CalculateDeposit, 15
CalculateExpression, 15
ConnectGraph, 15
controller, 14
ConvertArguments, 16
PlotCalculation, 16
SetPrecision, 16
s21::credit, 17
∼credit, 19
CalculateTotalPayment, 19
credit, 18
SetCredit, 19
s21::deposit, 24
∼deposit, 26
CalculateNumberOfDays, 26
delete_row_from_bot_table, 27
delete_row_from_top_table, 27
deposit, 26
GetListFromTable, 27
InitDate, 27
ManageBox, 28
SetError, 28
SetResize, 28
SetResult, 28
s21::deposit::arg, 4
s21::deposit::lists, 32
s21::deposit::result, 39
s21::Graph, 29
∼Graph, 31
Graph, 30
Plot, 31
PollSpins, 31
SetResult, 31
showEvent, 31
s21::model, 33
Calculate, 35
CalculateCredit, 35
CalculateDeposit, 35
DeleteUnaryPlus, 36
Evaluate, 36
IsDigit, 36
IsOperand, 37
ParseStringToRPN, 37
Prioritize, 37
PushOperand, 37
ReplaceFuncNameWithSymbol, 38
ReplaceSinglePointWithZero, 38
ReplaceUnaryMinusWithTilde, 38
SetExpression, 38
SetPrecision, 39
TransformStringBeforeParsing, 39
Validate, 39
s21::model::arg, 5
s21::model::DateHolder, 19
AddDay, 21
AddMonth, 21
CalculateNumberOfDays, 22
CheckIfReplenishmentWithdrawalToday, 22
Compare, 22
DateHolder, 20
DaysInMonth, 23
DaysInYear, 23
Tm, 23
ToDay, 23
s21::model::lists, 32
s21::model::result, 40
s21::model::unit, 40
CalculateWithOneOperand, 41
CalculateWithTwoOperands, 41
RoundToZero, 41
SetCredit
s21::credit, 19
SetError
s21::Calculator, 12
s21::deposit, 28
SetExpression
s21::model, 38
SetPrecision
s21::controller, 16
s21::model, 39
SetResize
s21::deposit, 28
SetResult
s21::Calculator, 12

s21::deposit, 28
s21::Graph, 31
SetStatusOK
s21::Calculator, 12
ShouldAddMultSymbol
s21::Calculator, 12
ShouldSkipOperatorInsertion
s21::Calculator, 12
showEvent
s21::Graph, 31
sources/calc-insertion.cc, 42
sources/calc.cc, 43
sources/controller.cc, 43
sources/credit.cc, 44
sources/deposit.cc, 44
sources/graph.cc, 45
sources/model-financial.cc, 46
sources/model.cc, 46

Tm
s21::model::DateHolder, 23
ToDay
s21::model::DateHolder, 23
TransformStringBeforeParsing
s21::model, 39

Validate
s21::model, 39