

Prosta Księgowość Domowa

Generated by Doxygen 1.8.17

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 Date Class Reference	7
4.1.1 Constructor & Destructor Documentation	8
4.1.1.1 Date() [1/3]	8
4.1.1.2 Date() [2/3]	8
4.1.1.3 Date() [3/3]	9
4.1.2 Member Function Documentation	9
4.1.2.1 getDay()	9
4.1.2.2 getMonth()	9
4.1.2.3 getYear()	10
4.1.2.4 isLeapYear()	10
4.1.2.5 loadDate()	11
4.1.2.6 loadDateFromFile()	11
4.1.2.7 operator<=()	12
4.1.2.8 operator>=()	12
4.1.2.9 print()	13
4.1.2.10 setDay()	13
4.1.2.11 setMonth()	14
4.1.2.12 setYear()	14
4.2 IStatistics Class Reference	15
4.2.1 Member Function Documentation	16
4.2.1.1 countMean() [1/4]	17
4.2.1.2 countMean() [2/4]	17
4.2.1.3 countMean() [3/4]	17
4.2.1.4 countMean() [4/4]	17
4.2.1.5 countMedian() [1/4]	17
4.2.1.6 countMedian() [2/4]	18
4.2.1.7 countMedian() [3/4]	18
4.2.1.8 countMedian() [4/4]	18
4.2.1.9 countStandardDeviation() [1/4]	18
4.2.1.10 countStandardDeviation() [2/4]	18
4.2.1.11 countStandardDeviation() [3/4]	19
4.2.1.12 countStandardDeviation() [4/4]	19
4.2.1.13 countSum() [1/4]	19

4.2.1.14 countSum() [2/4]	19
4.2.1.15 countSum() [3/4]	19
4.2.1.16 countSum() [4/4]	20
4.2.1.17 getMean()	20
4.2.1.18 getMedian()	20
4.2.1.19 getStDev()	20
4.2.1.20 getSum()	20
4.2.1.21 print()	20
4.3 Member Class Reference	21
4.3.1 Constructor & Destructor Documentation	21
4.3.1.1 Member()	21
4.3.2 Member Function Documentation	22
4.3.2.1 getName()	22
4.3.2.2 getPosition()	22
4.3.2.3 isSame()	23
4.3.2.4 loadMember()	23
4.3.2.5 loadMemberFromFile()	24
4.3.2.6 operator==()	24
4.4 MemberList Class Reference	24
4.4.1 Constructor & Destructor Documentation	25
4.4.1.1 MemberList()	25
4.4.1.2 ~MemberList()	26
4.4.2 Member Function Documentation	26
4.4.2.1 addNode() [1/2]	26
4.4.2.2 addNode() [2/2]	27
4.4.2.3 addNodeFromFile()	27
4.4.2.4 getHead()	28
4.4.2.5 isInList()	28
4.4.2.6 loadFromFile()	29
4.4.2.7 print()	29
4.4.2.8 printToFile()	30
4.4.2.9 setHead()	31
4.5 MemberNode Class Reference	31
4.5.1 Constructor & Destructor Documentation	32
4.5.1.1 MemberNode() [1/2]	32
4.5.1.2 MemberNode() [2/2]	32
4.5.2 Member Function Documentation	32
4.5.2.1 getNext()	32
4.5.2.2 getNode()	33
4.5.2.3 setNext()	33
4.5.2.4 setNode()	33
4.6 Product Class Reference	33

4.6.1 Constructor & Destructor Documentation	34
4.6.1.1 Product()	35
4.6.2 Member Function Documentation	35
4.6.2.1 getCategory()	35
4.6.2.2 getName()	35
4.6.2.3 getPrice()	36
4.6.2.4 getShop()	36
4.6.2.5 isSame()	37
4.6.2.6 loadProduct() [1/2]	37
4.6.2.7 loadProduct() [2/2]	38
4.6.2.8 loadProductFromFile()	38
4.6.2.9 operator==()	39
4.7 ProductList Class Reference	39
4.7.1 Constructor & Destructor Documentation	40
4.7.1.1 ProductList()	40
4.7.1.2 ~ProductList()	40
4.7.2 Member Function Documentation	40
4.7.2.1 addNode() [1/3]	41
4.7.2.2 addNode() [2/3]	41
4.7.2.3 addNode() [3/3]	42
4.7.2.4 addNodeFromFile()	42
4.7.2.5 getHead()	43
4.7.2.6 howManyInList()	43
4.7.2.7 isInList()	44
4.7.2.8 loadFromFile()	44
4.7.2.9 print()	45
4.7.2.10 printInLine()	45
4.7.2.11 printToFile()	46
4.7.2.12 printWithAmount()	47
4.7.2.13 setHead()	48
4.7.2.14 wasItBefore()	49
4.8 ProductNode Class Reference	49
4.8.1 Constructor & Destructor Documentation	50
4.8.1.1 ProductNode() [1/2]	50
4.8.1.2 ProductNode() [2/2]	50
4.8.2 Member Function Documentation	50
4.8.2.1 getNext()	51
4.8.2.2 getNode()	51
4.8.2.3 setNext()	51
4.8.2.4 setNode()	52
4.9 Statistics Class Reference	52
4.9.1 Member Function Documentation	54

4.9.1.1 countMean() [1/4]	54
4.9.1.2 countMean() [2/4]	55
4.9.1.3 countMean() [3/4]	55
4.9.1.4 countMean() [4/4]	56
4.9.1.5 countMedian() [1/4]	56
4.9.1.6 countMedian() [2/4]	57
4.9.1.7 countMedian() [3/4]	58
4.9.1.8 countMedian() [4/4]	59
4.9.1.9 countStandardDeviation() [1/4]	59
4.9.1.10 countStandardDeviation() [2/4]	60
4.9.1.11 countStandardDeviation() [3/4]	60
4.9.1.12 countStandardDeviation() [4/4]	61
4.9.1.13 countSum() [1/4]	62
4.9.1.14 countSum() [2/4]	62
4.9.1.15 countSum() [3/4]	63
4.9.1.16 countSum() [4/4]	64
4.9.1.17 getMean()	64
4.9.1.18 getMedian()	65
4.9.1.19 getStDev()	65
4.9.1.20 getSum()	65
4.9.1.21 print()	66
4.10 Transaction Class Reference	66
4.10.1 Constructor & Destructor Documentation	67
4.10.1.1 Transaction()	67
4.10.2 Member Function Documentation	68
4.10.2.1 getDate()	68
4.10.2.2 getMember()	68
4.10.2.3 getPrice()	69
4.10.2.4 getPurchaseList()	70
4.10.2.5 getShop()	70
4.10.2.6 loadTransaction()	70
4.10.2.7 loadTransactionFromFile()	71
4.11 TransactionList Class Reference	72
4.11.1 Constructor & Destructor Documentation	73
4.11.1.1 TransactionList()	73
4.11.1.2 ~TransactionList()	73
4.11.2 Member Function Documentation	73
4.11.2.1 addNode()	73
4.11.2.2 addNodeFromFile()	74
4.11.2.3 getHead()	74
4.11.2.4 loadFromFile()	75
4.11.2.5 print()	75

4.11.2.6 printToFile()	76
4.11.2.7 setHead()	77
4.12 TransactionNode Class Reference	78
4.12.1 Constructor & Destructor Documentation	78
4.12.1.1 TransactionNode() [1/2]	78
4.12.1.2 TransactionNode() [2/2]	79
4.12.2 Member Function Documentation	79
4.12.2.1 getNext()	79
4.12.2.2 getNode()	79
4.12.2.3 setNext()	80
4.12.2.4 setNode()	80
5 File Documentation	81
5.1 Date.cpp File Reference	81
5.2 Date.h File Reference	81
5.3 IStatistics.h File Reference	83
5.4 main.cpp File Reference	84
5.4.1 Function Documentation	85
5.4.1.1 main()	85
5.5 Member.cpp File Reference	86
5.6 Member.h File Reference	87
5.7 MemberList.cpp File Reference	88
5.8 MemberList.h File Reference	89
5.9 MemberNode.cpp File Reference	91
5.10 MemberNode.h File Reference	91
5.11 Product.cpp File Reference	93
5.12 Product.h File Reference	93
5.13 ProductList.cpp File Reference	94
5.14 ProductList.h File Reference	95
5.15 ProductNode.cpp File Reference	97
5.16 ProductNode.h File Reference	97
5.17 Statistics.cpp File Reference	99
5.18 Statistics.h File Reference	99
5.19 Transaction.cpp File Reference	101
5.20 Transaction.h File Reference	101
5.21 TransactionList.cpp File Reference	103
5.22 TransactionList.h File Reference	104
5.23 TransactionNode.cpp File Reference	106
5.24 TransactionNode.h File Reference	107
Index	111

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Date	7
IStatistics	15
Statistics	52
Member	21
MemberList	24
MemberNode	31
Product	33
ProductList	39
ProductNode	49
Transaction	66
TransactionList	72
TransactionNode	78

Chapter 2

Class Index

2.1 Class List

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

Date	7
IStatistics	15
Member	21
MemberList	24
MemberNode	31
Product	33
ProductList	39
ProductNode	49
Statistics	52
Transaction	66
TransactionList	72
TransactionNode	78

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Date.cpp	81
Date.h	81
IStatistics.h	83
main.cpp	84
Member.cpp	86
Member.h	87
MemberList.cpp	88
MemberList.h	89
MemberNode.cpp	91
MemberNode.h	91
Product.cpp	93
Product.h	93
ProductList.cpp	94
ProductList.h	95
ProductNode.cpp	97
ProductNode.h	97
Statistics.cpp	99
Statistics.h	99
Transaction.cpp	101
Transaction.h	101
TransactionList.cpp	103
TransactionList.h	104
TransactionNode.cpp	106
TransactionNode.h	107

Chapter 4

Class Documentation

4.1 Date Class Reference

```
#include <Date.h>
```

Collaboration diagram for Date:

Date
<div><div>+ Date() + Date() + Date() + operator<=() + operator>=() + setYear() + getYear() + setMonth() + getMonth() + setDay() + getDay() + loadDate() + loadDateFromFile() + isLeapYear() + print()</div></div>

Public Member Functions

- [Date](#) ()
- [Date](#) (std::ifstream &)
- [Date](#) (int day, int month, int year)

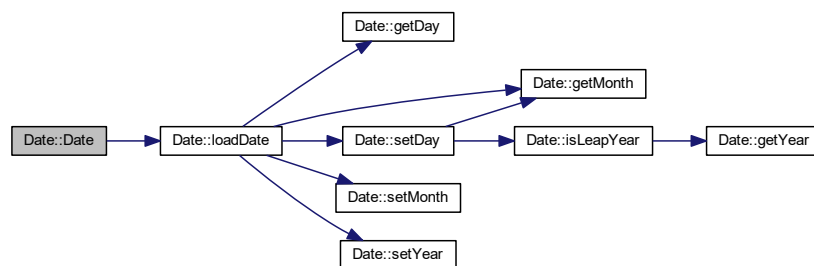
- bool `operator<=` (`Date`)
- bool `operator>=` (`Date`)
- void `setYear` (int year)
- int `getYear` ()
- void `setMonth` (int month)
- int `getMonth` ()
- void `setDay` (int day)
- int `getDay` ()
- `Date` `loadDate` ()
- `Date` `loadDateFromFile` (std::ifstream &)
- bool `isLeapYear` ()
- void `print` ()

4.1.1 Constructor & Destructor Documentation

4.1.1.1 `Date()` [1/3]

```
Date::Date ( )
```

Here is the call graph for this function:



4.1.1.2 `Date()` [2/3]

```
Date::Date (
    std::ifstream & file )
```

Here is the call graph for this function:



4.1.1.3 Date() [3/3]

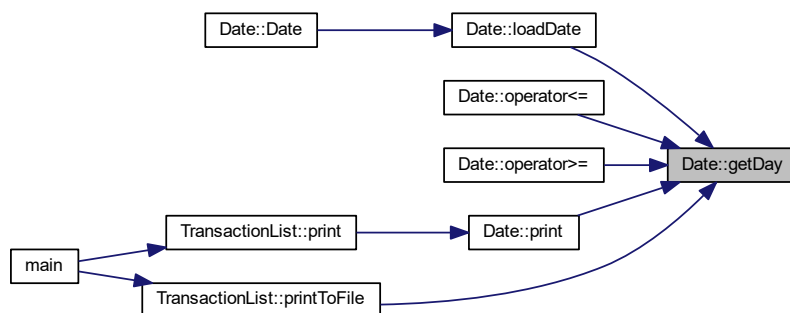
```
Date::Date (
    int day,
    int month,
    int year ) [inline]
```

4.1.2 Member Function Documentation

4.1.2.1 getDay()

```
int Date::getDay ( )
```

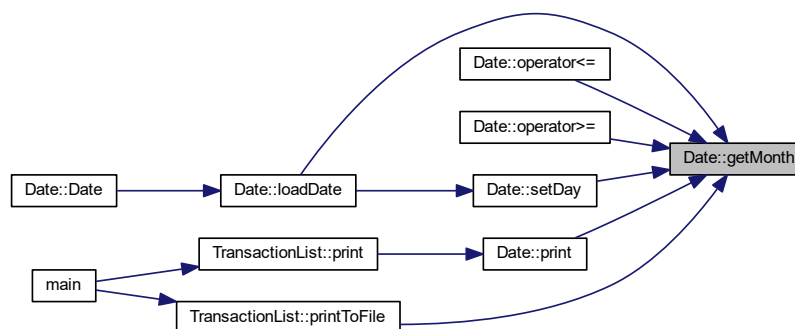
Here is the caller graph for this function:



4.1.2.2 getMonth()

```
int Date::getMonth ( )
```

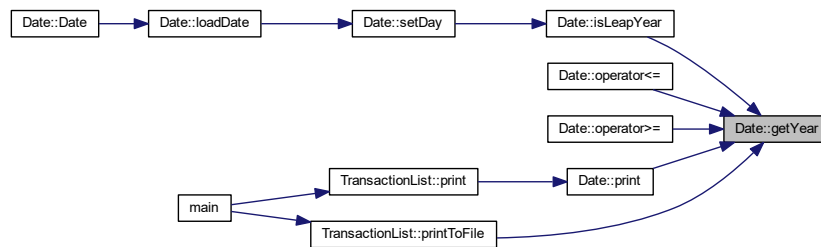
Here is the caller graph for this function:



4.1.2.3 getYear()

```
int Date::getYear ( )
```

Here is the caller graph for this function:



4.1.2.4 isLeapYear()

```
bool Date::isLeapYear ( )
```

Here is the call graph for this function:



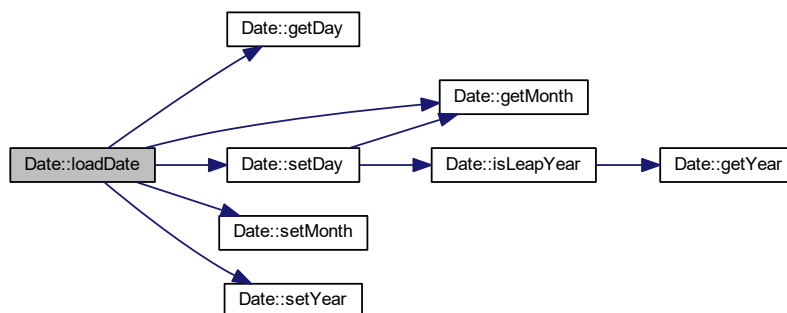
Here is the caller graph for this function:



4.1.2.5 loadDate()

```
Date Date::loadDate ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.2.6 loadDateFromFile()

```
Date Date::loadDateFromFile (
    std::ifstream & file )
```

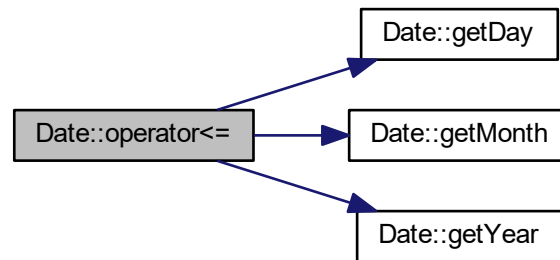
Here is the caller graph for this function:



4.1.2.7 operator<=()

```
bool Date::operator<= (
    Date date )
```

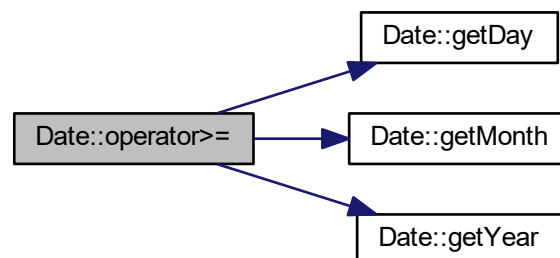
Here is the call graph for this function:



4.1.2.8 operator>=()

```
bool Date::operator>= (
    Date date )
```

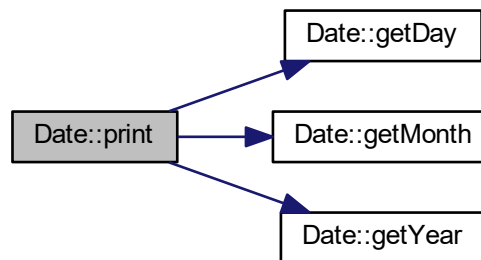
Here is the call graph for this function:



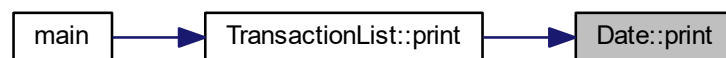
4.1.2.9 print()

```
void Date::print ( )
```

Here is the call graph for this function:



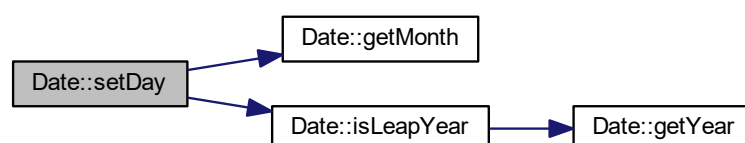
Here is the caller graph for this function:



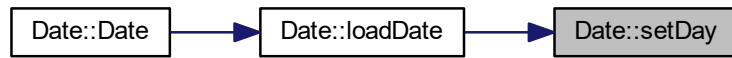
4.1.2.10 setDay()

```
void Date::setDay (
    int day )
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.1.2.11 `setMonth()`

```
void Date::setMonth (  
    int month )
```

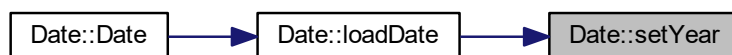
Here is the caller graph for this function:



4.1.2.12 `setYear()`

```
void Date::setYear (  
    int year )
```

Here is the caller graph for this function:



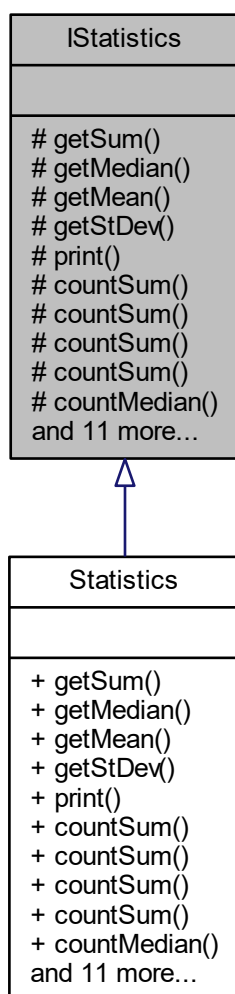
The documentation for this class was generated from the following files:

- [Date.h](#)
- [Date.cpp](#)

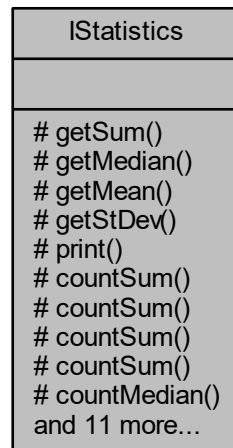
4.2 IStatistics Class Reference

```
#include <IStatistics.h>
```

Inheritance diagram for IStatistics:



Collaboration diagram for IStatistics:



Protected Member Functions

- virtual double [getSum](#) ()=0
- virtual double [getMedian](#) ()=0
- virtual double [getMean](#) ()=0
- virtual double [getStDev](#) ()=0
- virtual void [print](#) ()=0
- virtual void [countSum](#) (TransactionNode *)=0
- virtual void [countSum](#) (TransactionNode *, Member member)=0
- virtual void [countSum](#) (TransactionNode *, Member member, Date start, Date end)=0
- virtual void [countSum](#) (TransactionNode *, Date start, Date end)=0
- virtual void [countMedian](#) (TransactionNode *)=0
- virtual void [countMedian](#) (TransactionNode *, Member member)=0
- virtual void [countMedian](#) (TransactionNode *, Member member, Date start, Date end)=0
- virtual void [countMedian](#) (TransactionNode *, Date start, Date end)=0
- virtual void [countMean](#) (TransactionNode *)=0
- virtual void [countMean](#) (TransactionNode *, Member member)=0
- virtual void [countMean](#) (TransactionNode *, Member member, Date start, Date end)=0
- virtual void [countMean](#) (TransactionNode *, Date start, Date end)=0
- virtual void [countStandardDeviation](#) (TransactionNode *)=0
- virtual void [countStandardDeviation](#) (TransactionNode *, Member member)=0
- virtual void [countStandardDeviation](#) (TransactionNode *, Member member, Date start, Date end)=0
- virtual void [countStandardDeviation](#) (TransactionNode *, Date start, Date end)=0

4.2.1 Member Function Documentation

4.2.1.1 countMean() [1/4]

```
virtual void IStatistics::countMean (  
    TransactionNode * ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.2 countMean() [2/4]

```
virtual void IStatistics::countMean (  
    TransactionNode * ,  
    Date start,  
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.3 countMean() [3/4]

```
virtual void IStatistics::countMean (  
    TransactionNode * ,  
    Member member ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.4 countMean() [4/4]

```
virtual void IStatistics::countMean (  
    TransactionNode * ,  
    Member member,  
    Date start,  
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.5 countMedian() [1/4]

```
virtual void IStatistics::countMedian (  
    TransactionNode * ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.6 countMedian() [2/4]

```
virtual void IStatistics::countMedian (  
    TransactionNode * ,  
    Date start,  
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.7 countMedian() [3/4]

```
virtual void IStatistics::countMedian (  
    TransactionNode * ,  
    Member member ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.8 countMedian() [4/4]

```
virtual void IStatistics::countMedian (  
    TransactionNode * ,  
    Member member,  
    Date start,  
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.9 countStandardDeviation() [1/4]

```
virtual void IStatistics::countStandardDeviation (  
    TransactionNode * ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.10 countStandardDeviation() [2/4]

```
virtual void IStatistics::countStandardDeviation (  
    TransactionNode * ,  
    Date start,  
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.11 countStandardDeviation() [3/4]

```
virtual void IStatistics::countStandardDeviation (
    TransactionNode * ,
    Member member ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.12 countStandardDeviation() [4/4]

```
virtual void IStatistics::countStandardDeviation (
    TransactionNode * ,
    Member member,
    Date start,
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.13 countSum() [1/4]

```
virtual void IStatistics::countSum (
    TransactionNode * ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.14 countSum() [2/4]

```
virtual void IStatistics::countSum (
    TransactionNode * ,
    Date start,
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.15 countSum() [3/4]

```
virtual void IStatistics::countSum (
    TransactionNode * ,
    Member member ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.16 countSum() [4/4]

```
virtual void IStatistics::countSum (
    TransactionNode * ,
    Member member,
    Date start,
    Date end ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.17 getMean()

```
virtual double IStatistics::getMean ( ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.18 getMedian()

```
virtual double IStatistics::getMedian ( ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.19 getStDev()

```
virtual double IStatistics::getStDev ( ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.20 getSum()

```
virtual double IStatistics::getSum ( ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

4.2.1.21 print()

```
virtual void IStatistics::print ( ) [protected], [pure virtual]
```

Implemented in [Statistics](#).

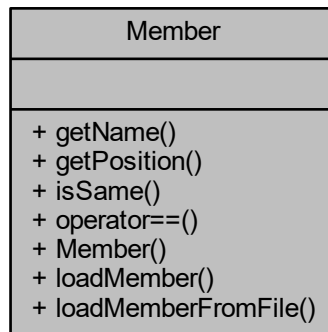
The documentation for this class was generated from the following file:

- [IStatistics.h](#)

4.3 Member Class Reference

```
#include <Member.h>
```

Collaboration diagram for Member:



Public Member Functions

- std::string [getName](#) ()
- std::string [getPosition](#) ()
- bool [isSame](#) ([Member](#) member)
- bool [operator==](#) ([Member](#) member)
- [Member](#) (std::string name, std::string position)

Static Public Member Functions

- static [Member](#) [loadMember](#) ()
- static [Member](#) [loadMemberFromFile](#) (std::ifstream &file)

4.3.1 Constructor & Destructor Documentation

4.3.1.1 Member()

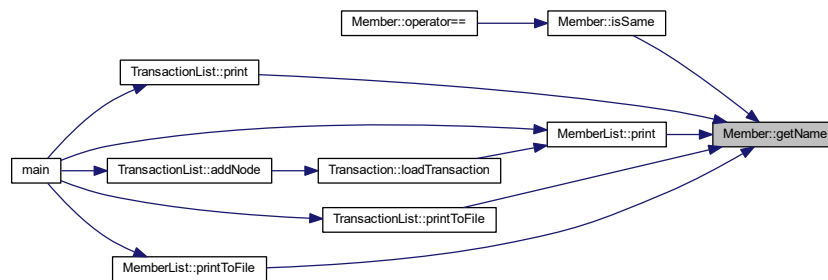
```
Member::Member (
    std::string name,
    std::string position ) [inline]
```

4.3.2 Member Function Documentation

4.3.2.1 getName()

```
std::string Member::getName ( )
```

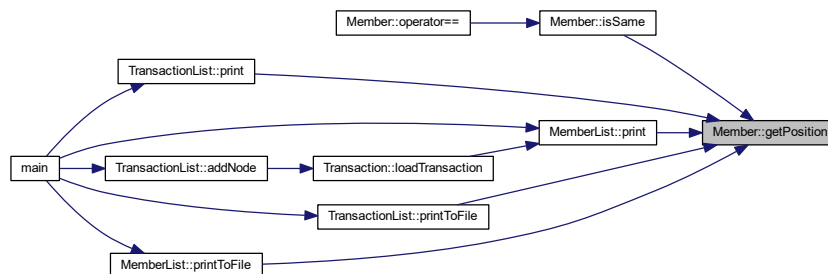
Here is the caller graph for this function:



4.3.2.2 getPosition()

```
std::string Member::getPosition ( )
```

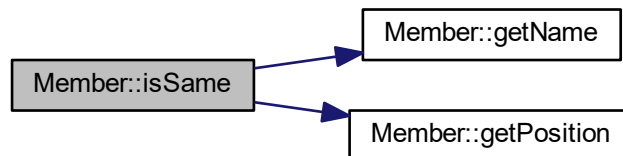
Here is the caller graph for this function:



4.3.2.3 isSame()

```
bool Member::isSame (
    Member member )
```

Here is the call graph for this function:



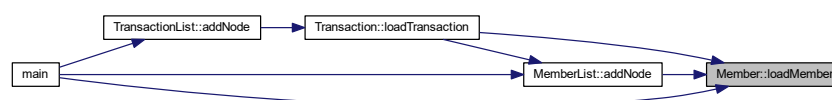
Here is the caller graph for this function:



4.3.2.4 loadMember()

```
Member Member::loadMember ( ) [static]
```

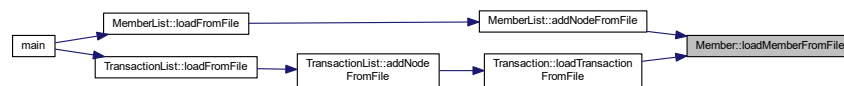
Here is the caller graph for this function:



4.3.2.5 loadMemberFromFile()

```
Member Member::loadMemberFromFile (
    std::ifstream & file ) [static]
```

Here is the caller graph for this function:



4.3.2.6 operator==()

```
bool Member::operator==(
    Member member )
```

Here is the call graph for this function:



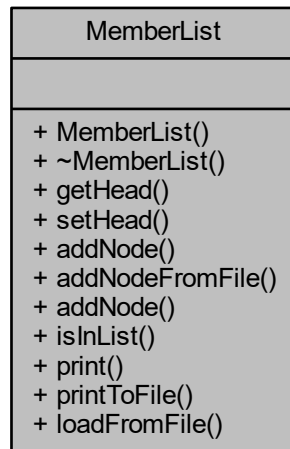
The documentation for this class was generated from the following files:

- [Member.h](#)
- [Member.cpp](#)

4.4 MemberList Class Reference

```
#include <MemberList.h>
```


Collaboration diagram for MemberList:



Public Member Functions

- [MemberList](#) ()
- [~MemberList](#) ()
- [MemberNode](#) * [getHead](#) ()
- void [setHead](#) ([MemberNode](#) *)
- void [addNode](#) ()
- void [addNodeFromFile](#) (std::ifstream &)
- void [addNode](#) ([Member](#), [MemberList](#) &)
- bool [isInList](#) ([Member](#))
- void [print](#) ()
- void [printToFile](#) ()
- void [loadFromFile](#) ()

4.4.1 Constructor & Destructor Documentation

4.4.1.1 MemberList()

```
MemberList::MemberList ( )
```

4.4.1.2 ~MemberList()

```
MemberList::~~MemberList ( )
```

Here is the call graph for this function:

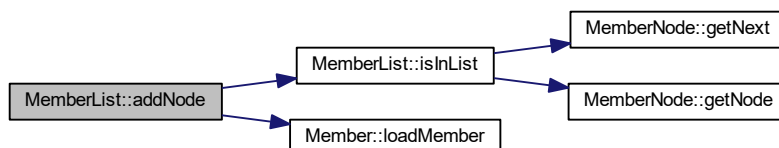


4.4.2 Member Function Documentation

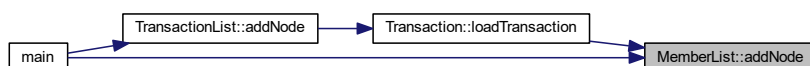
4.4.2.1 addNode() [1/2]

```
void MemberList::addNode ( )
```

Here is the call graph for this function:



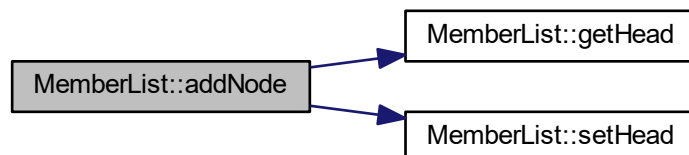
Here is the caller graph for this function:



4.4.2.2 addNode() [2/2]

```
void MemberList::addNode (  
    Member member,  
    MemberList & list )
```

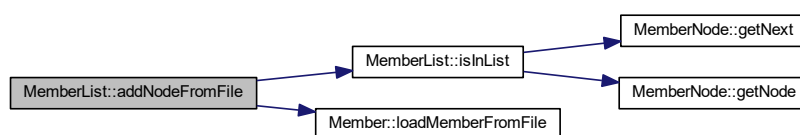
Here is the call graph for this function:



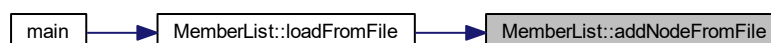
4.4.2.3 addNodeFromFile()

```
void MemberList::addNodeFromFile (  
    std::ifstream & file )
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.4.2.4 getHead()

```
MemberNode * MemberList::getHead ( )
```

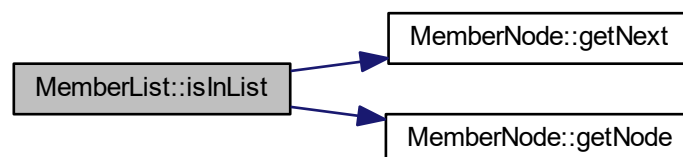
Here is the caller graph for this function:



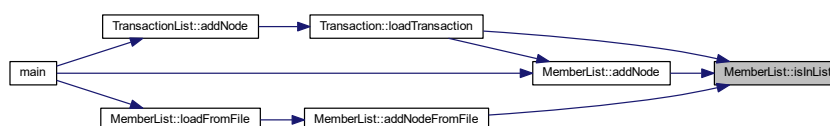
4.4.2.5 isInList()

```
bool MemberList::isInList (
    Member member )
```

Here is the call graph for this function:



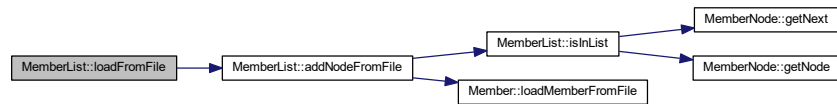
Here is the caller graph for this function:



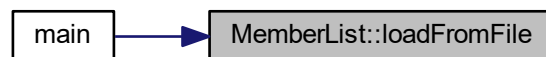
4.4.2.6 loadFromFile()

```
void MemberList::loadFromFile ( )
```

Here is the call graph for this function:



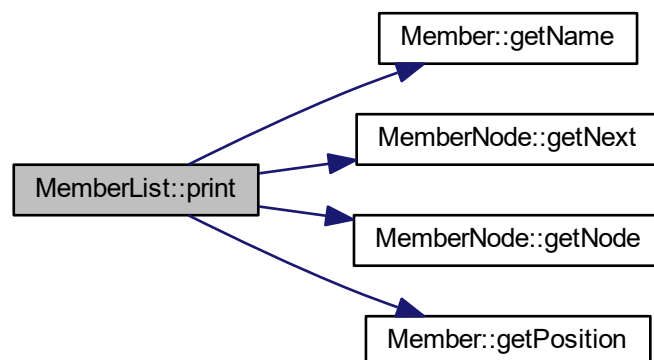
Here is the caller graph for this function:



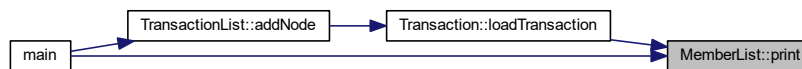
4.4.2.7 print()

```
void MemberList::print ( )
```

Here is the call graph for this function:



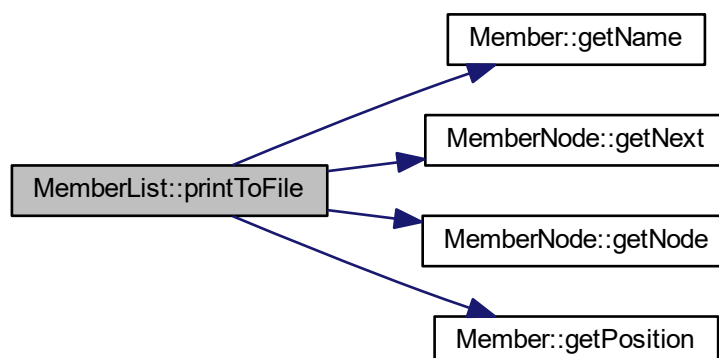
Here is the caller graph for this function:



4.4.2.8 printToFile()

```
void MemberList::printToFile ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.4.2.9 setHead()

```
void MemberList::setHead (
    MemberNode * Node )
```

Here is the caller graph for this function:



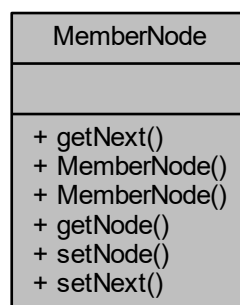
The documentation for this class was generated from the following files:

- [MemberList.h](#)
- [MemberList.cpp](#)

4.5 MemberNode Class Reference

```
#include <MemberNode.h>
```

Collaboration diagram for MemberNode:



Public Member Functions

- [MemberNode * getNext \(\)](#)
- [MemberNode \(Member Node\)](#)
- [MemberNode \(Member Node, MemberNode *pNext\)](#)
- [Member getNode \(\)](#)
- [void setNode \(Member\)](#)
- [void setNext \(MemberNode *\)](#)

4.5.1 Constructor & Destructor Documentation

4.5.1.1 MemberNode() [1/2]

```
MemberNode::MemberNode (
    Member Node ) [inline]
```

4.5.1.2 MemberNode() [2/2]

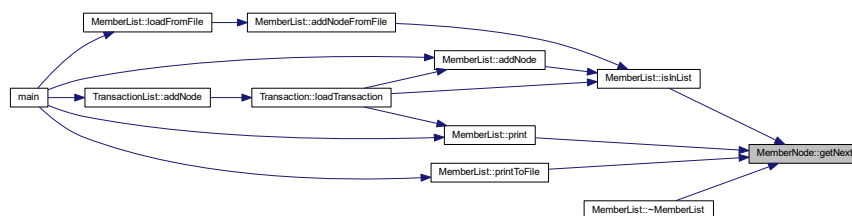
```
MemberNode::MemberNode (
    Member Node,
    MemberNode * pNext ) [inline]
```

4.5.2 Member Function Documentation

4.5.2.1 getNext()

```
MemberNode * MemberNode::getNext ( )
```

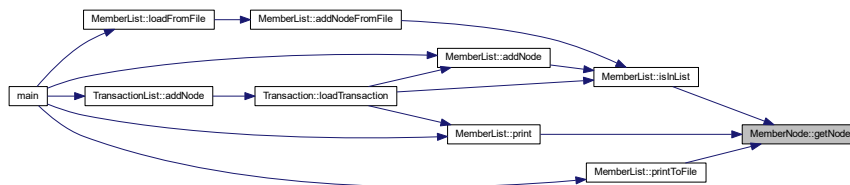
Here is the caller graph for this function:



4.5.2.2 getNode()

```
Member MemberNode::getNode ( )
```

Here is the caller graph for this function:



4.5.2.3 setNext()

```
void MemberNode::setNext (
    MemberNode * pNext )
```

4.5.2.4 setNode()

```
void MemberNode::setNode (
    Member member )
```

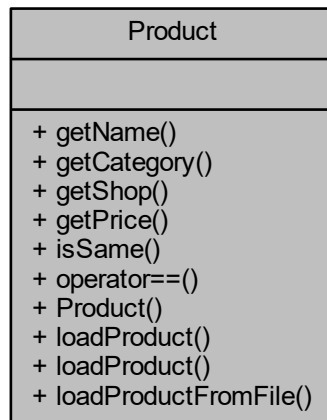
The documentation for this class was generated from the following files:

- [MemberNode.h](#)
- [MemberNode.cpp](#)

4.6 Product Class Reference

```
#include <Product.h>
```

Collaboration diagram for Product:



Public Member Functions

- `std::string getName ()`
- `std::string getCategory ()`
- `std::string getShop ()`
- `double getPrice ()`
- `bool isSame (Product)`
- `bool operator== (Product)`
- `Product (std::string name, std::string category, std::string shop, double price)`

Static Public Member Functions

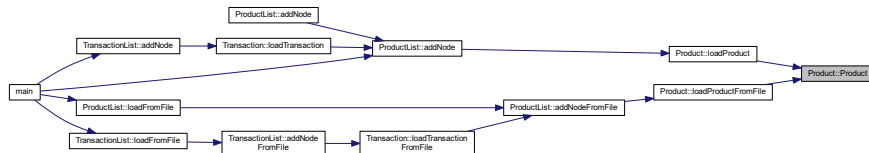
- `static Product loadProduct ()`
- `static Product loadProduct (std::string shop)`
- `static Product loadProductFromFile (std::ifstream &file)`

4.6.1 Constructor & Destructor Documentation

4.6.1.1 Product()

```
Product::Product (
    std::string name,
    std::string category,
    std::string shop,
    double price ) [inline]
```

Here is the caller graph for this function:

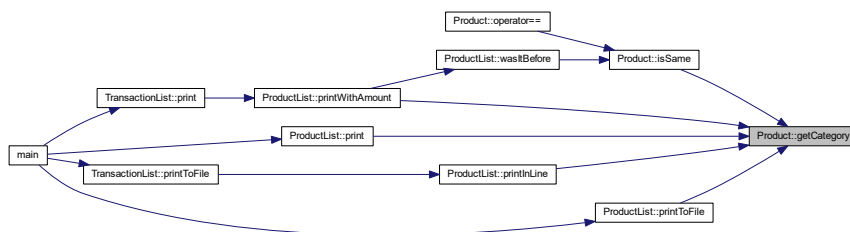


4.6.2 Member Function Documentation

4.6.2.1 getCategory()

```
std::string Product::getCategory ( )
```

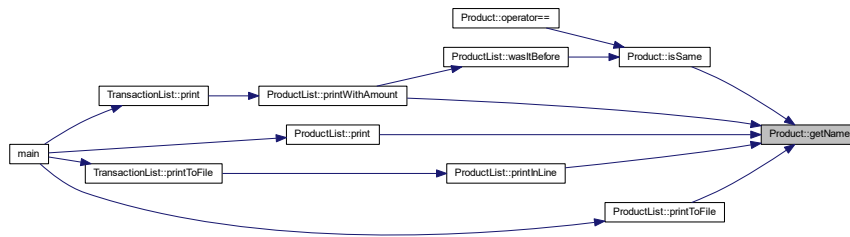
Here is the caller graph for this function:



4.6.2.2 getName()

```
std::string Product::getName ( )
```

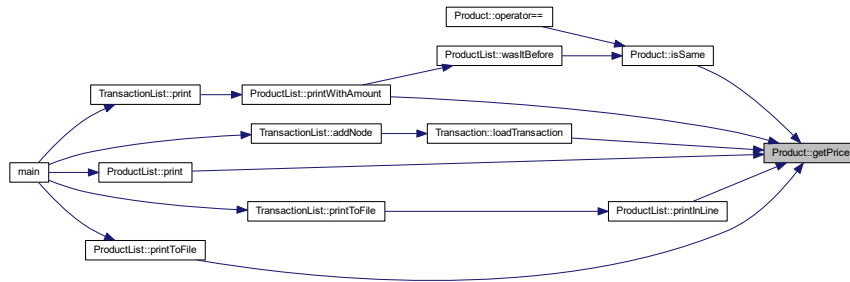
Here is the caller graph for this function:



4.6.2.3 getPrice()

```
double Product::getPrice ( )
```

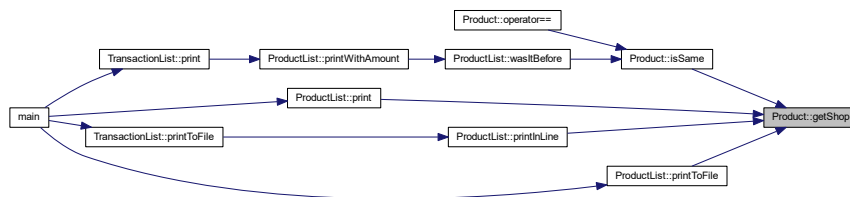
Here is the caller graph for this function:



4.6.2.4 getShop()

```
std::string Product::getShop ( )
```

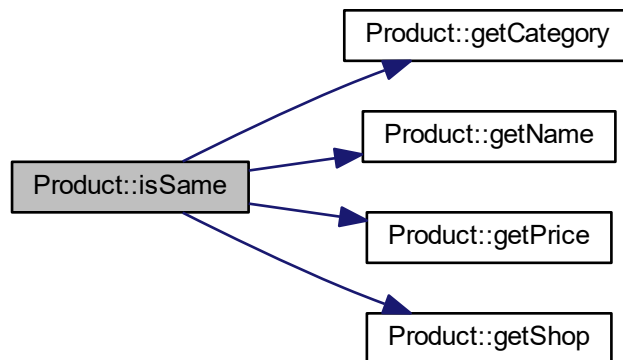
Here is the caller graph for this function:



4.6.2.5 isSame()

```
bool Product::isSame (
    Product product )
```

Here is the call graph for this function:



Here is the caller graph for this function:



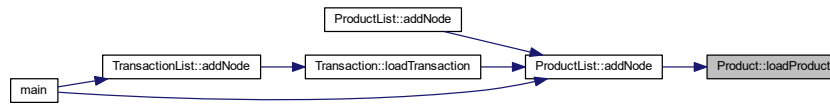
4.6.2.6 loadProduct() [1/2]

```
Product Product::loadProduct ( ) [static]
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.6.2.7 loadProduct() [2/2]

```
Product Product::loadProduct (
    std::string shop ) [static]
```

Here is the call graph for this function:



4.6.2.8 loadProductFromFile()

```
Product Product::loadProductFromFile (
    std::ifstream & file ) [static]
```

Here is the call graph for this function:



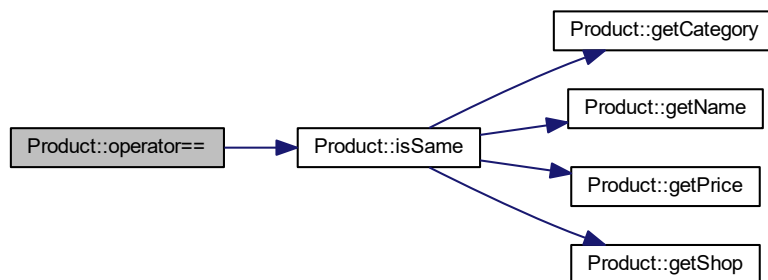
Here is the caller graph for this function:



4.6.2.9 operator==()

```
bool Product::operator== (
    Product product )
```

Here is the call graph for this function:



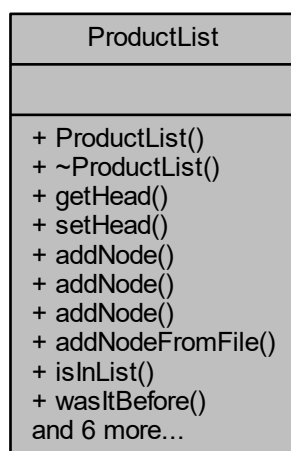
The documentation for this class was generated from the following files:

- [Product.h](#)
- [Product.cpp](#)

4.7 ProductList Class Reference

```
#include <ProductList.h>
```

Collaboration diagram for ProductList:



Public Member Functions

- [ProductList](#) ()
- [~ProductList](#) ()
- [ProductNode](#) * [getHead](#) ()
- void [setHead](#) ([ProductNode](#) *)
- void [addNode](#) ()
- void [addNode](#) ([Product](#), [ProductList](#) &)
- void [addNode](#) ([ProductList](#) &, std::string shop, int amount)
- void [addNodeFromFile](#) (std::ifstream &)
- bool [isInList](#) ([Product](#))
- bool [wasItBefore](#) ([ProductNode](#) *Node)
- int [howManyInList](#) ([Product](#))
- void [print](#) ()
- void [printWithAmount](#) ()
- void [printToFile](#) ()
- void [printlnLine](#) (std::ofstream &)
- void [loadFromFile](#) ()

4.7.1 Constructor & Destructor Documentation

4.7.1.1 ProductList()

```
ProductList::ProductList ( )
```

4.7.1.2 ~ProductList()

```
ProductList::~~ProductList ( )
```

Here is the call graph for this function:

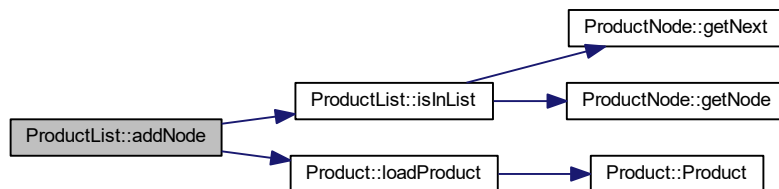


4.7.2 Member Function Documentation

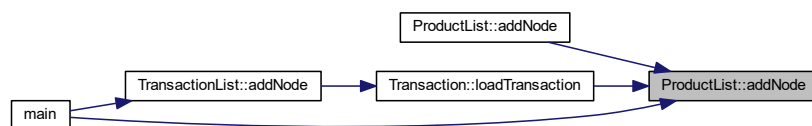
4.7.2.1 addNode() [1/3]

```
void ProductList::addNode ( )
```

Here is the call graph for this function:



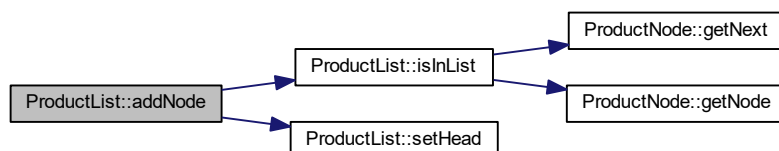
Here is the caller graph for this function:



4.7.2.2 addNode() [2/3]

```
void ProductList::addNode (
    Product product,
    ProductList & products )
```

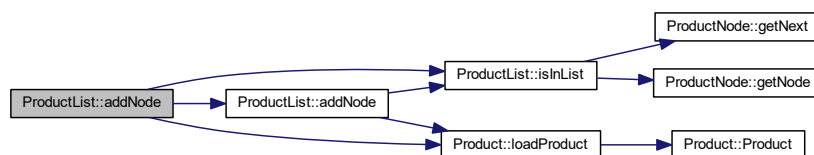
Here is the call graph for this function:



4.7.2.3 addNode() [3/3]

```
void ProductList::addNode (
    ProductList & products,
    std::string shop,
    int amount )
```

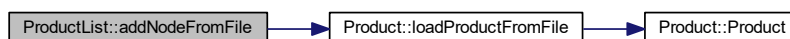
Here is the call graph for this function:



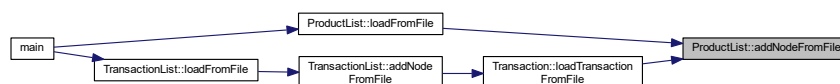
4.7.2.4 addNodeFromFile()

```
void ProductList::addNodeFromFile (
    std::ifstream & file )
```

Here is the call graph for this function:



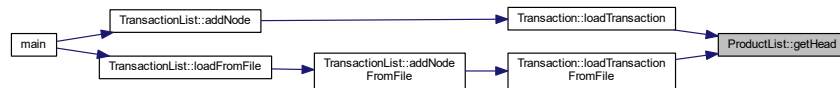
Here is the caller graph for this function:



4.7.2.5 getHead()

```
ProductNode * ProductList::getHead ( )
```

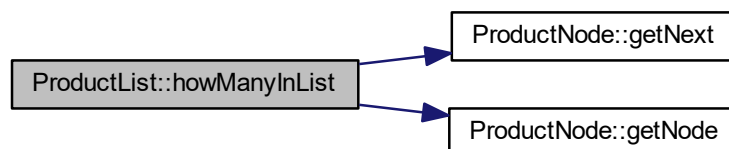
Here is the caller graph for this function:



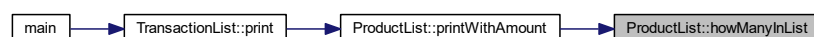
4.7.2.6 howManyInList()

```
int ProductList::howManyInList (
    Product product )
```

Here is the call graph for this function:



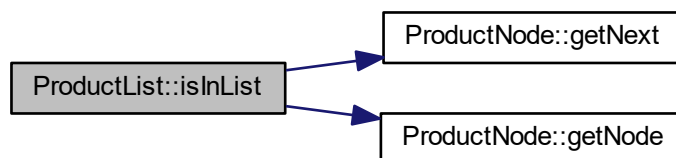
Here is the caller graph for this function:



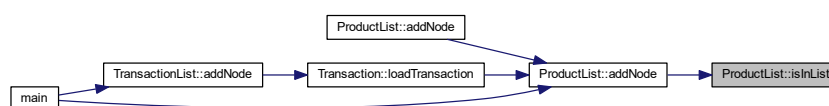
4.7.2.7 isInList()

```
bool ProductList::isInList (
    Product product )
```

Here is the call graph for this function:



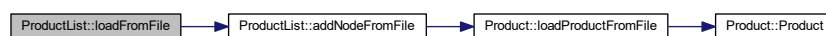
Here is the caller graph for this function:



4.7.2.8 loadFromFile()

```
void ProductList::loadFromFile ( )
```

Here is the call graph for this function:



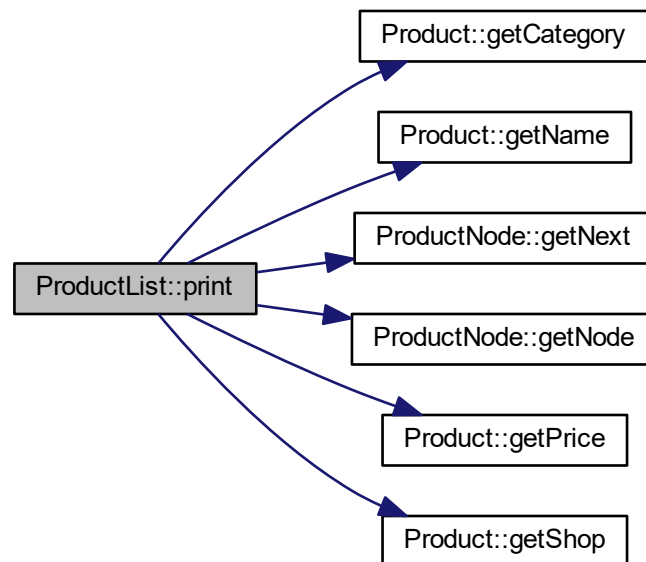
Here is the caller graph for this function:



4.7.2.9 print()

```
void ProductList::print ( )
```

Here is the call graph for this function:



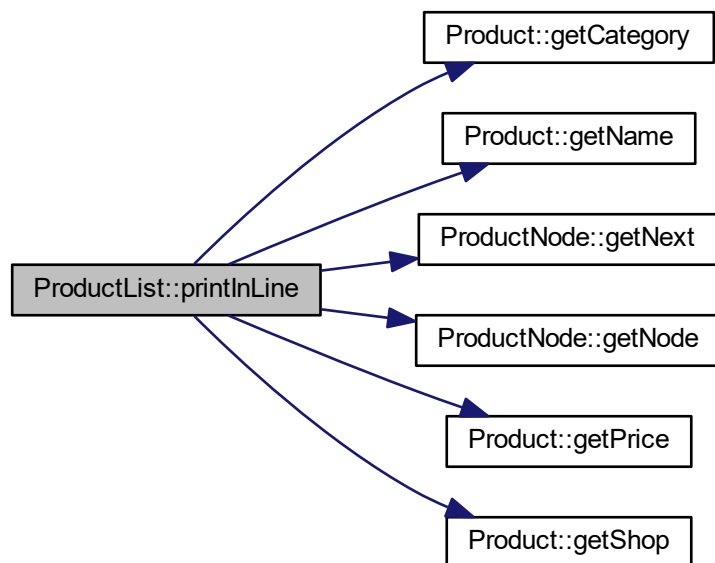
Here is the caller graph for this function:



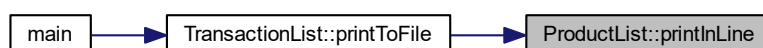
4.7.2.10 printInLine()

```
void ProductList::printInLine (
    std::ofstream & file )
```

Here is the call graph for this function:



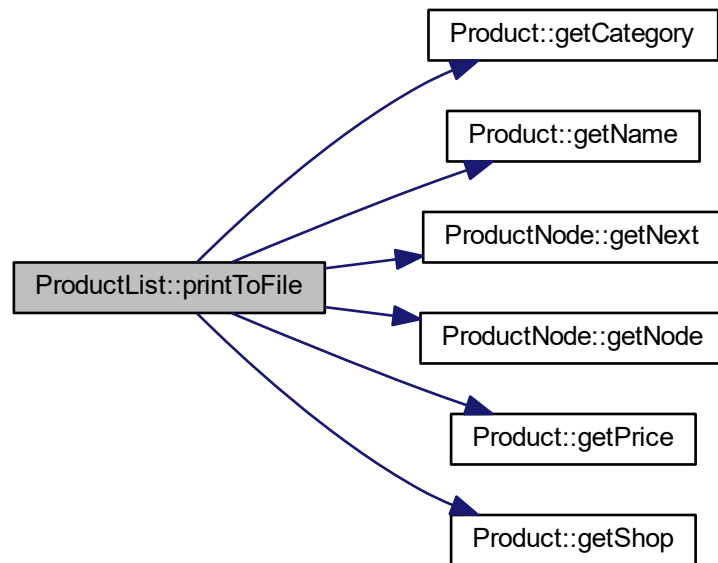
Here is the caller graph for this function:



4.7.2.11 printToFile()

```
void ProductList::printToFile ( )
```

Here is the call graph for this function:



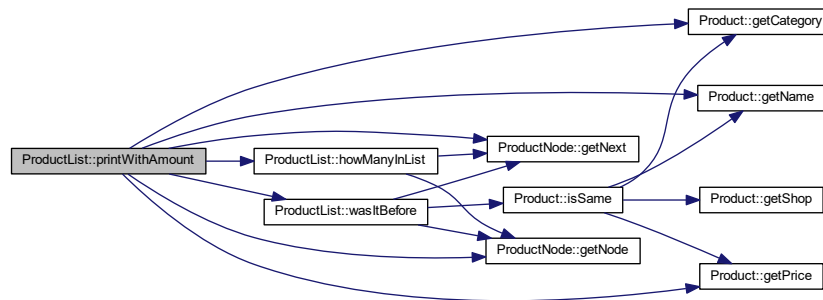
Here is the caller graph for this function:



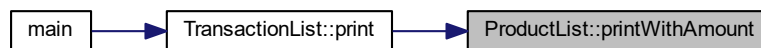
4.7.2.12 printWithAmount()

```
void ProductList::printWithAmount ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.7.2.13 setHead()

```
void ProductList::setHead (
    ProductNode * Node )
```

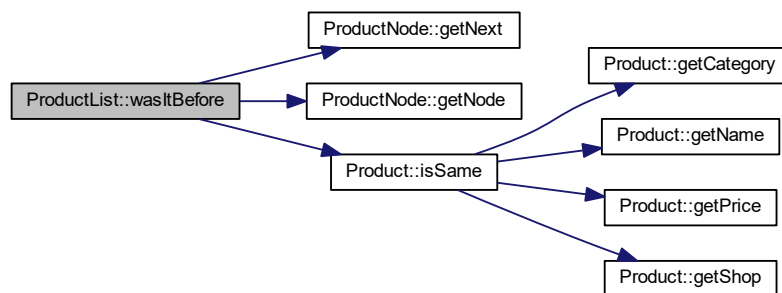
Here is the caller graph for this function:



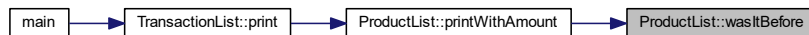
4.7.2.14 wasItBefore()

```
bool ProductList::wasItBefore (
    ProductNode * Node )
```

Here is the call graph for this function:



Here is the caller graph for this function:



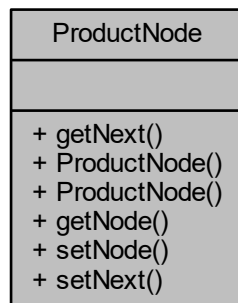
The documentation for this class was generated from the following files:

- [ProductList.h](#)
- [ProductList.cpp](#)

4.8 ProductNode Class Reference

```
#include <ProductNode.h>
```

Collaboration diagram for ProductNode:



Public Member Functions

- [ProductNode *](#) [getNext](#) ()
- [ProductNode](#) ([Product](#) Node)
- [ProductNode](#) ([Product](#) Node, [ProductNode *](#)pNext)
- [Product](#) [getNode](#) ()
- void [setNode](#) ([Product](#))
- void [setNext](#) ([ProductNode *](#))

4.8.1 Constructor & Destructor Documentation

4.8.1.1 ProductNode() [1/2]

```
ProductNode::ProductNode (  
    Product Node ) [inline]
```

4.8.1.2 ProductNode() [2/2]

```
ProductNode::ProductNode (  
    Product Node,  
    ProductNode \* pNext ) [inline]
```

4.8.2 Member Function Documentation

```
ProductNode * ProductNode::getNext ( )
```

```
Product ProductNode::getNode ( )
```

```
void ProductNode::setNext (
    ProductNode * pNext )
```

4.8.2.4 setNode()

```
void ProductNode::setNode (
    Product Product )
```

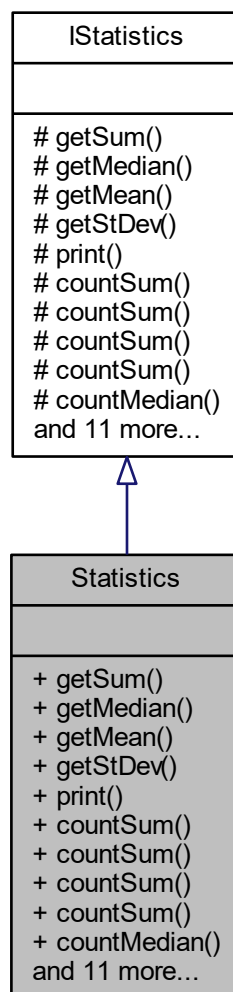
The documentation for this class was generated from the following files:

- [ProductNode.h](#)
- [ProductNode.cpp](#)

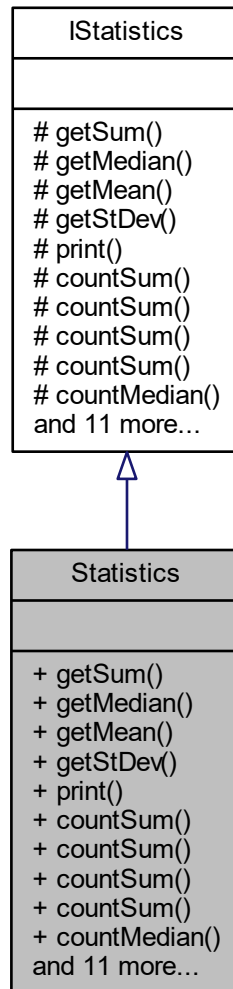
4.9 Statistics Class Reference

```
#include <Statistics.h>
```

Inheritance diagram for Statistics:



Collaboration diagram for Statistics:



Public Member Functions

- double `getSum ()`
- double `getMedian ()`
- double `getMean ()`
- double `getStDev ()`
- void `print ()`
- void `countSum (TransactionNode *)`
- void `countSum (TransactionNode *, Member member)`
- void `countSum (TransactionNode *, Member member, Date start, Date end)`
- void `countSum (TransactionNode *, Date start, Date end)`
- void `countMedian (TransactionNode *)`
- void `countMedian (TransactionNode *, Member member)`
- void `countMedian (TransactionNode *, Member member, Date start, Date end)`

- void [countMedian](#) ([TransactionNode](#) *, [Date](#) start, [Date](#) end)
- void [countMean](#) ([TransactionNode](#) *)
- void [countMean](#) ([TransactionNode](#) *, [Member](#) member)
- void [countMean](#) ([TransactionNode](#) *, [Member](#) member, [Date](#) start, [Date](#) end)
- void [countMean](#) ([TransactionNode](#) *, [Date](#) start, [Date](#) end)
- void [countStandardDeviation](#) ([TransactionNode](#) *)
- void [countStandardDeviation](#) ([TransactionNode](#) *, [Member](#) member)
- void [countStandardDeviation](#) ([TransactionNode](#) *, [Member](#) member, [Date](#) start, [Date](#) end)
- void [countStandardDeviation](#) ([TransactionNode](#) *, [Date](#) start, [Date](#) end)

Additional Inherited Members

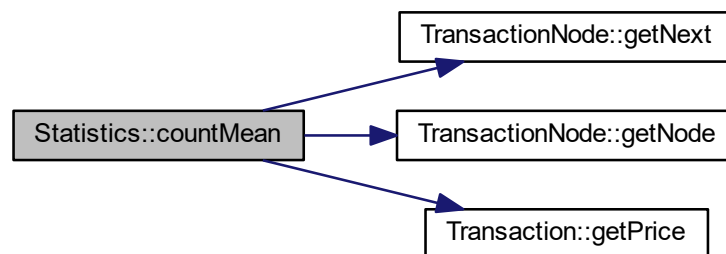
4.9.1 Member Function Documentation

4.9.1.1 [countMean\(\)](#) [1/4]

```
void Statistics::countMean (
    TransactionNode * pHead )    [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:



Here is the caller graph for this function:

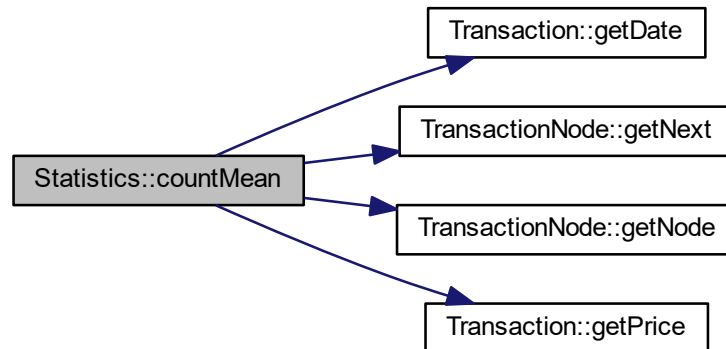


4.9.1.2 countMean() [2/4]

```
void Statistics::countMean (  
    TransactionNode * pHead,  
    Date start,  
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

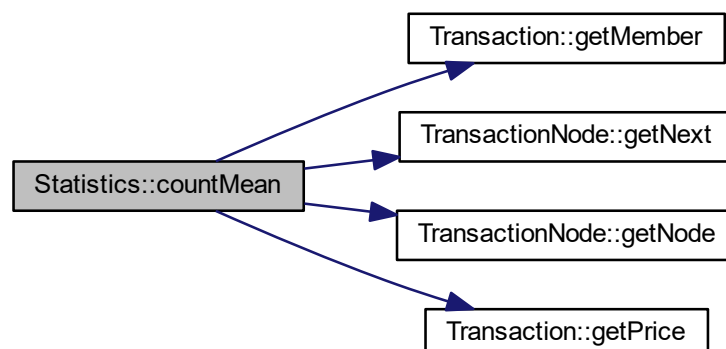


4.9.1.3 countMean() [3/4]

```
void Statistics::countMean (  
    TransactionNode * pHead,  
    Member member ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

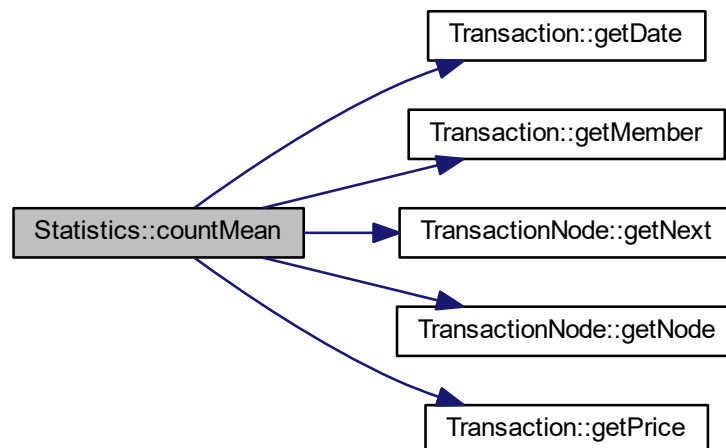


4.9.1.4 countMean() [4/4]

```
void Statistics::countMean (
    TransactionNode * pHead,
    Member member,
    Date start,
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

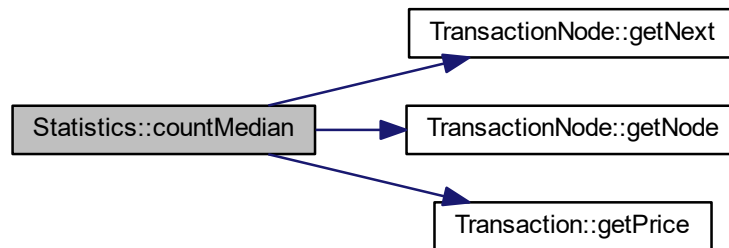


4.9.1.5 countMedian() [1/4]

```
void Statistics::countMedian (
    TransactionNode * pHead ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:



Here is the caller graph for this function:

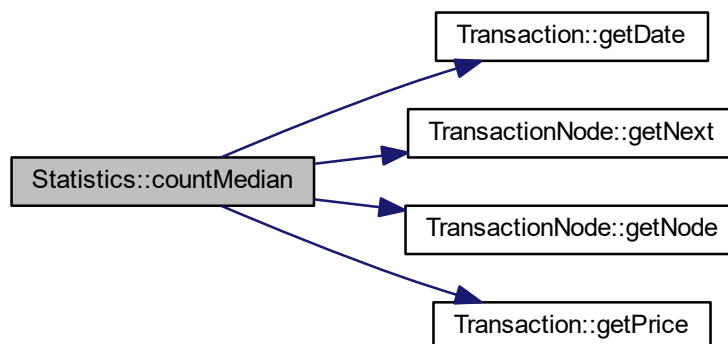


4.9.1.6 countMedian() [2/4]

```
void Statistics::countMedian (  
    TransactionNode * pHead,  
    Date start,  
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

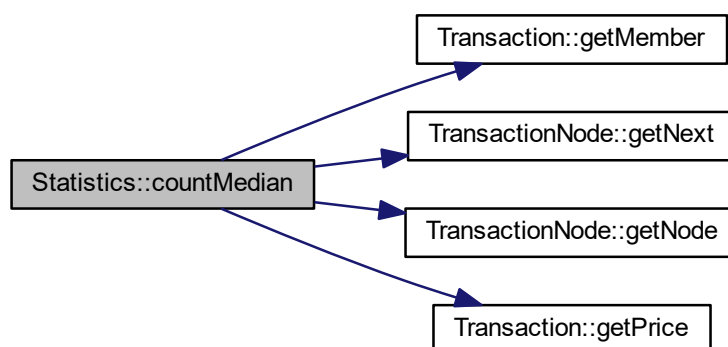


4.9.1.7 countMedian() [3/4]

```
void Statistics::countMedian (  
    TransactionNode * pHead,  
    Member member ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

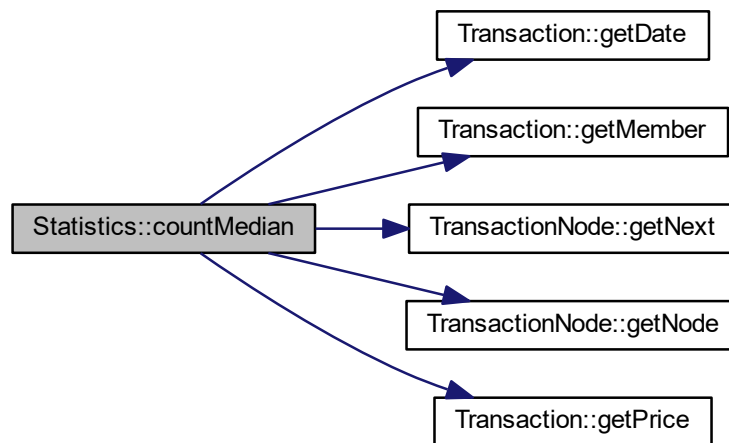


4.9.1.8 countMedian() [4/4]

```
void Statistics::countMedian (
    TransactionNode * pHead,
    Member member,
    Date start,
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

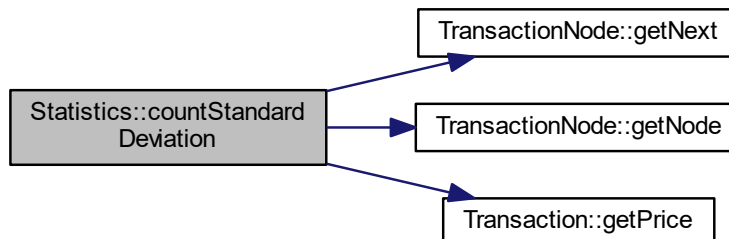


4.9.1.9 countStandardDeviation() [1/4]

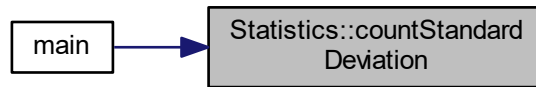
```
void Statistics::countStandardDeviation (
    TransactionNode * pHead ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:



Here is the caller graph for this function:

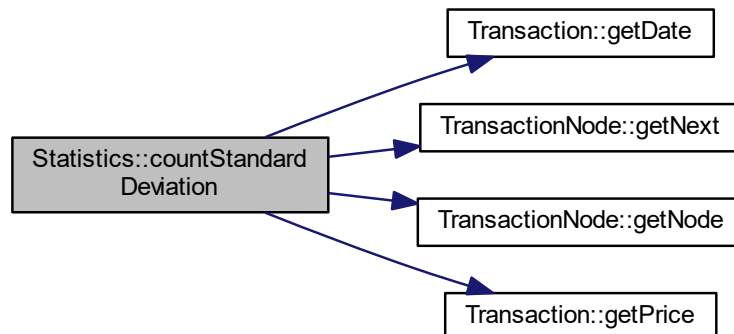


4.9.1.10 countStandardDeviation() [2/4]

```
void Statistics::countStandardDeviation (
    TransactionNode * pHead,
    Date start,
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

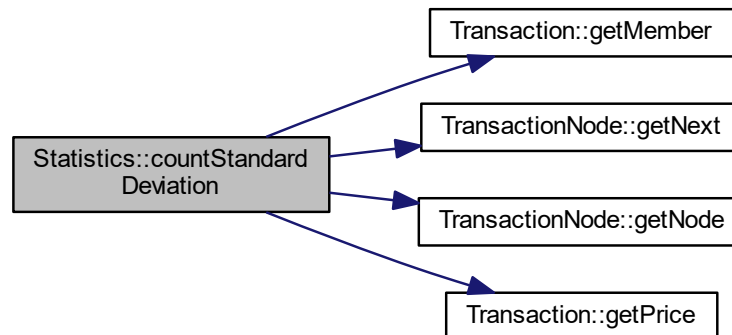


4.9.1.11 countStandardDeviation() [3/4]

```
void Statistics::countStandardDeviation (
    TransactionNode * pHead,
    Member member ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

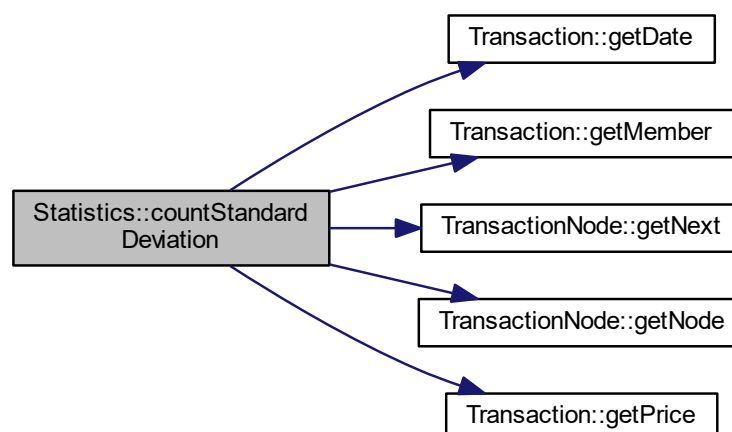


4.9.1.12 countStandardDeviation() [4/4]

```
void Statistics::countStandardDeviation (
    TransactionNode * pHead,
    Member member,
    Date start,
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

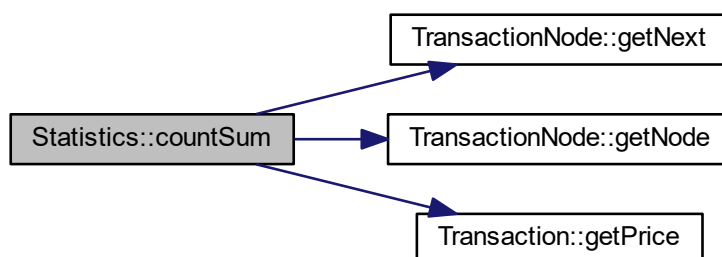


4.9.1.13 countSum() [1/4]

```
void Statistics::countSum (  
    TransactionNode * pHead )    [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:



Here is the caller graph for this function:

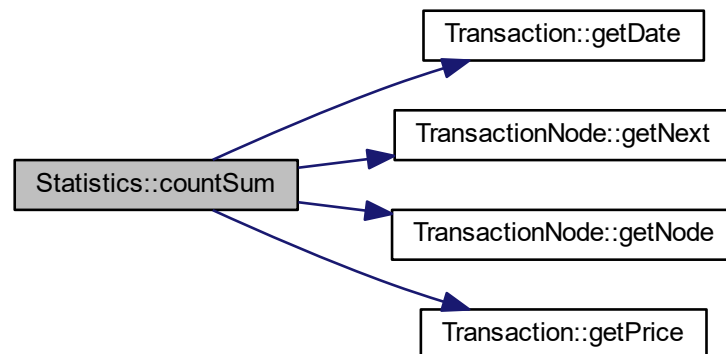


4.9.1.14 countSum() [2/4]

```
void Statistics::countSum (  
    TransactionNode * pHead,  
    Date start,  
    Date end )    [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

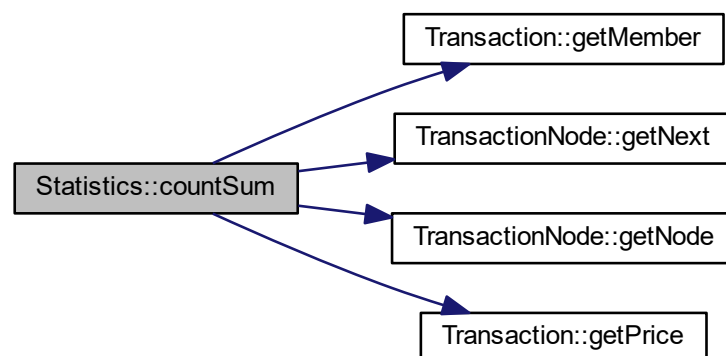


4.9.1.15 countSum() [3/4]

```
void Statistics::countSum (  
    TransactionNode * pHead,  
    Member member ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

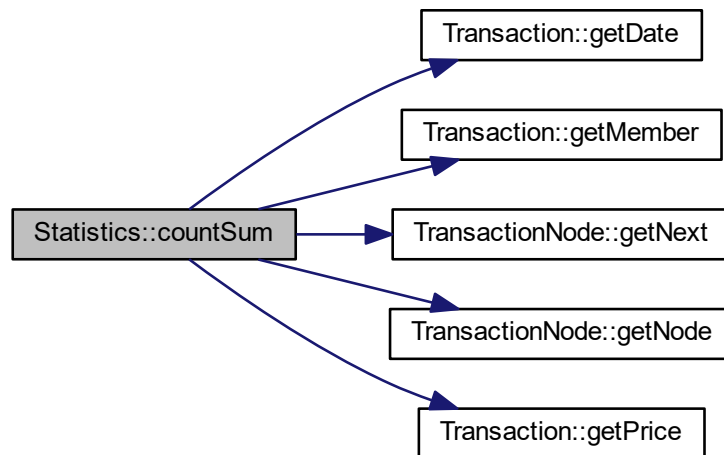


4.9.1.16 countSum() [4/4]

```
void Statistics::countSum (
    TransactionNode * pHead,
    Member member,
    Date start,
    Date end ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:

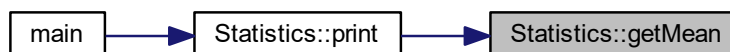


4.9.1.17 getMean()

```
double Statistics::getMean ( ) [virtual]
```

Implements [IStatistics](#).

Here is the caller graph for this function:



4.9.1.18 getMedian()

```
double Statistics::getMedian ( ) [virtual]
```

Implements [IStatistics](#).

Here is the caller graph for this function:

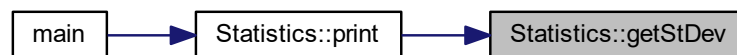


4.9.1.19 getStDev()

```
double Statistics::getStDev ( ) [virtual]
```

Implements [IStatistics](#).

Here is the caller graph for this function:



4.9.1.20 getSum()

```
double Statistics::getSum ( ) [virtual]
```

Implements [IStatistics](#).

Here is the caller graph for this function:

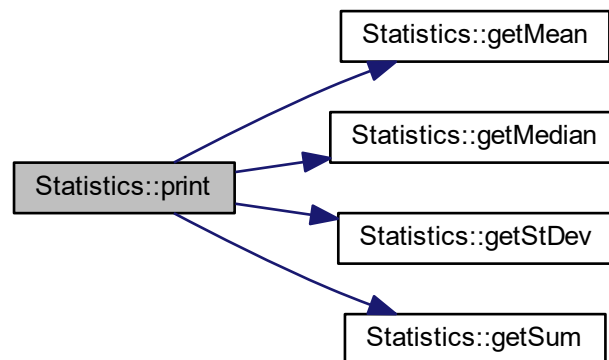


4.9.1.21 print()

```
void Statistics::print ( ) [virtual]
```

Implements [IStatistics](#).

Here is the call graph for this function:



Here is the caller graph for this function:



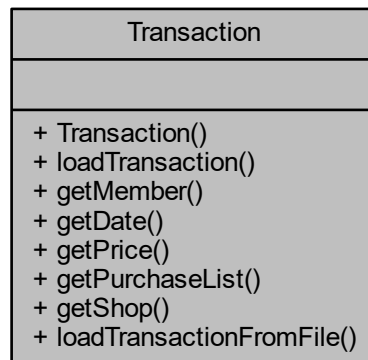
The documentation for this class was generated from the following files:

- [Statistics.h](#)
- [Statistics.cpp](#)

4.10 Transaction Class Reference

```
#include <Transaction.h>
```

Collaboration diagram for Transaction:



Public Member Functions

- [Transaction](#) ([Date](#) date, [Member](#) member, double price, [ProductList](#) *PurchaseList, std::string shop)
- [Transaction](#) loadTransaction ([MemberList](#) &members, [ProductList](#) &products)
- [Member](#) getMember ()
- [Date](#) getDate ()
- double getPrice ()
- [ProductList](#) * getPurchaseList ()
- std::string getShop ()

Static Public Member Functions

- static [Transaction](#) loadTransactionFromFile (std::ifstream &file)

4.10.1 Constructor & Destructor Documentation

4.10.1.1 Transaction()

```

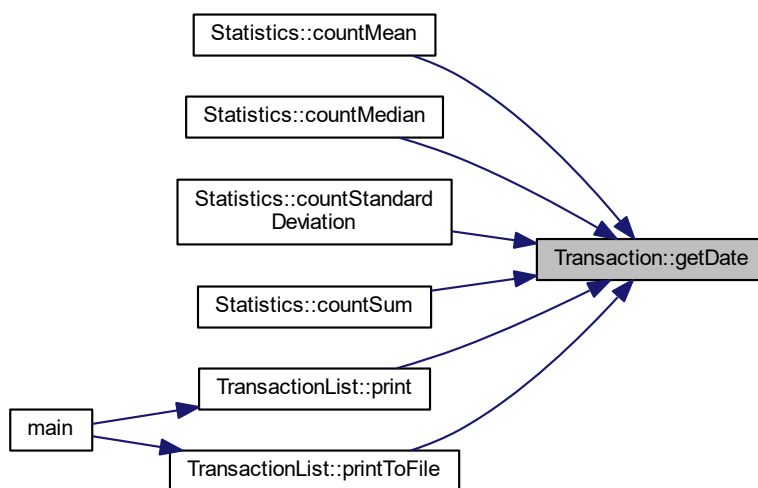
Transaction::Transaction (
    Date date,
    Member member,
    double price,
    ProductList * PurchaseList,
    std::string shop ) [inline]
  
```

4.10.2 Member Function Documentation

4.10.2.1 getDate()

```
Date Transaction::getDate ( )
```

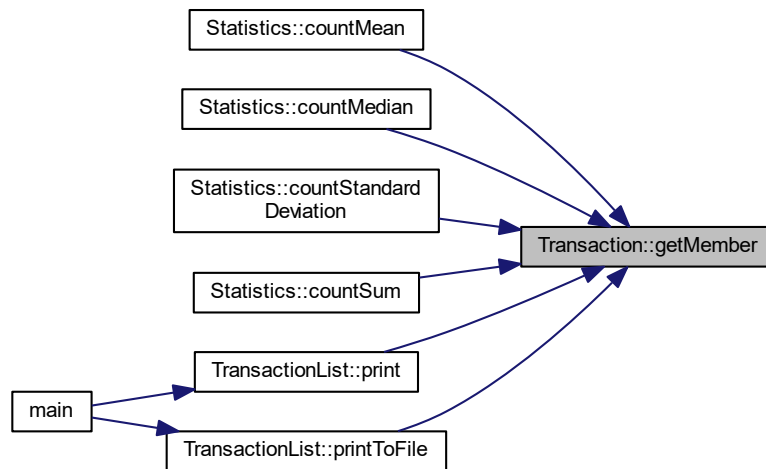
Here is the caller graph for this function:



4.10.2.2 getMember()

```
Member Transaction::getMember ( )
```

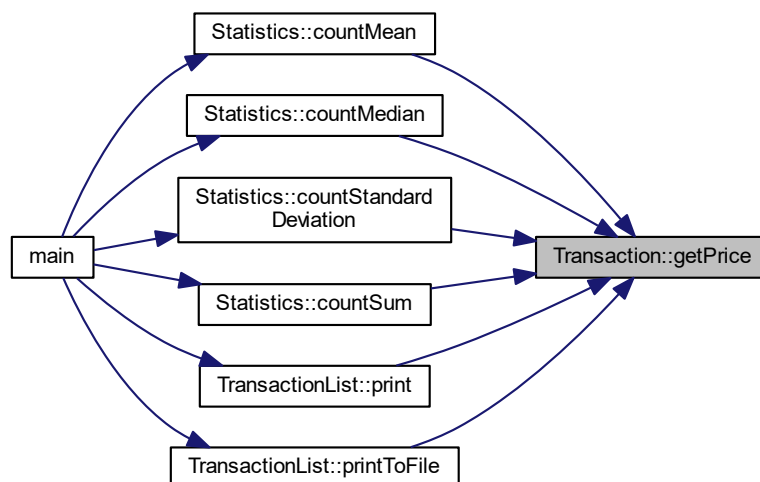
Here is the caller graph for this function:



4.10.2.3 getPrice()

```
double Transaction::getPrice ( )
```

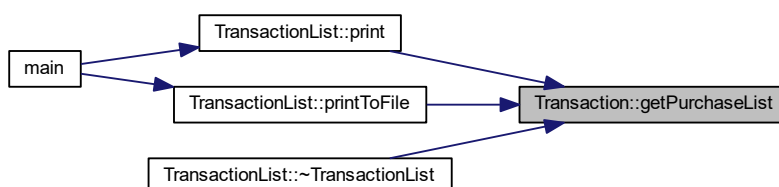
Here is the caller graph for this function:



4.10.2.4 getPurchaseList()

```
ProductList * Transaction::getPurchaseList ( )
```

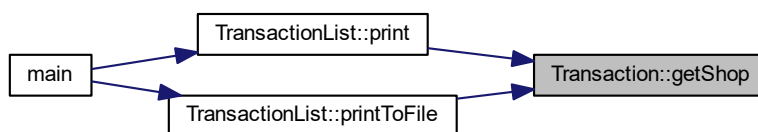
Here is the caller graph for this function:



4.10.2.5 getShop()

```
std::string Transaction::getShop ( )
```

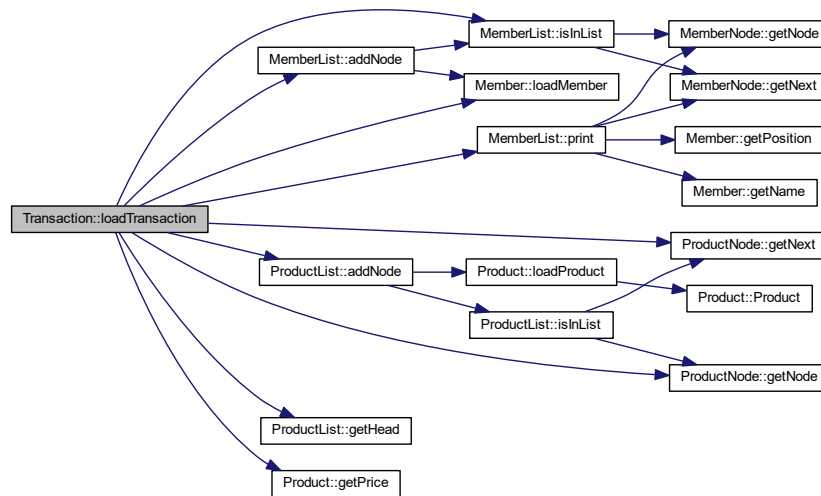
Here is the caller graph for this function:



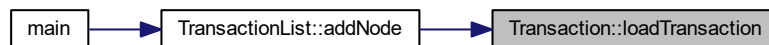
4.10.2.6 loadTransaction()

```
Transaction Transaction::loadTransaction (
    MemberList & members,
    ProductList & products )
```

Here is the call graph for this function:



Here is the caller graph for this function:

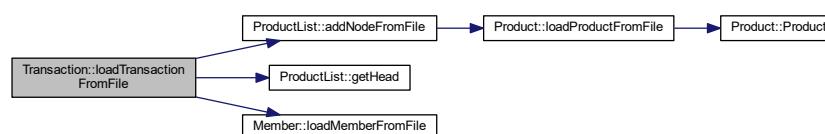


4.10.2.7 loadTransactionFromFile()

```

Transaction Transaction::loadTransactionFromFile (
    std::ifstream & file ) [static]
  
```

Here is the call graph for this function:



Here is the caller graph for this function:



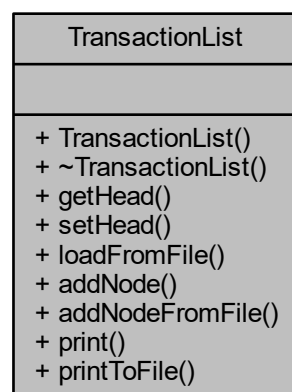
The documentation for this class was generated from the following files:

- [Transaction.h](#)
- [Transaction.cpp](#)

4.11 TransactionList Class Reference

```
#include <TransactionList.h>
```

Collaboration diagram for TransactionList:



Public Member Functions

- [TransactionList](#) ()
- [~TransactionList](#) ()
- [TransactionNode *](#) [getHead](#) ()
- void [setHead](#) ([TransactionNode *](#))
- void [loadFromFile](#) ()
- void [addNode](#) ([MemberList](#) &members, [ProductList](#) &products)
- void [addNodeFromFile](#) (std::ifstream &)
- void [print](#) ()
- void [printToFile](#) ()

4.11.1 Constructor & Destructor Documentation

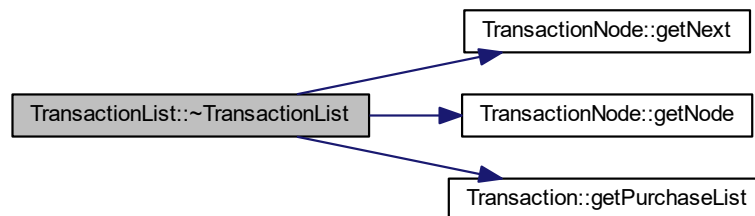
4.11.1.1 TransactionList()

```
TransactionList::TransactionList ( )
```

4.11.1.2 ~TransactionList()

```
TransactionList::~~TransactionList ( )
```

Here is the call graph for this function:

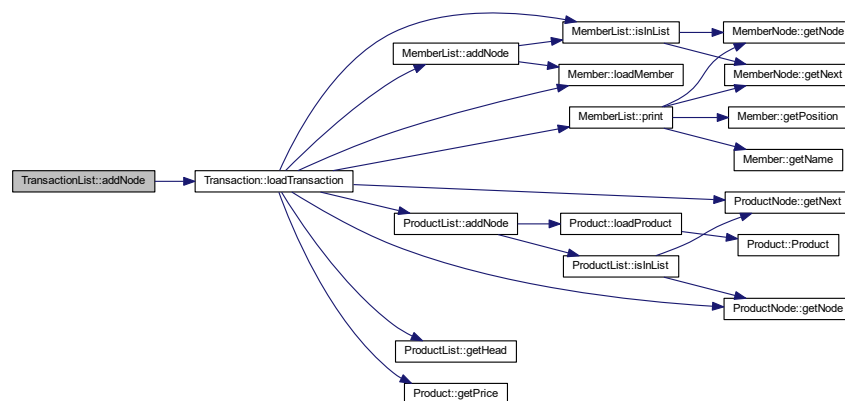


4.11.2 Member Function Documentation

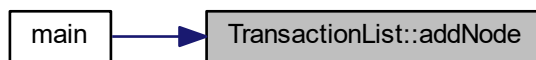
4.11.2.1 addNode()

```
void TransactionList::addNode (
    MemberList & members,
    ProductList & products )
```

Here is the call graph for this function:



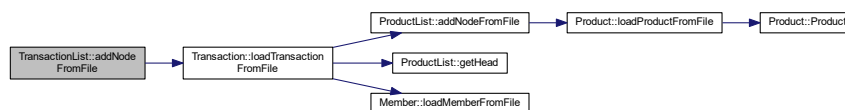
Here is the caller graph for this function:



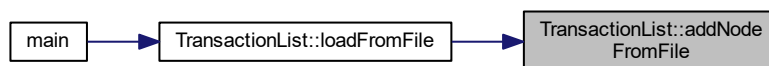
4.11.2.2 addNodeFromFile()

```
void TransactionList::addNodeFromFile (
    std::ifstream & file )
```

Here is the call graph for this function:



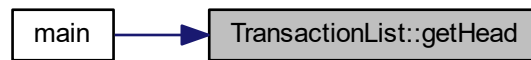
Here is the caller graph for this function:



4.11.2.3 getHead()

```
TransactionNode * TransactionList::getHead ( )
```

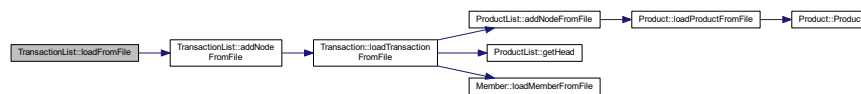
Here is the caller graph for this function:



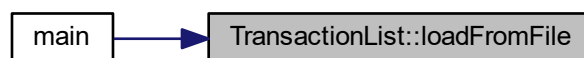
4.11.2.4 loadFromFile()

```
void TransactionList::loadFromFile ( )
```

Here is the call graph for this function:



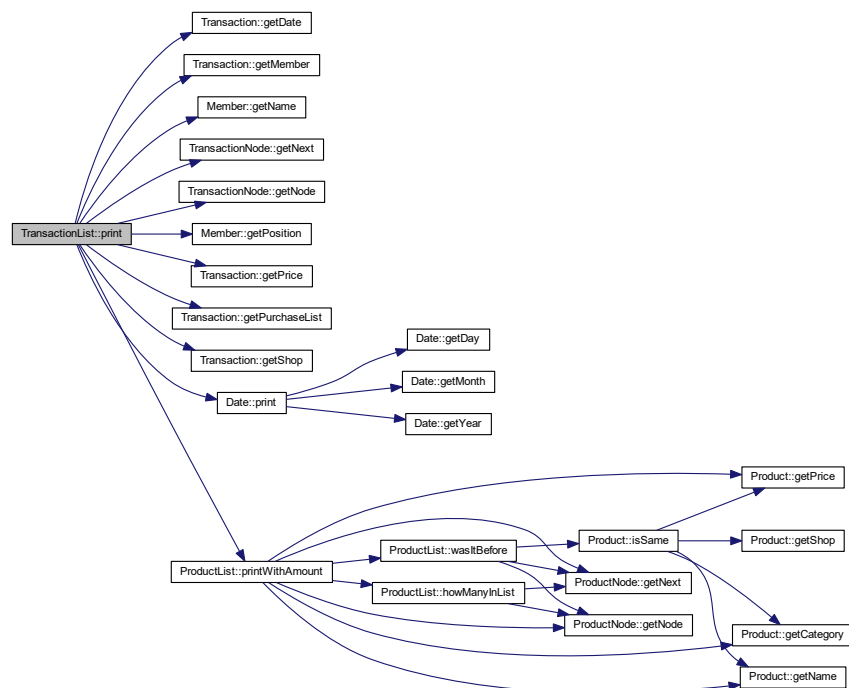
Here is the caller graph for this function:



4.11.2.5 print()

```
void TransactionList::print ( )
```

Here is the call graph for this function:



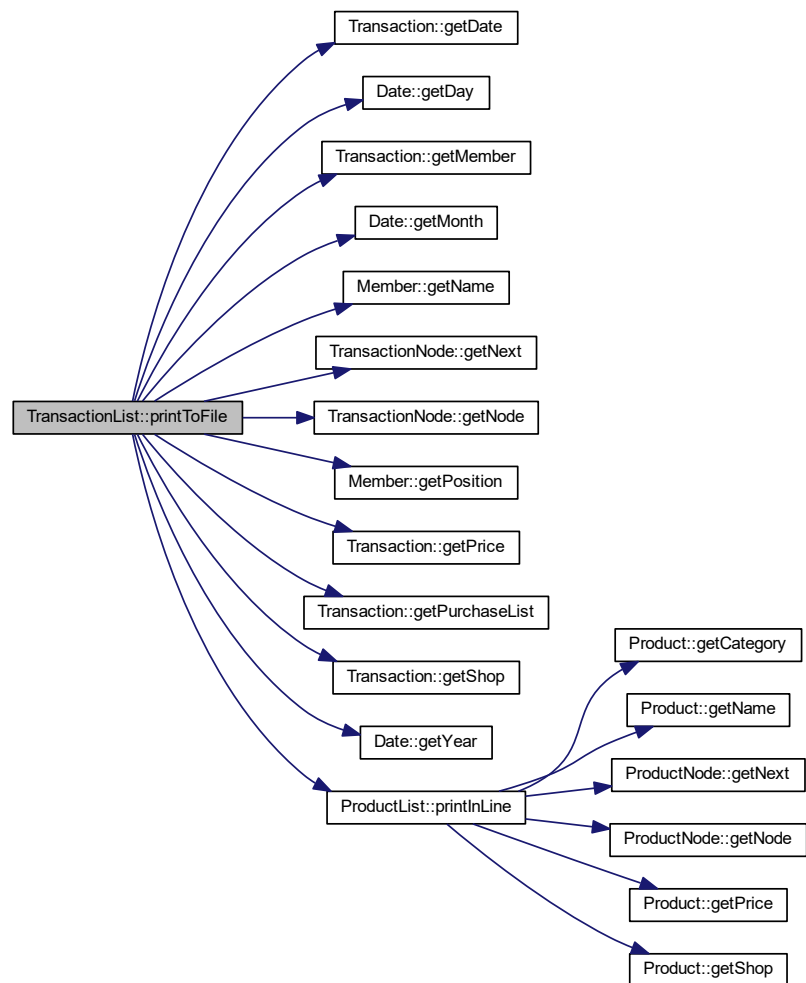
Here is the caller graph for this function:



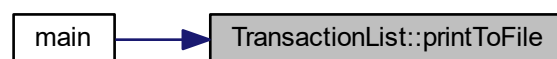
4.11.2.6 printToFile()

```
void TransactionList::printToFile ( )
```

Here is the call graph for this function:



Here is the caller graph for this function:



4.11.2.7 setHead()

```

void TransactionList::setHead (
    TransactionNode * pHead )
  
```

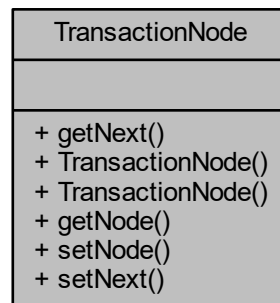
The documentation for this class was generated from the following files:

- [TransactionList.h](#)
- [TransactionList.cpp](#)

4.12 TransactionNode Class Reference

```
#include <TransactionNode.h>
```

Collaboration diagram for TransactionNode:



Public Member Functions

- [TransactionNode * getNext \(\)](#)
- [TransactionNode \(Transaction Node\)](#)
- [TransactionNode \(Transaction Node, TransactionNode *pNext\)](#)
- [TransactionNode getNode \(\)](#)
- void [setNode \(Transaction\)](#)
- void [setNext \(TransactionNode *\)](#)

4.12.1 Constructor & Destructor Documentation

4.12.1.1 TransactionNode() [1/2]

```
TransactionNode::TransactionNode (  
    Transaction Node ) [inline]
```

4.12.1.2 TransactionNode() [2/2]

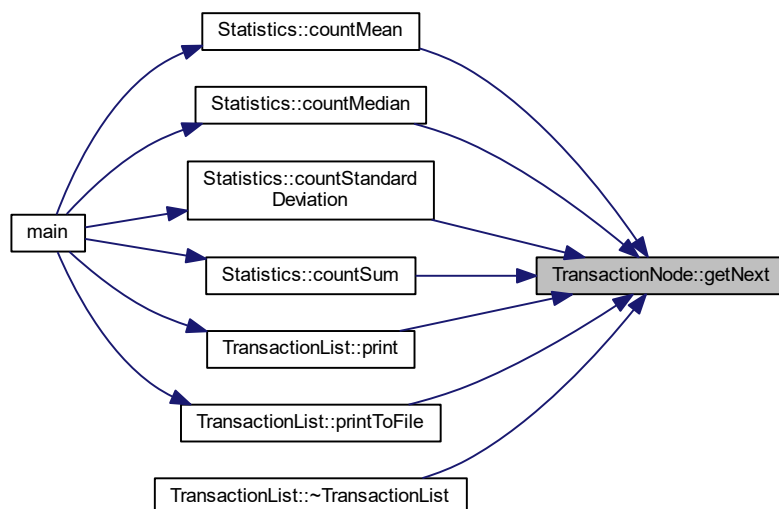
```
TransactionNode::TransactionNode (
    Transaction Node,
    TransactionNode * pNext ) [inline]
```

4.12.2 Member Function Documentation

4.12.2.1 getNext()

```
TransactionNode * TransactionNode::getNext ( )
```

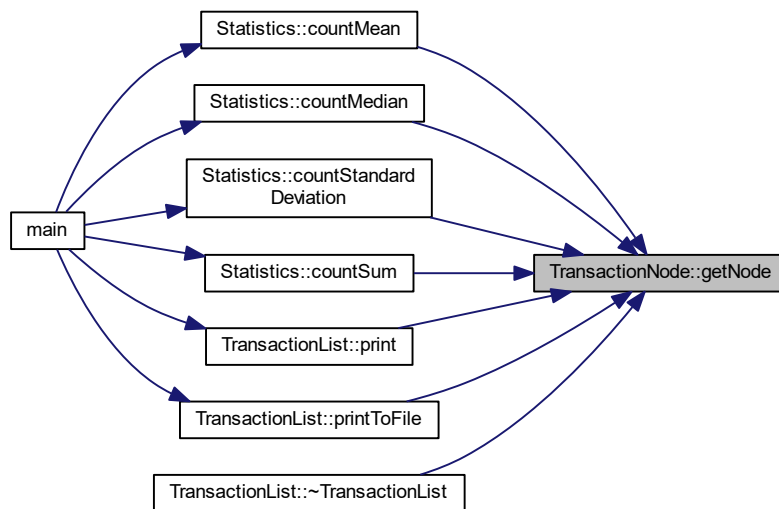
Here is the caller graph for this function:



4.12.2.2 getNode()

```
Transaction TransactionNode::getNode ( )
```

Here is the caller graph for this function:



4.12.2.3 setNext()

```
void TransactionNode::setNext (
    TransactionNode * pNext )
```

4.12.2.4 setNode()

```
void TransactionNode::setNode (
    Transaction Transaction )
```

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

- [TransactionNode.h](#)
- [TransactionNode.cpp](#)

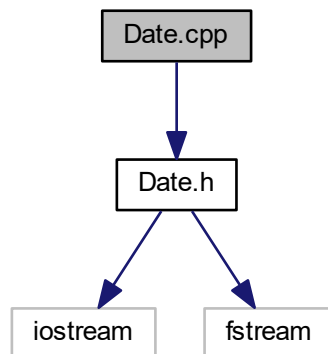
Chapter 5

File Documentation

5.1 Date.cpp File Reference

```
#include "Date.h"
```

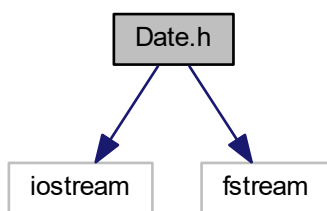
Include dependency graph for Date.cpp:



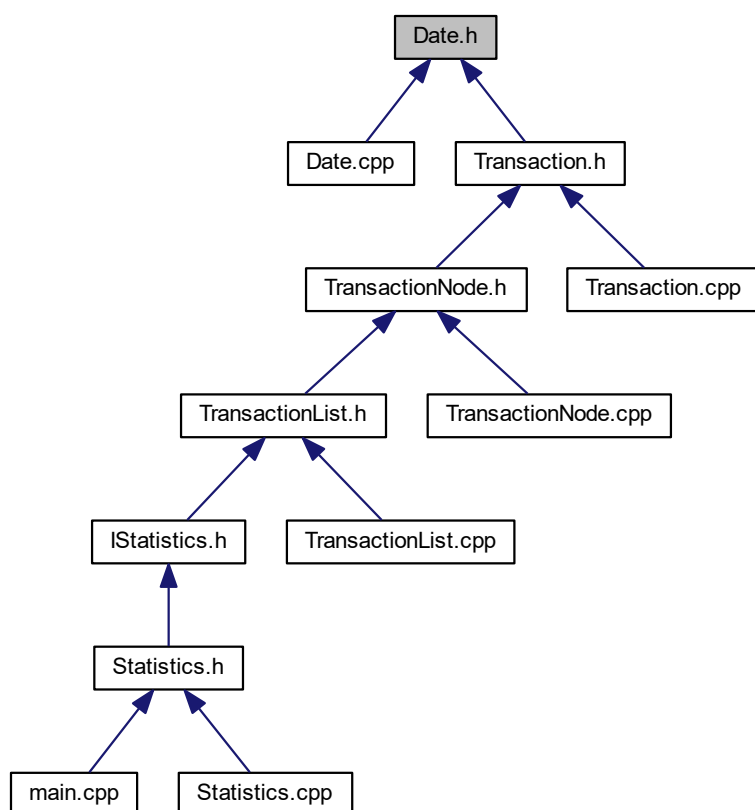
5.2 Date.h File Reference

```
#include <iostream>
#include <fstream>
```

Include dependency graph for Date.h:



This graph shows which files directly or indirectly include this file:

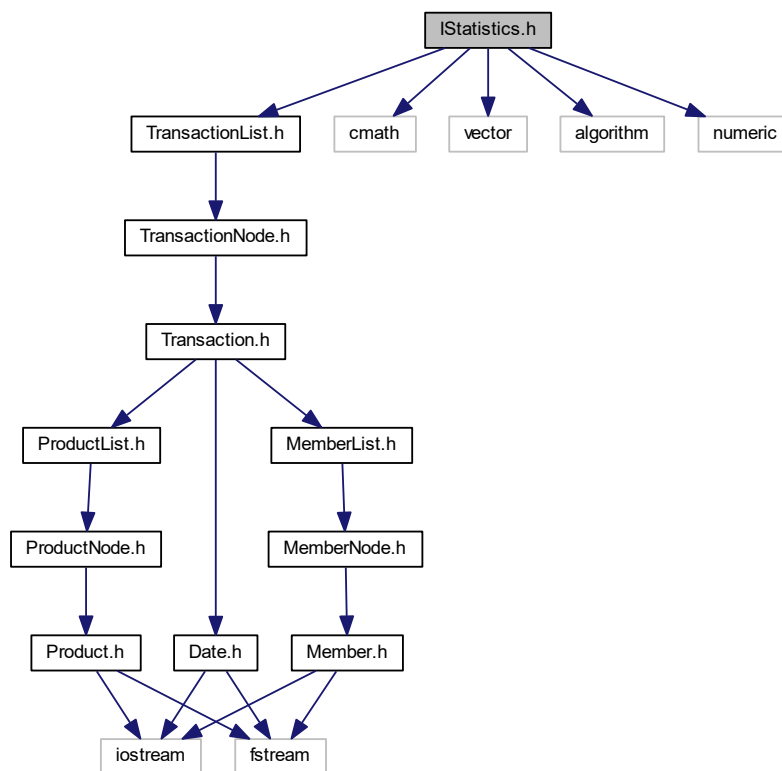


Classes

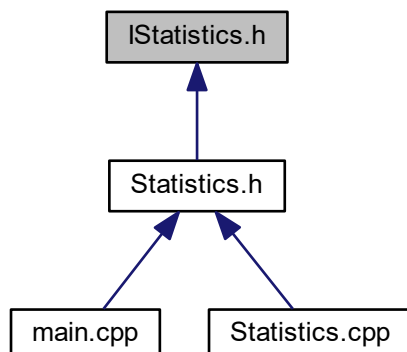
- class [Date](#)

5.3 IStatistics.h File Reference

```
#include "TransactionList.h"  
#include <cmath>  
#include <vector>  
#include <algorithm>  
#include <numeric>  
Include dependency graph for IStatistics.h:
```



This graph shows which files directly or indirectly include this file:



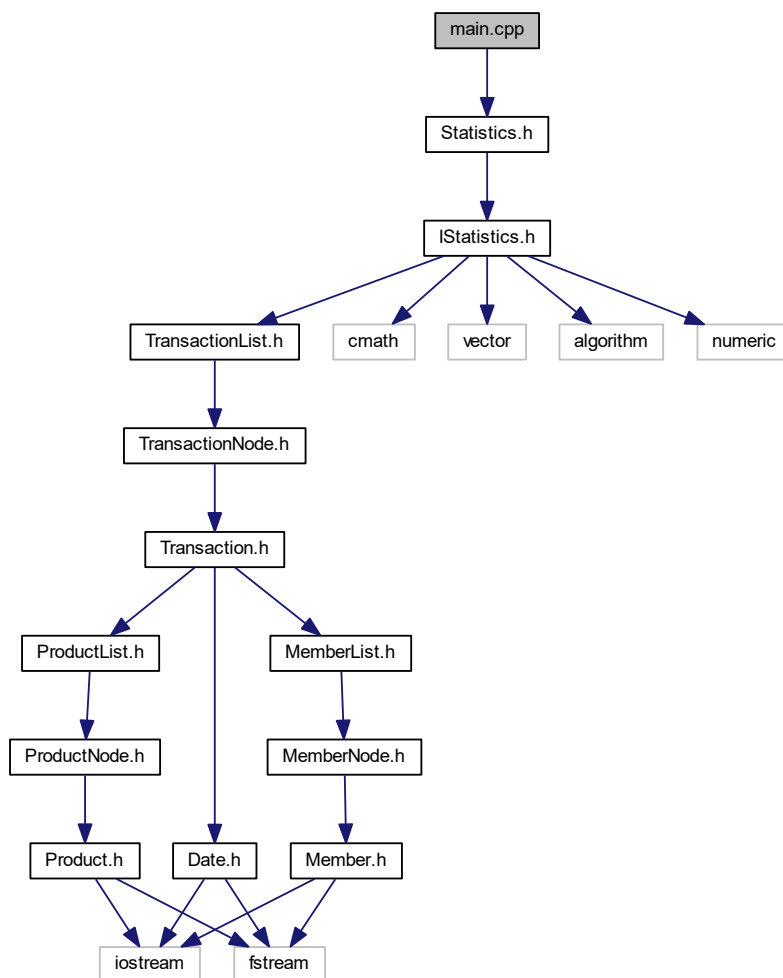
Classes

- class [IStatistics](#)

5.4 main.cpp File Reference

```
#include "Statistics.h"
```

Include dependency graph for main.cpp:



Functions

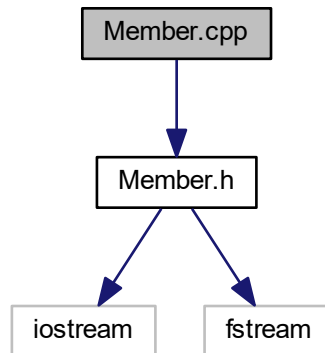
- int [main](#) ()

5.4.1 Function Documentation

5.4.1.1 main()

```
int main ( )
```

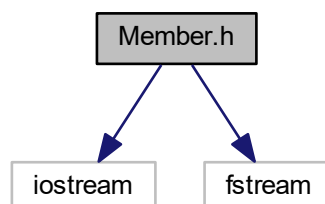

Include dependency graph for Member.cpp:



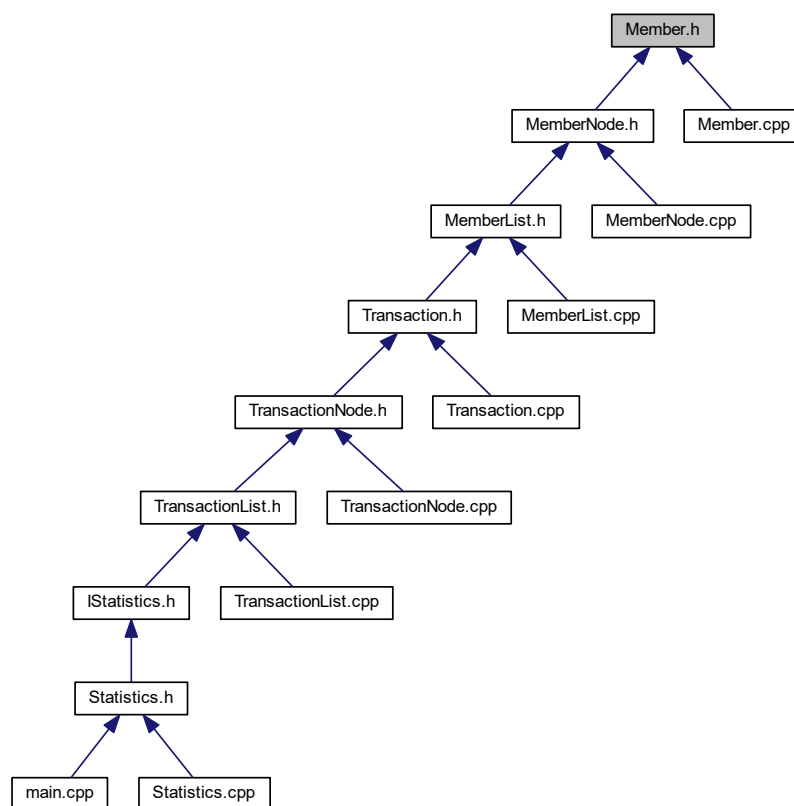
5.6 Member.h File Reference

```
#include <iostream>
#include <fstream>
```

Include dependency graph for Member.h:



This graph shows which files directly or indirectly include this file:



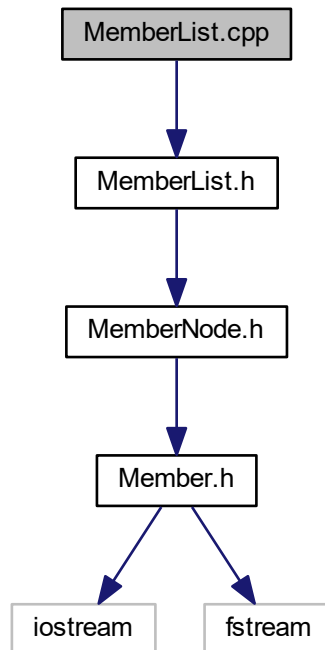
Classes

- class [Member](#)

5.7 MemberList.cpp File Reference

```
#include "MemberList.h"
```

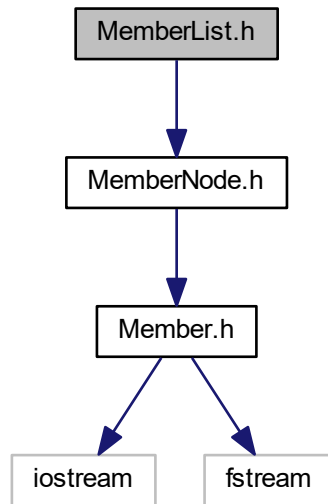

Include dependency graph for MemberList.cpp:



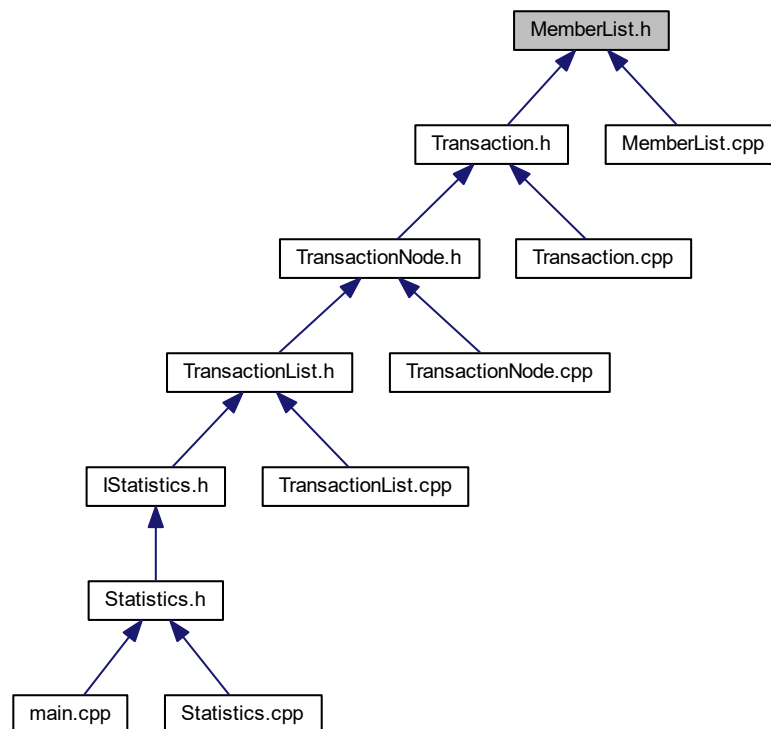
5.8 MemberList.h File Reference

```
#include "MemberNode.h"
```

Include dependency graph for MemberList.h:



This graph shows which files directly or indirectly include this file:



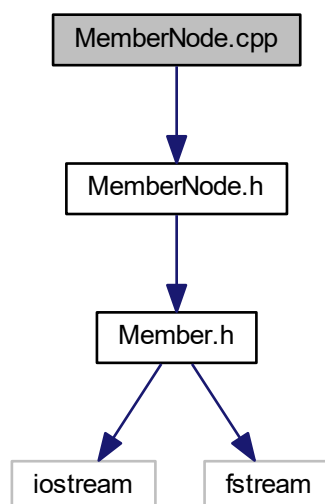
Classes

- class [MemberList](#)

5.9 MemberNode.cpp File Reference

```
#include "MemberNode.h"
```

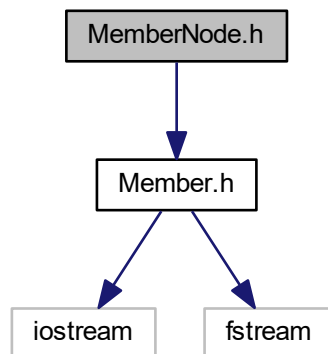
Include dependency graph for MemberNode.cpp:



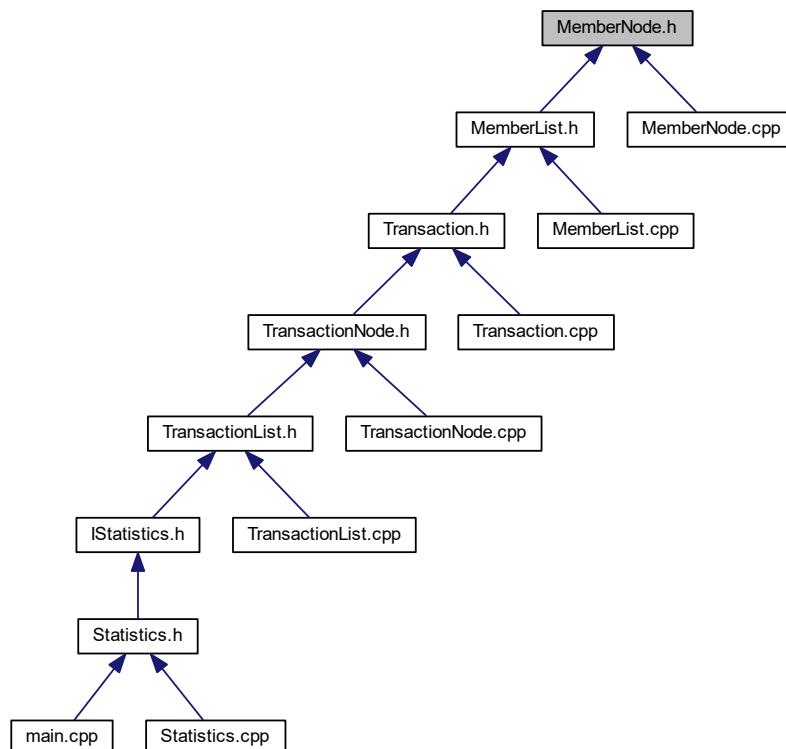
5.10 MemberNode.h File Reference

```
#include "Member.h"
```

Include dependency graph for MemberNode.h:



This graph shows which files directly or indirectly include this file:



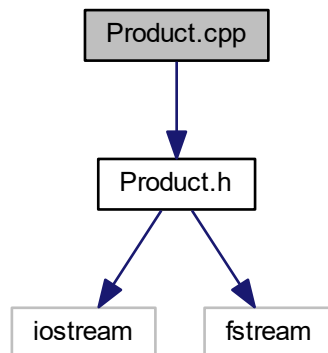
Classes

- class [MemberNode](#)

5.11 Product.cpp File Reference

```
#include "Product.h"
```

Include dependency graph for Product.cpp:

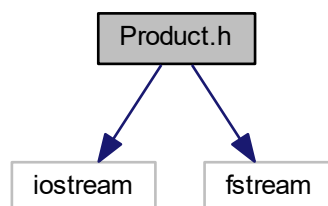


5.12 Product.h File Reference

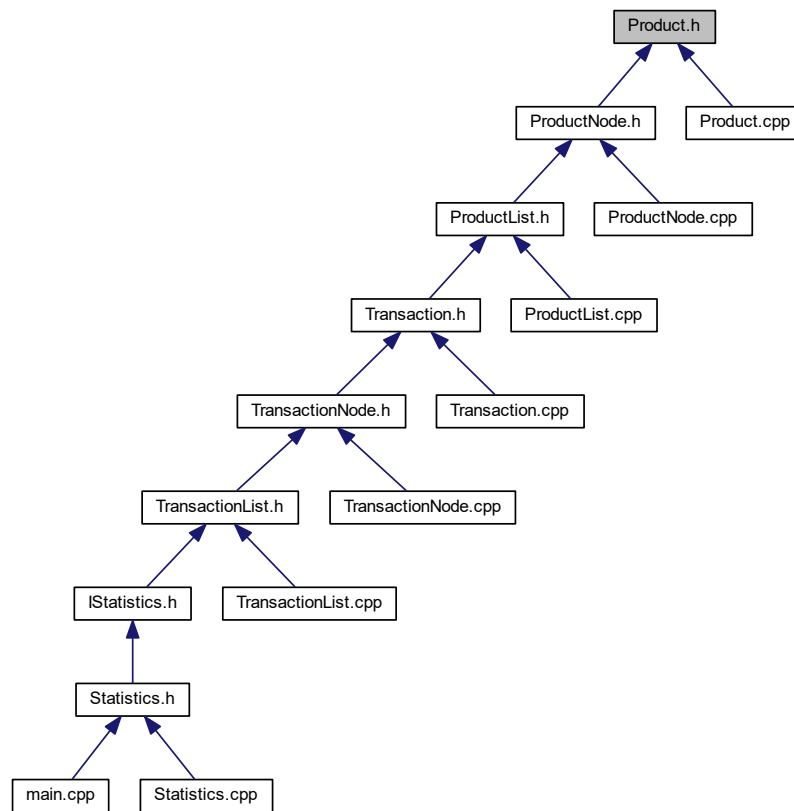
```
#include <iostream>
```

```
#include <fstream>
```

Include dependency graph for Product.h:



This graph shows which files directly or indirectly include this file:



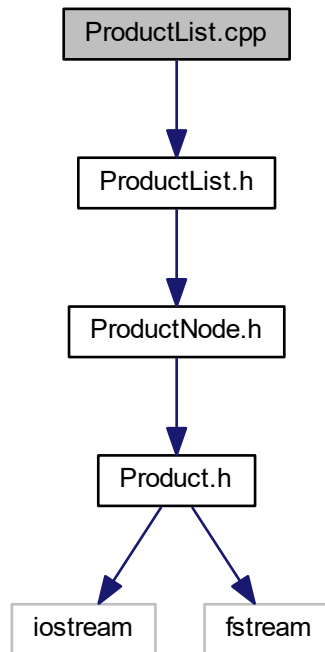
Classes

- class [Product](#)

5.13 ProductList.cpp File Reference

```
#include "ProductList.h"
```

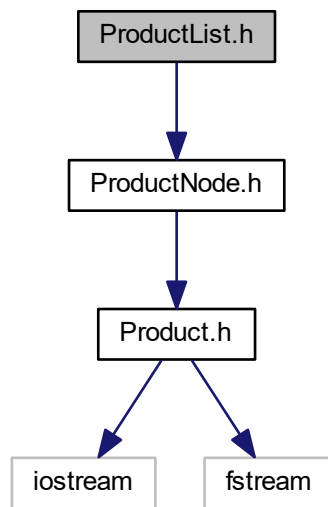
Include dependency graph for ProductList.cpp:



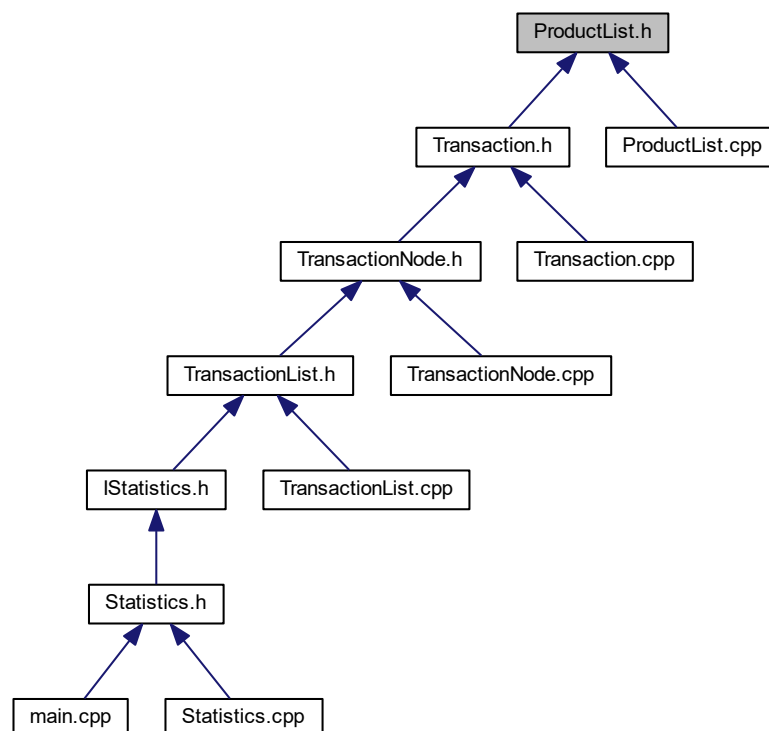
5.14 ProductList.h File Reference

```
#include "ProductNode.h"
```

Include dependency graph for ProductList.h:



This graph shows which files directly or indirectly include this file:



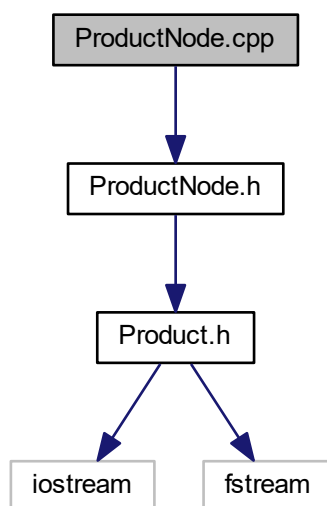
Classes

- class [ProductList](#)

5.15 ProductNode.cpp File Reference

```
#include "ProductNode.h"
```

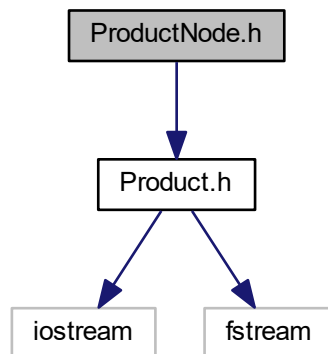
Include dependency graph for ProductNode.cpp:



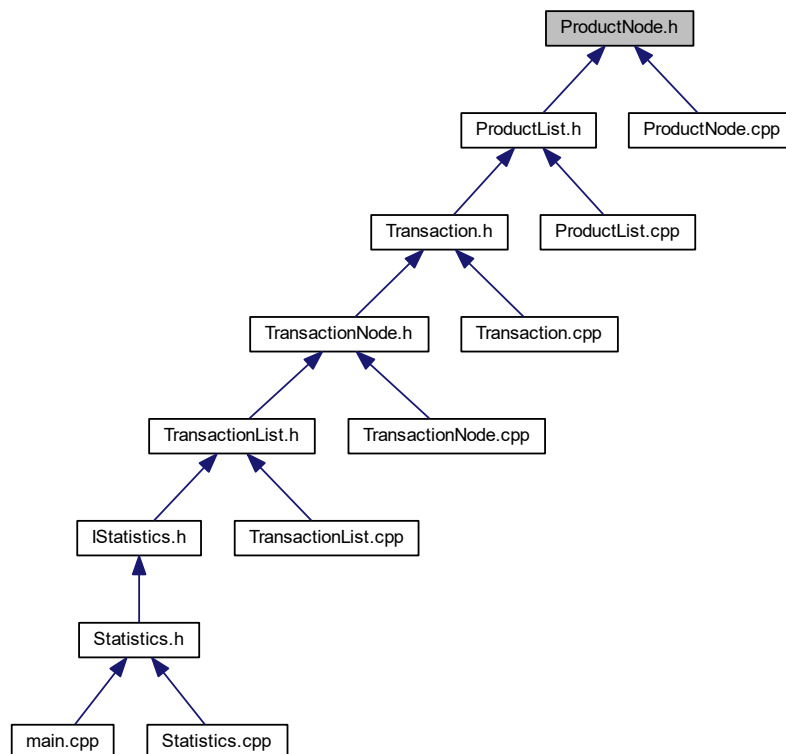
5.16 ProductNode.h File Reference

```
#include "Product.h"
```

Include dependency graph for ProductNode.h:



This graph shows which files directly or indirectly include this file:



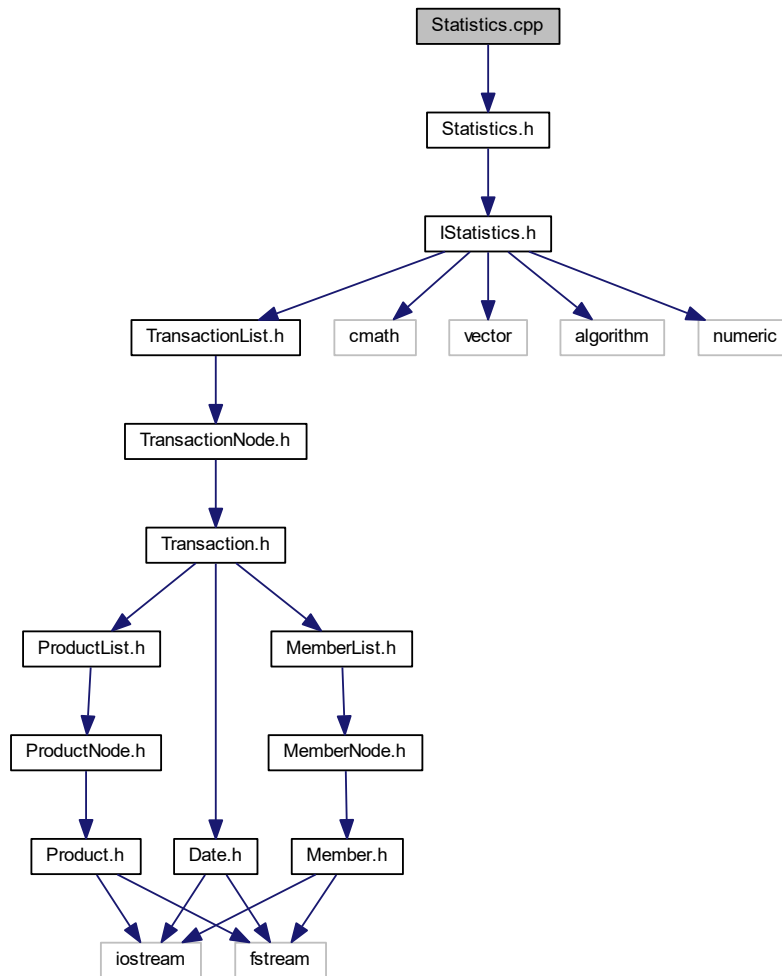
Classes

- class [ProductNode](#)

5.17 Statistics.cpp File Reference

```
#include "Statistics.h"
```

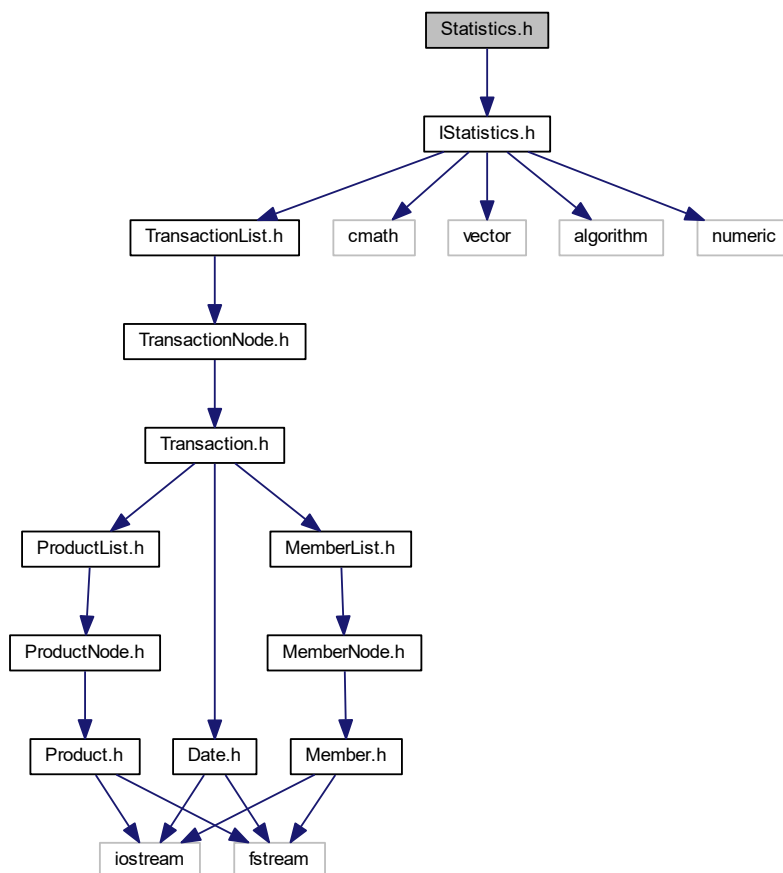
Include dependency graph for Statistics.cpp:



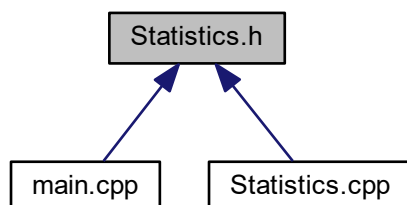
5.18 Statistics.h File Reference

```
#include "IStatistics.h"
```

Include dependency graph for Statistics.h:



This graph shows which files directly or indirectly include this file:



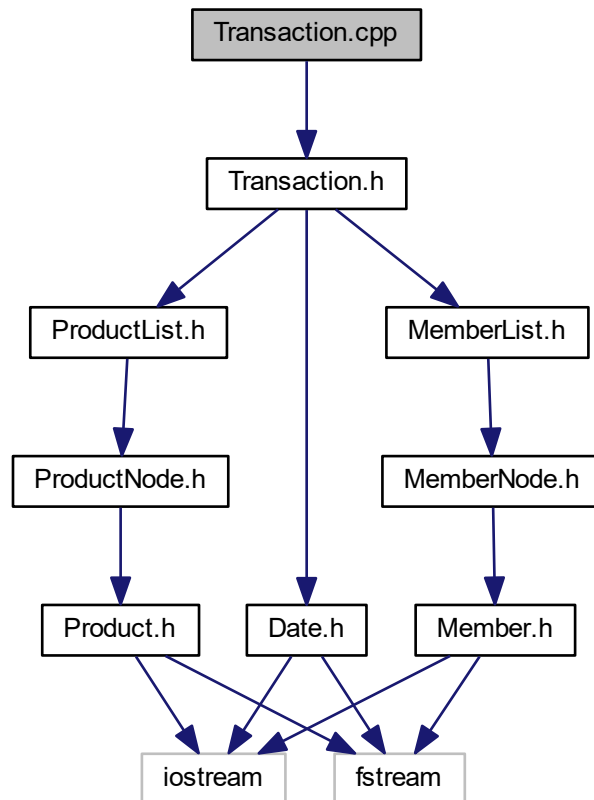
Classes

- class [Statistics](#)

5.19 Transaction.cpp File Reference

```
#include "Transaction.h"
```

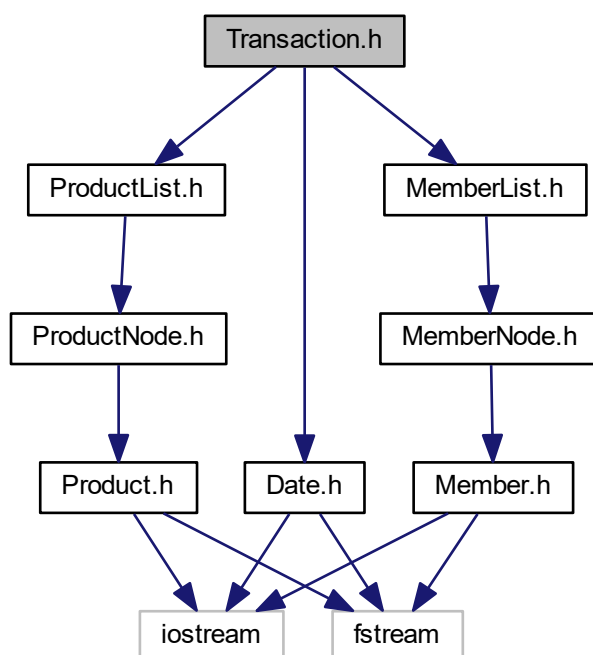
Include dependency graph for Transaction.cpp:



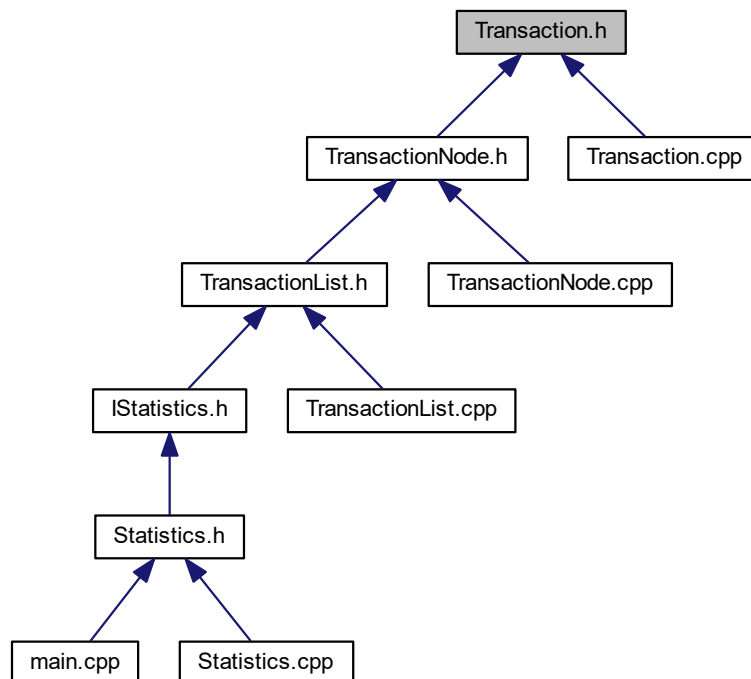
5.20 Transaction.h File Reference

```
#include "ProductList.h"  
#include "Date.h"  
#include "MemberList.h"
```

Include dependency graph for Transaction.h:



This graph shows which files directly or indirectly include this file:



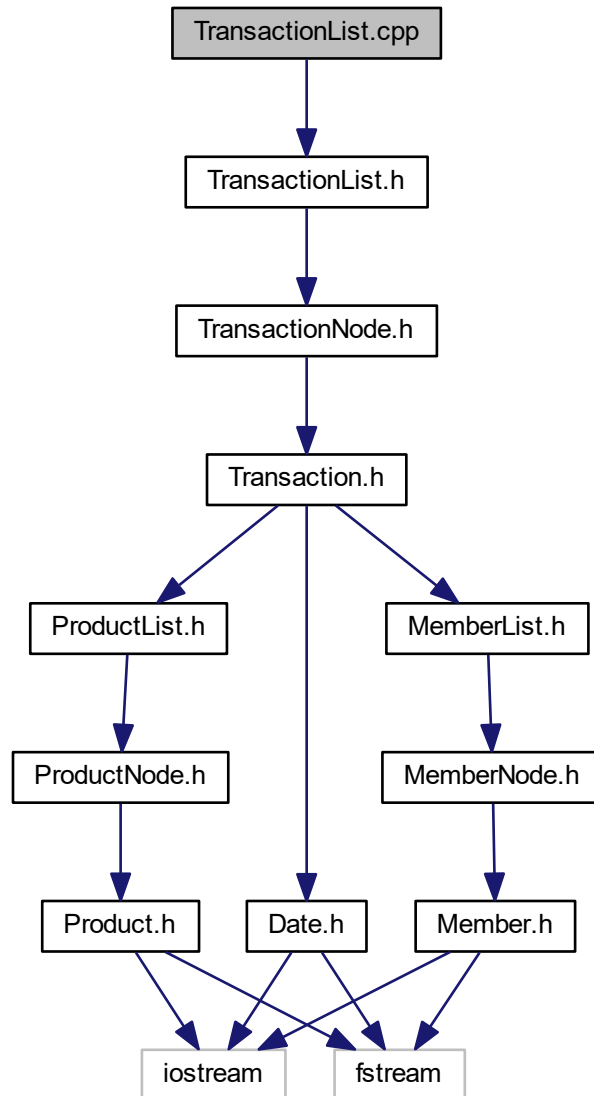
Classes

- class [Transaction](#)

5.21 TransactionList.cpp File Reference

```
#include "TransactionList.h"
```

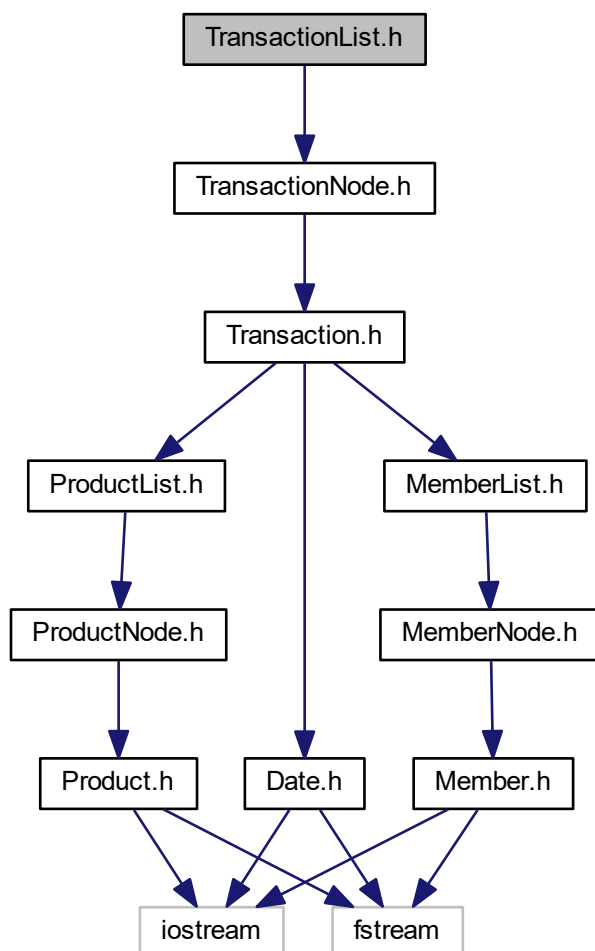
Include dependency graph for TransactionList.cpp:



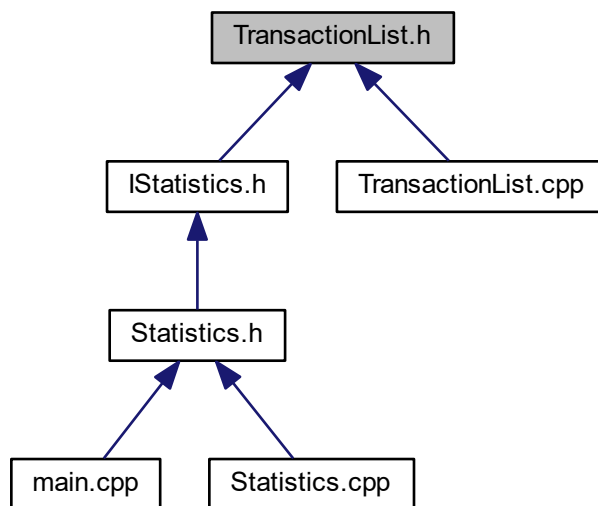
5.22 TransactionList.h File Reference

```
#include "TransactionNode.h"
```


Include dependency graph for TransactionList.h:



This graph shows which files directly or indirectly include this file:



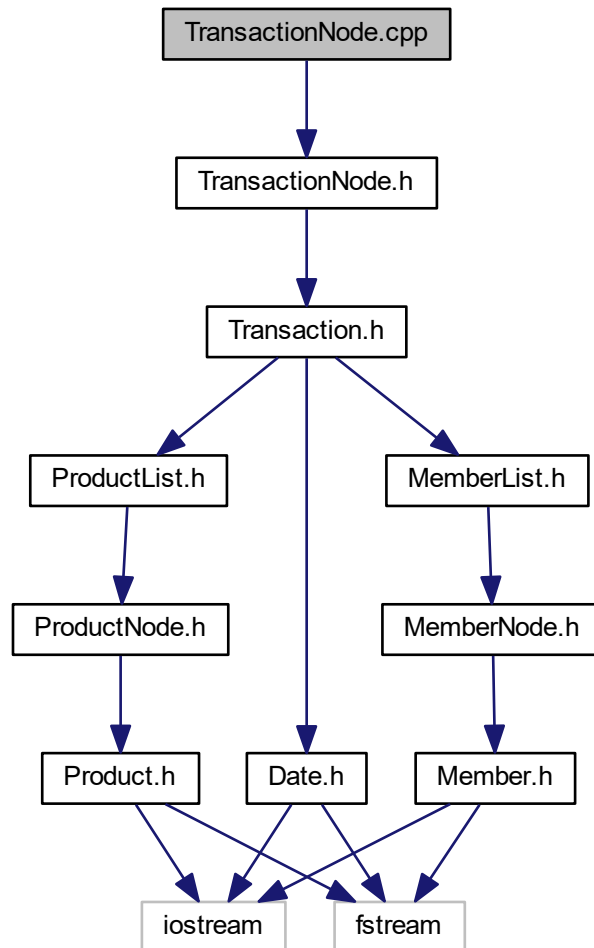
Classes

- class [TransactionList](#)

5.23 TransactionNode.cpp File Reference

```
#include "TransactionNode.h"
```

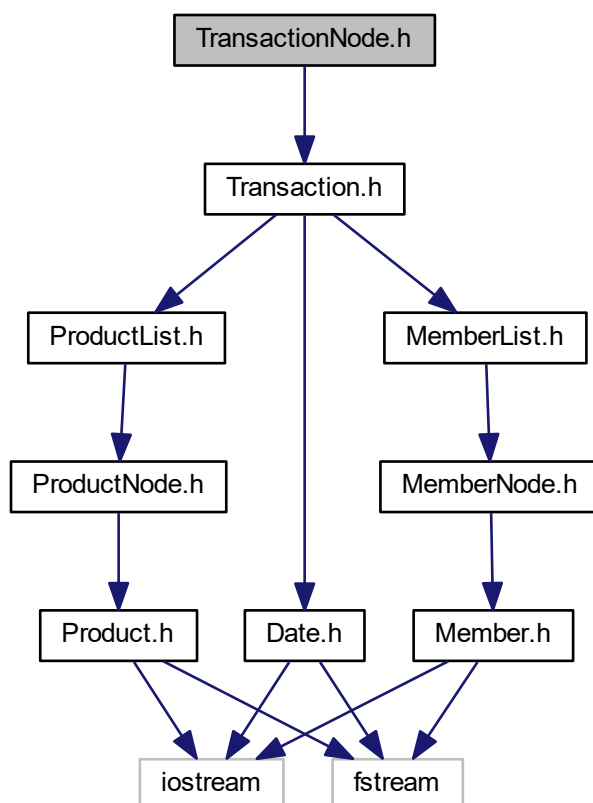
Include dependency graph for TransactionNode.cpp:



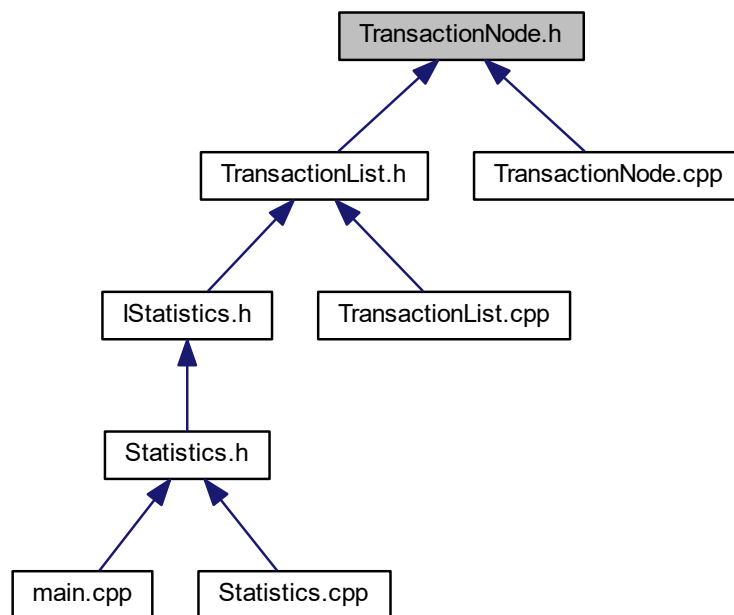
5.24 TransactionNode.h File Reference

```
#include "Transaction.h"
```

Include dependency graph for TransactionNode.h:



This graph shows which files directly or indirectly include this file:



Classes

- class [TransactionNode](#)

Index

- ~MemberList
 - MemberList, [25](#)
- ~ProductList
 - ProductList, [40](#)
- ~TransactionList
 - TransactionList, [73](#)

- addNode
 - MemberList, [26](#)
 - ProductList, [40](#), [41](#)
 - TransactionList, [73](#)
- addNodeFromFile
 - MemberList, [27](#)
 - ProductList, [42](#)
 - TransactionList, [74](#)

- countMean
 - IStatistics, [16](#), [17](#)
 - Statistics, [54–56](#)
- countMedian
 - IStatistics, [17](#), [18](#)
 - Statistics, [56–58](#)
- countStandardDeviation
 - IStatistics, [18](#), [19](#)
 - Statistics, [59–61](#)
- countSum
 - IStatistics, [19](#)
 - Statistics, [61–63](#)

- Date, [7](#)
 - Date, [8](#)
 - getDay, [9](#)
 - getMonth, [9](#)
 - getYear, [10](#)
 - isLeapYear, [10](#)
 - loadDate, [10](#)
 - loadDateFromFile, [11](#)
 - operator<=, [11](#)
 - operator>=, [12](#)
 - print, [12](#)
 - setDay, [13](#)
 - setMonth, [14](#)
 - setYear, [14](#)
- Date.cpp, [81](#)
- Date.h, [81](#)

- getCategory
 - Product, [35](#)
- getDate
 - Transaction, [68](#)

- getDay
 - Date, [9](#)
- getHead
 - MemberList, [27](#)
 - ProductList, [42](#)
 - TransactionList, [74](#)
- getMean
 - IStatistics, [20](#)
 - Statistics, [64](#)
- getMedian
 - IStatistics, [20](#)
 - Statistics, [64](#)
- getMember
 - Transaction, [68](#)
- getMonth
 - Date, [9](#)
- getName
 - Member, [22](#)
 - Product, [35](#)
- getNext
 - MemberNode, [32](#)
 - ProductNode, [50](#)
 - TransactionNode, [79](#)
- getNode
 - MemberNode, [32](#)
 - ProductNode, [51](#)
 - TransactionNode, [79](#)
- getPosition
 - Member, [22](#)
- getPrice
 - Product, [36](#)
 - Transaction, [69](#)
- getPurchaseList
 - Transaction, [69](#)
- getShop
 - Product, [36](#)
 - Transaction, [70](#)
- getStDev
 - IStatistics, [20](#)
 - Statistics, [65](#)
- getSum
 - IStatistics, [20](#)
 - Statistics, [65](#)
- getYear
 - Date, [10](#)

- howManyInList
 - ProductList, [43](#)
- isInList

- MemberList, 28
- ProductList, 43
- isLeapYear
 - Date, 10
- isSame
 - Member, 22
 - Product, 36
- IStatistics, 15
 - countMean, 16, 17
 - countMedian, 17, 18
 - countStandardDeviation, 18, 19
 - countSum, 19
 - getMean, 20
 - getMedian, 20
 - getStDev, 20
 - getSum, 20
 - print, 20
- IStatistics.h, 83
- loadDate
 - Date, 10
- loadDateFromFile
 - Date, 11
- loadFromFile
 - MemberList, 28
 - ProductList, 44
 - TransactionList, 75
- loadMember
 - Member, 23
- loadMemberFromFile
 - Member, 23
- loadProduct
 - Product, 37, 38
- loadProductFromFile
 - Product, 38
- loadTransaction
 - Transaction, 70
- loadTransactionFromFile
 - Transaction, 71
- main
 - main.cpp, 85
- main.cpp, 84
 - main, 85
- Member, 21
 - getName, 22
 - getPosition, 22
 - isSame, 22
 - loadMember, 23
 - loadMemberFromFile, 23
 - Member, 21
 - operator==, 24
- Member.cpp, 86
- Member.h, 87
- MemberList, 24
 - ~MemberList, 25
 - addNode, 26
 - addNodeFromFile, 27
 - getHead, 27
 - isInList, 28
 - loadFromFile, 28
 - MemberList, 25
 - print, 29
 - printToFile, 30
 - setHead, 30
- MemberList.cpp, 88
- MemberList.h, 89
- MemberNode, 31
 - getNext, 32
 - getNode, 32
 - MemberNode, 32
 - setNext, 33
 - setNode, 33
- MemberNode.cpp, 91
- MemberNode.h, 91
- operator<=
 - Date, 11
- operator>=
 - Date, 12
- operator==
 - Member, 24
 - Product, 38
- print
 - Date, 12
 - IStatistics, 20
 - MemberList, 29
 - ProductList, 44
 - Statistics, 65
 - TransactionList, 75
- printlnLine
 - ProductList, 45
- printToFile
 - MemberList, 30
 - ProductList, 46
 - TransactionList, 76
- printWithAmount
 - ProductList, 47
- Product, 33
 - getCategory, 35
 - getName, 35
 - getPrice, 36
 - getShop, 36
 - isSame, 36
 - loadProduct, 37, 38
 - loadProductFromFile, 38
 - operator==, 38
 - Product, 34
- Product.cpp, 93
- Product.h, 93
- ProductList, 39
 - ~ProductList, 40
 - addNode, 40, 41
 - addNodeFromFile, 42
 - getHead, 42
 - howManyInList, 43
 - isInList, 43

- loadFromFile, 44
- print, 44
- printlnLine, 45
- printToFile, 46
- printWithAmount, 47
- ProductList, 40
- setHead, 48
- wasItBefore, 48
- ProductList.cpp, 94
- ProductList.h, 95
- ProductNode, 49
 - getNext, 50
 - getNode, 51
 - ProductNode, 50
 - setNext, 51
 - setNode, 51
- ProductNode.cpp, 97
- ProductNode.h, 97
- setDay
 - Date, 13
- setHead
 - MemberList, 30
 - ProductList, 48
 - TransactionList, 77
- setMonth
 - Date, 14
- setNext
 - MemberNode, 33
 - ProductNode, 51
 - TransactionNode, 80
- setNode
 - MemberNode, 33
 - ProductNode, 51
 - TransactionNode, 80
- setYear
 - Date, 14
- Statistics, 52
 - countMean, 54–56
 - countMedian, 56–58
 - countStandardDeviation, 59–61
 - countSum, 61–63
 - getMean, 64
 - getMedian, 64
 - getStDev, 65
 - getSum, 65
 - print, 65
- Statistics.cpp, 99
- Statistics.h, 99
- Transaction, 66
 - getDate, 68
 - getMember, 68
 - getPrice, 69
 - getPurchaseList, 69
 - getShop, 70
 - loadTransaction, 70
 - loadTransactionFromFile, 71
 - Transaction, 67
- Transaction.cpp, 101
- Transaction.h, 101
- TransactionList, 72
 - ~TransactionList, 73
 - addNode, 73
 - addNodeFromFile, 74
 - getHead, 74
 - loadFromFile, 75
 - print, 75
 - printToFile, 76
 - setHead, 77
 - TransactionList, 73
- TransactionList.cpp, 103
- TransactionList.h, 104
- TransactionNode, 78
 - getNext, 79
 - getNode, 79
 - setNext, 80
 - setNode, 80
 - TransactionNode, 78
- TransactionNode.cpp, 106
- TransactionNode.h, 107
- wasItBefore
 - ProductList, 48