Die Class Reference

Public Member Functions

	Die (int x) Dice Constructor. More...
void	**roll** () Rolls the dice. More...
int	**getSides** () Returns the sides of a dice. More...
int	**getValue** () Get value function. More...

Private Attributes

int	**sides**
int	**value**

Constructor & Destructor Documentation

◆ Die()

Die::Die (int x)

Dice Constructor.

Dice Constructor (sides)

Parameters

x The number of sides on the dice

Member Function Documentation

◆ getSides()

int Die::getSides ( )

Returns the sides of a dice.

The getSides member function returns the number of for this dice

Returns

The total sides

◆ getValue()

int Die::getValue ( )

Get value function.

The getValue member function returns the die's value

Returns

dice value

◆ roll()

void Die::roll ( )

Rolls the dice.

The roll member function simulates the rolling of a dice

Member Data Documentation

◆ sides

int Die::sides private

Number of sides the dice has

◆ value

int Die::value private

Value the dice was rolled

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

- C:/Users/ebald/source/repos/Project3/Project3/**Dice.h**
- C:/Users/ebald/source/repos/Project3/Project3/Dice.cpp

ParkedCar Class Reference

`#include <ParkedCar.h>`

Public Member Functions

	ParkedCar (string ma, string mo, string c, string lic, int m)
	ParkedCar ()
string	**getMake** ()
string	**getModel** ()
string	**getColor** ()
string	**getLicense** ()
int	**getMinutes** ()
void	**setMake** (string ma)
void	**setModel** (string mo)
void	**setColor** (string c)
void	**setLicense** (string li)
void	**setMinutes** (int m)

Private Attributes

string	**make**
string	**model**
string	**color**
string	**license**
int	**minutes**

Detailed Description

Represents a parked car

Constructor & Destructor Documentation

◆ ParkedCar() [1/2]

```
ParkedCar::ParkedCar ( string  ma,
                       string  mo,
                       string  c,
                       string  lic,
                       int     m
                     )
```

inline

A constructor for the car, the car is defined by the parameters below

Parameters

ma	The make of the car
mo	The model of the car
c	The color of the car
lic	The license of the car
m	The minutes the car has been parked

◆ ParkedCar() [2/2]

```
ParkedCar::ParkedCar ( )
```

inline

Default constructor for car

Sets the values to "" and -1;

Member Function Documentation

◆ getColor()

```
string ParkedCar::getColor ( )
```

inline

Gets the color

Returns the color of the car

Returns

color

◆ getLicense()

string ParkedCar::getLicense ( ) inline

Gets the license

Returns the license plate of the car

Returns

license

◆ getMake()

string ParkedCar::getMake ( ) inline

Gets the make

Returns the make of the car

Returns

make

◆ getMinutes()

int ParkedCar::getMinutes ( ) inline

Gets the minutes parked

Returns the minutes the car has been parked

Returns

minutes

◆ getModel()

string ParkedCar::getModel ( ) inline

Gets the model

Returns the model of the car

Returns

model

◆ setColor()

void ParkedCar::setColor (string c) inline

Sets the color

Sets the color of the car to the parameter inputted

Parameters

c The desired color for the car

◆ setLicense()

void ParkedCar::setLicense (string li) inline

Sets the license

Sets the license of the car

Parameters

li The desired license for the car

◆ setMake()

void ParkedCar::setMake (string *ma*) inline

Sets the make

Sets the make to the parameter inputted

Parameters

ma the make to be set to

◆ setMinutes()

void ParkedCar::setMinutes (int *m*) inline

Sets the parked time

Sets the cars parked time at its location

Parameters

m The desired parked time

◆ setModel()

void ParkedCar::setModel (string *mo*) inline

Sets the model

Sets the model of the car to the parameter inputted

Parameters

mo The desired model for the car

Member Data Documentation

◆ color

string ParkedCar::color private

The color of the car

◆ license

string ParkedCar::license private

The licesnse plate of the car

◆ make

string ParkedCar::make private

The make of the car

◆ minutes

int ParkedCar::minutes private

The total parked time of the car

◆ model

string ParkedCar::model private

The model of the car

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

- C:/Users/ebald/source/repos/Project3/Project3/**ParkedCar.h**
- C:/Users/ebald/source/repos/Project3/Project3/ParkedCar.cpp

Generated by doxygen 1.8.15

ParkingMeter Class Reference

```
#include <ParkingMeter.h>
```

Public Member Functions

	ParkingMeter (int pTime)
	ParkingMeter ()
int	**getPurchasedTime** ()
void	**setpurchasedTime** (int pTime)

Private Attributes

int	**purchasedTime**

Detailed Description

The Parking Meter class

Represents a parking meter

Constructor & Destructor Documentation

◆ ParkingMeter() [1/2]

ParkingMeter::ParkingMeter (int *pTime*) inline

Constructor

Creates a parking meter with a specified purchased time

Parameters

pTime The purchased time

◆ ParkingMeter() [2/2]

ParkingMeter::ParkingMeter ( ) inline

Default constructor

Sets the time purchased to zero

Member Function Documentation

◆ getPurchasedTime()

int ParkingMeter::getPurchasedTime ( ) inline

Gets purchassed time

Returns the purchased time for the meter

Returns

purchasedTime

◆ setpurchasedTime()

void ParkingMeter::setpurchasedTime (int pTime) inline

Sets the purchased time

Sets the purchased time of a parking meter to the input

Parameters

pTime The parking time purchased

Member Data Documentation

◆ purchasedTime

int ParkingMeter::purchasedTime

private

Purchased ammount of time on the meter

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

- C:/Users/ebald/source/repos/Project3/Project3/**ParkingMeter.h**
- C:/Users/ebald/source/repos/Project3/Project3/ParkingMeter.cpp

Generated by doxygen 1.8.15

ParkingTicket Class Reference

```
#include <ParkingTicket.h>
```

Public Member Functions

	ParkingTicket (int bn, string n, **ParkedCar** &c, **ParkingMeter** &m)
int	**getFine** ()
int	**getBadgeNumber** ()
string	**getOfficerName** ()
void	**setFine** (**ParkedCar** &**car**, **ParkingMeter** &**meter**)
void	**printTicket** ()

Private Attributes

int	**fine**
int	**badgeNumber**
string	**officerName**
ParkedCar	**car**
ParkingMeter	**meter**

Detailed Description

The Parking Ticket Class

Represents a parking ticket

Constructor & Destructor Documentation

◆ ParkingTicket()

```
ParkingTicket::ParkingTicket ( int             bn,
                               string          n,
                               ParkedCar &     c,
                               ParkingMeter &  m
                             )
```

inline

Costructor for a parking meter

Parameters

bn	The officer's badge number
n	The officer's name
c	The reference of the violating car
m	The reference to the **ParkingMeter**

Member Function Documentation

◆ getBadgeNumber()

```
int ParkingTicket::getBadgeNumber ( )
```

inline

Gets the badge number

Returns the issuing officer's badge number

Returns

badgeNumber

◆ getFine()

```
int ParkingTicket::getFine ( )
```

inline

Gets the fine

Returns the cost of the fine

Returns

fine

◆ getOfficerName()

```
string ParkingTicket::getOfficerName ( )
```

inline

Gets the officer's name

Returns the issuing officer's name

Returns

officerName

◆ printTicket()

```
void ParkingTicket::printTicket ( )
```

inline

Prints out the ticket information

◆ setFine()

```
void ParkingTicket::setFine ( ParkedCar &    car,
                              ParkingMeter & meter
                            )
```

inline

Sets and calculates the fine for a car

Parameters

car	The car to get the fine
meter	The meter the violating car is parked at

Member Data Documentation

◆ badgeNumber

int ParkingTicket::badgeNumber private

The badge number of the issuing officer

◆ car

ParkedCar ParkingTicket::car private

The car that got the ticket

◆ fine

int ParkingTicket::fine private

The cost of the fine

◆ meter

ParkingMeter ParkingTicket::meter private

The meter the car was parked at

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

- C:/Users/ebald/source/repos/Project3/Project3/**ParkingTicket.h**
- C:/Users/ebald/source/repos/Project3/Project3/ParkingTicket.cpp

Generated by doxygen 1.8.15

PoliceOfficer Class Reference

`#include <PoliceOfficer.h>`

Public Member Functions

	PoliceOfficer (string n, int badge)
string	**getName** ()
int	**getBadgeNumber** ()
void	**setName** (string s)
void	**setBadgeNumber** (int x)
int	**checkParkedCar** (**ParkedCar** &car, **ParkingMeter** &meter)
ParkingTicket	**issueTicket** (**ParkedCar** &car, **ParkingMeter** &meter)

Private Attributes

string	**name**
int	**badgeNumber**

Detailed Description

The Police Officer class

Represents a police officer

Constructor & Destructor Documentation

◆ PoliceOfficer()

```
PoliceOfficer::PoliceOfficer ( string  n,
                               int     badge
                             )                                   inline
```

The constructor for a Police Officer

Creates a police officer object

Parameters

n	The name of the officer
badge	The badge number of the officer

Member Function Documentation

◆ checkParkedCar()

```
int PoliceOfficer::checkParkedCar ( ParkedCar &     car,
                                    ParkingMeter &  meter
                                  )                              inline
```

Checks to see for expired time

Parameters

car	The car parked
meter	The meter the car is parked at

Returns

1 for violate and 0 for no violation

◆ getBadgeNumber()

int PoliceOfficer::getBadgeNumber ( ) inline

Gets badge number

Returns the badge number of the officer

Returns

badge number

◆ getName()

string PoliceOfficer::getName ( ) inline

Gets the name

Returns the name of the officer

Returns

name

◆ issueTicket()

```
ParkingTicket PoliceOfficer::issueTicket ( ParkedCar &     car,
                                           ParkingMeter &  meter
                                         )                         inline
```

Issues a parking ticket

Parameters

car	The car to be ticketed
meter	The meter the car is parked at

Returns

A ticket object

◆ setBadgeNumber()

void PoliceOfficer::setBadgeNumber (int x) inline

Sets the badge number

Sets the badge number of the officer

Parameters

- **x** The number for the officer's badge

◆ setName()

void PoliceOfficer::setName (string s) inline

Sets the name

Sets the name of the officer to s

Parameters

- **s** The desired name to be set

Member Data Documentation

◆ badgeNumber

int PoliceOfficer::badgeNumber private

The officer' badge number

◆ name

string PoliceOfficer::name private

The name of the officer

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

- C:/Users/ebald/source/repos/Project3/Project3/**PoliceOfficer.h**

- C:/Users/ebald/source/repos/Project3/Project3/PoliceOfficer.cpp

Generated by doxygen 1.8.15