

## Pizzeria

Generated by Doxygen 1.9.1



<b>1 OOP Projekt - Pizzeria</b>	<b>1</b>
1.1 Zadání	1
1.1.1 Checklist	2
<b>2 Hierarchical Index</b>	<b>3</b>
2.1 Class Hierarchy	3
<b>3 Class Index</b>	<b>5</b>
3.1 Class List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 AbstractEmployee Class Reference	7
4.1.1 Detailed Description	7
4.2 AbstractOrder Class Reference	8
4.2.1 Detailed Description	8
4.3 Cheese Class Reference	8
4.3.1 Detailed Description	9
4.3.2 Constructor & Destructor Documentation	9
4.3.2.1 Cheese()	10
4.3.3 Member Function Documentation	10
4.3.3.1 getType()	10
4.3.3.2 setType()	10
4.4 Chef Class Reference	11
4.4.1 Detailed Description	12
4.4.2 Constructor & Destructor Documentation	12
4.4.2.1 Chef() [1/2]	12
4.4.2.2 Chef() [2/2]	12
4.4.3 Member Function Documentation	13
4.4.3.1 assingOrder()	13
4.4.3.2 getMaxOrderCount()	13
4.4.3.3 getOrder()	13
4.4.3.4 removeOrder()	14
4.4.3.5 setMaxOrderCount()	14
4.5 Custom Class Reference	14
4.5.1 Detailed Description	16
4.5.2 Constructor & Destructor Documentation	16
4.5.2.1 Custom()	16
4.5.3 Member Function Documentation	16
4.5.3.1 addCheese()	16
4.5.3.2 addMeat()	16
4.5.3.3 addVegetables()	17
4.6 Dough Class Reference	17
4.6.1 Detailed Description	18

4.6.2 Constructor & Destructor Documentation	18
4.6.2.1 Dough()	18
4.7 Employee Class Reference	19
4.7.1 Detailed Description	20
4.7.2 Constructor & Destructor Documentation	20
4.7.2.1 Employee()	20
4.7.3 Member Function Documentation	20
4.7.3.1 getAge()	20
4.7.3.2 getName()	21
4.7.3.3 getSurname()	21
4.7.3.4 setAge()	21
4.7.3.5 setName()	21
4.7.3.6 setSurname()	22
4.8 Ingredient Class Reference	22
4.8.1 Detailed Description	23
4.8.2 Constructor & Destructor Documentation	23
4.8.2.1 Ingredient()	23
4.8.3 Member Function Documentation	23
4.8.3.1 getName()	23
4.8.3.2 getQuantity()	23
4.8.3.3 setName()	24
4.8.3.4 setQuantity()	24
4.9 Meat Class Reference	24
4.9.1 Detailed Description	25
4.9.2 Constructor & Destructor Documentation	25
4.9.2.1 Meat()	25
4.10 OnlineOrder Class Reference	26
4.10.1 Detailed Description	27
4.10.2 Constructor & Destructor Documentation	27
4.10.2.1 OnlineOrder()	27
4.10.3 Member Function Documentation	27
4.10.3.1 getDeliveryPrice()	27
4.10.3.2 getPrice()	28
4.11 Order Class Reference	28
4.11.1 Detailed Description	29
4.11.2 Constructor & Destructor Documentation	29
4.11.2.1 Order()	30
4.11.3 Member Function Documentation	30
4.11.3.1 addPizza()	30
4.11.3.2 getAddress()	30
4.11.3.3 getName()	31
4.11.3.4 getPhone()	31

4.11.3.5 getPizzas()	31
4.11.3.6 getPizzasCount()	31
4.11.3.7 getPrice()	32
4.11.3.8 removePizza()	32
4.12 PersonalOrder Class Reference	32
4.12.1 Detailed Description	33
4.12.2 Constructor & Destructor Documentation	34
4.12.2.1 PersonalOrder()	34
4.13 Pizza Class Reference	34
4.13.1 Detailed Description	36
4.13.2 Constructor & Destructor Documentation	36
4.13.2.1 Pizza()	36
4.13.3 Member Function Documentation	36
4.13.3.1 addDough()	36
4.13.3.2 addSauce()	37
4.13.3.3 getClassCount()	37
4.13.3.4 getIngredient()	37
4.13.3.5 getIngredientsCount()	37
4.13.3.6 getName()	38
4.13.3.7 getPrice()	38
4.13.3.8 removeIngredient()	38
4.13.3.9 setPrice()	38
4.14 Pizzeria Class Reference	39
4.14.1 Detailed Description	39
4.14.2 Constructor & Destructor Documentation	40
4.14.2.1 Pizzeria()	40
4.14.3 Member Function Documentation	40
4.14.3.1 addEmployee()	40
4.14.3.2 addOrder()	40
4.14.3.3 getEmployee()	41
4.14.3.4 getName()	41
4.14.3.5 getOrder()	41
4.14.3.6 printEmployees()	42
4.14.3.7 printOrders()	42
4.14.3.8 removeEmployee()	42
4.14.3.9 removeOrder()	42
4.14.3.10 setName()	43
4.15 Sauce Class Reference	43
4.15.1 Detailed Description	44
4.15.2 Constructor & Destructor Documentation	44
4.15.2.1 Sauce()	44
4.16 Service Class Reference	45

4.16.1 Detailed Description . . . . .	46
4.16.2 Constructor & Destructor Documentation . . . . .	46
4.16.2.1 Service() . . . . .	46
4.17 Vegetables Class Reference . . . . .	46
4.17.1 Detailed Description . . . . .	47
4.17.2 Constructor & Destructor Documentation . . . . .	47
4.17.2.1 Vegetables() . . . . .	47
4.18 Vegetarian Class Reference . . . . .	48
4.18.1 Detailed Description . . . . .	49
4.18.2 Member Function Documentation . . . . .	49
4.18.2.1 addVegetables() . . . . .	49

<b>Index</b>	<b>51</b>
--------------	-----------

# Chapter 1

## OOP Projekt - Pizzeria

### 1.1 Zadání

1. Alespoň 7 rozdílných tříd. Třídy mají atributy a/nebo metody, konstruktor a destruktory. **0,5b** Třídy jsou v samostatných .h a .cpp souborech, v hlavním .cpp souboru je pouze funkce main(). **0,5b**
2. V main() funkci je vytvořeno několik instancí tříd a jsou volány metody, které demonstrierují funkci vytvořených struktur a dávají smysl v kontextu projektu. **1b**
3. Alespoň jedna třída má implementováno dynamické pole ukazatelů na objekty jiné třídy. **0,5b** Jsou implementovány metody pro přidání a mazání prvků pole (včetně přeskládání prázdného místa). **0,5b**
4. Alespoň jedna třída má přetížený konstruktor. **1b**
5. Alespoň jedna třída má třídní proměnnou (static), která udržuje počet instancí třídy, včetně implementace do konstruktoru a destruktory. **1b**
6. V projektu existuje dědičná hierarchie obsahující alespoň 3 třídy. **0,5b**
7. Dědičnost bude obsahovat a používat rozšíření (dat i metod), změna chování s využitou pozdní vazbou **0,5b**
8. K projektu je napsána průvodní dokumentace. **0,5b**
9. V projektu je implementován protected atribut. **0,5b**
10. V projektu je implementována alespoň jedna ryze abstraktní třída. **1b**
11. V projektu je využit polymorfismus (např. v dynamickém poli ukazatelů na objekty různých tříd). **1b**
  - Projekt je odevzdán do 8.5. 2023 23:59 na emailovou adresu [martin.radvansky1@vsb.cz](mailto:martin.radvansky1@vsb.cz)
  - Projekt je v jednom souboru .cpp nebo v archivu se soubory .cpp a .h
  - Soubory nebo archiv obsahují v názvu vaše univerzitní ID (aaa0000)
  - Je přiložen soubor s dokumentací k projektu (.docx, .pdf, .txt)
  - Projekt je možné zkompileovat a spustit v MS Visual Studio (pokud programujete- v jiném programu nebo pod Linuxem, ověřte si to! Zvláště si pohlídejte- správnou inicializaci proměnných a polí)
  - Projekt obsahuje pouze aktivní kód a komentáře, usnadňující orientaci či- vysvětlující funkci částí kódu (žádné bloky "vypnutého" kódu)
  - Projekt má textový výstup do konzole, který je srozumitelný i bez náhledu do- kódu
  - Všechny třídy a metody mají název v angličtině
  - Pokud nebude projekt splňovat výše uvedené body, pak ho ani nebudu opravovat

**Minimální počet bodů pro udělení zápočtu a úspěšně obhájený projekt je 5 z 9 bodů.**

### 1.1.1 Checklist

1. 1.1. **Ok** 1.2. **Ok**

1. **Ne**

2. 3.1. **Ok** 3.2. **Ok**

3. **Ok**

4. **Ok**

5. **Ok**

6. **Ok**

7. **Ne**

8. **Ok**

9. **Ok**

10. **Ok**



## Chapter 2

# Hierarchical Index

### 2.1 Class Hierarchy

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

AbstractEmployee . . . . .	7
Employee . . . . .	19
Chef . . . . .	11
Service . . . . .	45
AbstractOrder . . . . .	8
Order . . . . .	28
OnlineOrder . . . . .	26
PersonalOrder . . . . .	32
Ingredient . . . . .	22
Cheese . . . . .	8
Dough . . . . .	17
Meat . . . . .	24
Sauce . . . . .	43
Vegetables . . . . .	46
Pizza . . . . .	34
Custom . . . . .	14
Vegetarian . . . . .	48
Pizzeria . . . . .	39



## Chapter 3

# Class Index

### 3.1 Class List

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

<a href="#">AbstractEmployee</a>		
	The <a href="#">AbstractEmployee</a> class . . . . .	7
<a href="#">AbstractOrder</a>		
	The <a href="#">AbstractOrder</a> class . . . . .	8
<a href="#">Cheese</a>		
	The <a href="#">Cheese</a> class . . . . .	8
<a href="#">Chef</a>		
	The <a href="#">Chef</a> class . . . . .	11
<a href="#">Custom</a>		
	<a href="#">Custom Pizza</a> class . . . . .	14
<a href="#">Dough</a>		
	The <a href="#">Dough</a> class . . . . .	17
<a href="#">Employee</a>		
	The <a href="#">Employee</a> class . . . . .	19
<a href="#">Ingredient</a>		
	The <a href="#">Ingredient</a> class . . . . .	22
<a href="#">Meat</a>		
	The <a href="#">Meat</a> class . . . . .	24
<a href="#">OnlineOrder</a>		
	The <a href="#">OnlineOrder</a> class . . . . .	26
<a href="#">Order</a>		
	<a href="#">Order</a> class . . . . .	28
<a href="#">PersonalOrder</a>		
	The <a href="#">PersonalOrder</a> class . . . . .	32
<a href="#">Pizza</a>		
	The <a href="#">Pizza</a> class . . . . .	34
<a href="#">Pizzeria</a>		
	Class that represents a pizzeria . . . . .	39
<a href="#">Sauce</a>		
	The <a href="#">Sauce</a> class . . . . .	43
<a href="#">Service</a>		
	The <a href="#">Service</a> class . . . . .	45
<a href="#">Vegetables</a>		
	The <a href="#">Vegetables</a> class . . . . .	46
<a href="#">Vegetarian</a>		
	The <a href="#">Vegetarian</a> pizza class . . . . .	48



## Chapter 4

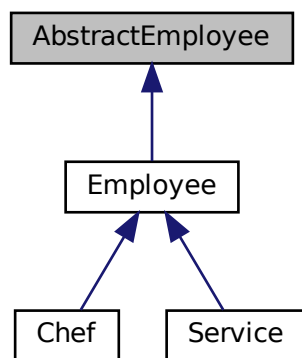
# Class Documentation

### 4.1 AbstractEmployee Class Reference

The [AbstractEmployee](#) class.

```
#include <AbstractEmployee.h>
```

Inheritance diagram for AbstractEmployee:



#### 4.1.1 Detailed Description

The [AbstractEmployee](#) class.

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

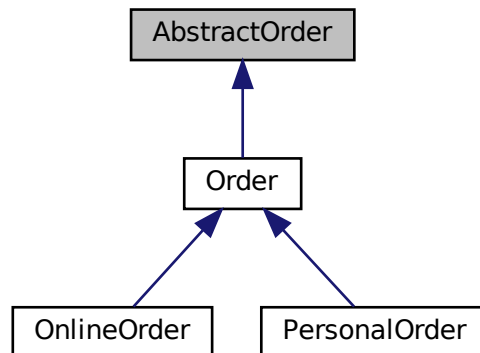
- `AbstractEmployee.h`
- `AbstractEmployee.cpp`

## 4.2 AbstractOrder Class Reference

The [AbstractOrder](#) class.

```
#include <AbstractOrder.h>
```

Inheritance diagram for AbstractOrder:



### Public Member Functions

- virtual int **getPrice** ()=0

### 4.2.1 Detailed Description

The [AbstractOrder](#) class.

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

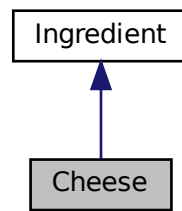
- AbstractOrder.h
- AbstractOrder.cpp

## 4.3 Cheese Class Reference

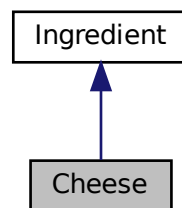
The [Cheese](#) class.

```
#include <Cheese.h>
```

Inheritance diagram for Cheese:



Collaboration diagram for Cheese:



## Public Member Functions

- `Cheese` (int quantity, std::string type)  
*Constructor.*
- `~Cheese` ()  
*Destructor.*
- std::string `getType` ()  
*get type of cheese*
- void `setType` (std::string type)  
*set type of cheese*

### 4.3.1 Detailed Description

The `Cheese` class.

### 4.3.2 Constructor & Destructor Documentation

#### 4.3.2.1 Cheese()

```
Cheese::Cheese (
    int quantity,
    std::string type )
```

Constructor.

##### Parameters

<i>quantity</i>	Quantity of cheese
<i>type</i>	Type of cheese

### 4.3.3 Member Function Documentation

#### 4.3.3.1 getType()

```
std::string Cheese::getType ( )
```

get type of cheese

##### Returns

type of cheese

#### 4.3.3.2 setType()

```
void Cheese::setType (
    std::string type )
```

set type of cheese

##### Parameters

<i>type</i>	type of cheese
-------------	----------------

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

- Cheese.h
- Cheese.cpp

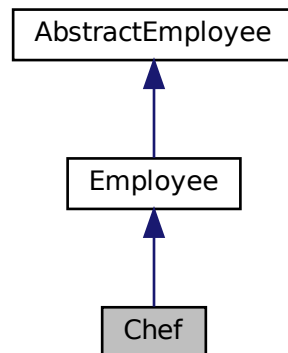


## 4.4 Chef Class Reference

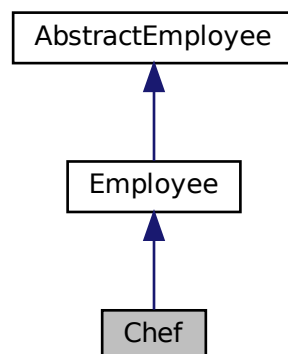
The [Chef](#) class.

```
#include <Chef.h>
```

Inheritance diagram for Chef:



Collaboration diagram for Chef:



### Public Member Functions

- [Chef](#) (std::string name, std::string surname, int age)  
*Constructor.*
- [Chef](#) (std::string name, std::string surname, int age, int maxOrderCount)

- Constructor.*
  - `~Chef ()`
- Destructor.*
  - void `assingOrder (Order *order)`
- Assing order to chef.*
  - void `removeOrder (int index)`
- Remove order from chef.*
  - void `setMaxOrderCount (int maxOrderCount)`
- Set maximum number of orders.*
  - int `getMaxOrderCount ()`
- Get maximum number of orders.*
  - `Order * getOrder (int index)`
- Get specific order.*

#### 4.4.1 Detailed Description

The `Chef` class.

#### 4.4.2 Constructor & Destructor Documentation

##### 4.4.2.1 Chef() [1/2]

```
Chef::Chef (
    std::string name,
    std::string surname,
    int age )
```

Constructor.

Parameters

<i>name</i>	Name of chef
<i>surname</i>	Surname of chef
<i>age</i>	Age of chef

##### 4.4.2.2 Chef() [2/2]

```
Chef::Chef (
    std::string name,
    std::string surname,
    int age,
    int maxOrderCount )
```

Constructor.

## Parameters

<i>name</i>	Name of chef
<i>surname</i>	Surname of chef
<i>age</i>	Age of chef
<i>maxOrderCount</i>	Maximum number of orders

## 4.4.3 Member Function Documentation

### 4.4.3.1 assingOrder()

```
void Chef::assingOrder (
    Order * order )
```

Assing order to chef.

## Parameters

<i>order</i>	Pointer of order to assing
--------------	----------------------------

### 4.4.3.2 getMaxOrderCount()

```
int Chef::getMaxOrderCount ( )
```

Get maximum number of orders.

## Returns

Maximum number of orders

### 4.4.3.3 getOrder()

```
Order * Chef::getOrder (
    int index )
```

Get specific order.

## Parameters

<i>index</i>	Index of order
--------------	----------------

**Returns**

Pointer of order

**4.4.3.4 removeOrder()**

```
void Chef::removeOrder (
    int index )
```

Remove order from chef.

**Parameters**

<i>index</i>	Index of order to remove
--------------	--------------------------

**4.4.3.5 setMaxOrderCount()**

```
void Chef::setMaxOrderCount (
    int maxOrderCount )
```

Set maximum number of orders.

**Parameters**

<i>maxOrderCount</i>	Maximum number of orders
----------------------	--------------------------

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

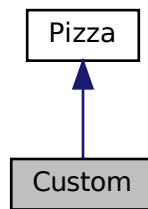
- Chef.h
- Chef.cpp

## 4.5 Custom Class Reference

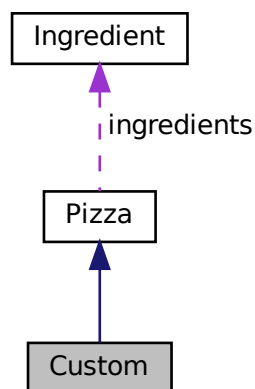
[Custom Pizza](#) class.

```
#include <Custom.h>
```

Inheritance diagram for Custom:



Collaboration diagram for Custom:



## Public Member Functions

- **Custom** (std::string name, int bonusIngredientsCount)  
*Constructor.*
- **~Custom** ()  
*Destructor.*
- void **addCheese** (int quantity, std::string type)  
*Add cheese to pizza.*
- void **addMeat** (int quantity)  
*Add meat to pizza.*
- void **addVegetables** (int quantity)  
*Add vegetables to pizza.*

## Additional Inherited Members

### 4.5.1 Detailed Description

[Custom Pizza](#) class.

### 4.5.2 Constructor & Destructor Documentation

#### 4.5.2.1 Custom()

```
Custom::Custom (
    std::string name,
    int bonusIngredientsCount )
```

Constructor.

##### Parameters

<i>name</i>	Name of pizza
<i>bonusIngredientsCount</i>	Number of bonus ingredients

### 4.5.3 Member Function Documentation

#### 4.5.3.1 addCheese()

```
void Custom::addCheese (
    int quantity,
    std::string type )
```

Add cheese to pizza.

##### Parameters

<i>quantity</i>	Quantity of cheese
<i>type</i>	Type of cheese

#### 4.5.3.2 addMeat()

```
void Custom::addMeat (
    int quantity )
```

Add meat to pizza.

**Parameters**

<i>quantity</i>	Quantity of meat
-----------------	------------------

**4.5.3.3 addVegetables()**

```
void Custom::addVegetables (
    int quantity )
```

Add vegetables to pizza.

**Parameters**

<i>quantity</i>	Quantity of vegetables
-----------------	------------------------

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

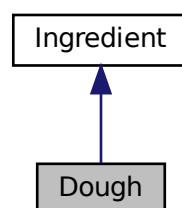
- Custom.h
- Custom.cpp

## 4.6 Dough Class Reference

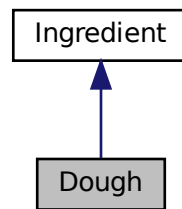
The [Dough](#) class.

```
#include <Dough.h>
```

Inheritance diagram for Dough:



Collaboration diagram for Dough:



## Public Member Functions

- [Dough](#) (int quantity)  
*Constructor.*
- [~Dough](#) ()  
*Destructor.*

### 4.6.1 Detailed Description

The [Dough](#) class.

### 4.6.2 Constructor & Destructor Documentation

#### 4.6.2.1 Dough()

```
Dough::Dough (  
    int quantity )
```

Constructor.

##### Parameters

<i>quantity</i>	Quantity of dough
-----------------	-------------------

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

- Dough.h
- Dough.cpp

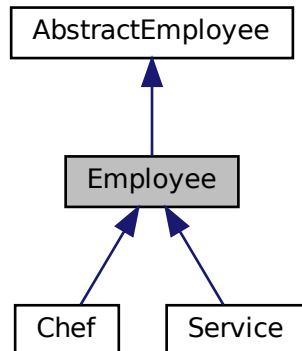


## 4.7 Employee Class Reference

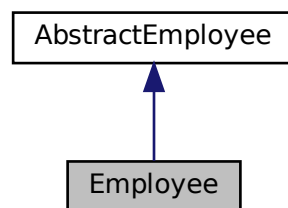
The `Employee` class.

```
#include <Employee.h>
```

Inheritance diagram for `Employee`:



Collaboration diagram for `Employee`:



### Public Member Functions

- `Employee` (`std::string name`, `std::string surname`, `int age`)  
*Constructor.*
- `~Employee` ()  
*Destructor.*
- `std::string getName` ()  
*Get name of employee.*
- `std::string getSurname` ()

- Get surname of employee.*
- int `getAge` ()  
*Get age of employee.*
- void `setName` (std::string name)  
*Set name of employee.*
- void `setSurname` (std::string surname)  
*Set surname of employee.*
- void `setAge` (int age)  
*Set age of employee.*

### 4.7.1 Detailed Description

The `Employee` class.

### 4.7.2 Constructor & Destructor Documentation

#### 4.7.2.1 Employee()

```
Employee::Employee (
    std::string name,
    std::string surname,
    int age )
```

Constructor.

##### Parameters

<i>name</i>	Name of employee
<i>surname</i>	Surname of employee
<i>age</i>	Age of employee

### 4.7.3 Member Function Documentation

#### 4.7.3.1 getAge()

```
int Employee::getAge ( )
```

Get age of employee.

##### Returns

Age of employee

#### 4.7.3.2 getName()

```
std::string Employee::getName ( )
```

Get name of employee.

##### Returns

Name of employee

#### 4.7.3.3 getSurname()

```
std::string Employee::getSurname ( )
```

Get surname of employee.

##### Returns

Surname of employee

#### 4.7.3.4 setAge()

```
void Employee::setAge (
    int age )
```

Set age of employee.

##### Parameters

<i>age</i>	Age of employee
------------	-----------------

#### 4.7.3.5 setName()

```
void Employee::setName (
    std::string name )
```

Set name of employee.

##### Parameters

<i>name</i>	Name of employee
-------------	------------------

#### 4.7.3.6 setSurname()

```
void Employee::setSurname (
    std::string surname )
```

Set surname of employee.

##### Parameters

<i>surname</i>	Surname of employee
----------------	---------------------

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

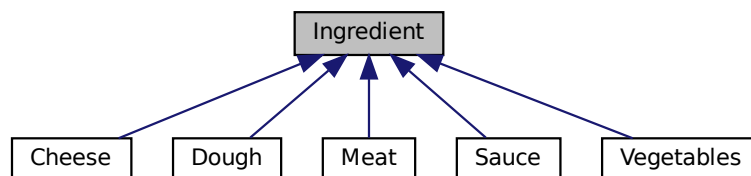
- Employee.h
- Employee.cpp

## 4.8 Ingredient Class Reference

The [Ingredient](#) class.

```
#include <Ingredient.h>
```

Inheritance diagram for Ingredient:



### Public Member Functions

- [Ingredient](#) (std::string name, int quantity)  
*Constructor.*
- virtual [~Ingredient](#) ()  
*Destructor.*
- std::string [getName](#) ()  
*Get name of ingredient.*
- int [getQuantity](#) ()  
*Get quantity of ingredient.*
- void [setName](#) (std::string name)  
*Set name of ingredient.*
- void [setQuantity](#) (int quantity)  
*Set quantity of ingredient.*

## 4.8.1 Detailed Description

The [Ingredient](#) class.

## 4.8.2 Constructor & Destructor Documentation

### 4.8.2.1 Ingredient()

```
Ingredient::Ingredient (
    std::string name,
    int quantity )
```

Constructor.

#### Parameters

<i>name</i>	Name of ingredient
<i>quantity</i>	Quantity of ingredient

## 4.8.3 Member Function Documentation

### 4.8.3.1 getName()

```
std::string Ingredient::getName ( )
```

Get name of ingredient.

#### Returns

Name of ingredient

### 4.8.3.2 getQuantity()

```
int Ingredient::getQuantity ( )
```

Get quantity of ingredient.

#### Returns

Quantity of ingredient

#### 4.8.3.3 setName()

```
void Ingredient::setName (
    std::string name )
```

Set name of ingredient.

##### Parameters

<i>name</i>	Name of ingredient
-------------	--------------------

#### 4.8.3.4 setQuantity()

```
void Ingredient::setQuantity (
    int quantity )
```

Set quantity of ingredient.

##### Parameters

<i>quantity</i>	Quantity of ingredient
-----------------	------------------------

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

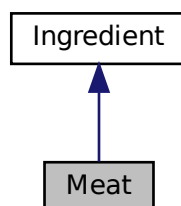
- Ingredient.h
- Ingredient.cpp

## 4.9 Meat Class Reference

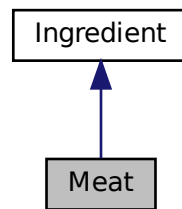
The [Meat](#) class.

```
#include <Meat.h>
```

Inheritance diagram for Meat:



Collaboration diagram for Meat:



## Public Member Functions

- `Meat` (int quantity)  
*Constructor.*
- `~Meat` ()  
*Destructor.*

### 4.9.1 Detailed Description

The `Meat` class.

### 4.9.2 Constructor & Destructor Documentation

#### 4.9.2.1 Meat()

```
Meat::Meat (
    int quantity )
```

Constructor.

Parameters

<i>quantity</i>	Quantity of meat
-----------------	------------------

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

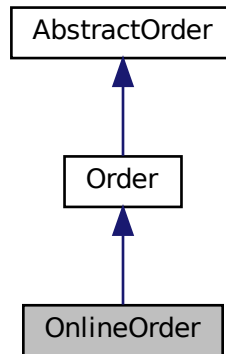
- Meat.h
- Meat.cpp

## 4.10 `OnlineOrder` Class Reference

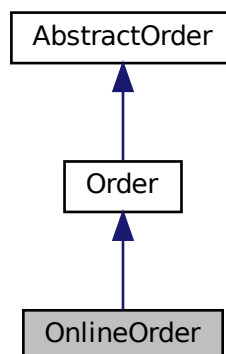
The `OnlineOrder` class.

```
#include <OnlineOrder.h>
```

Inheritance diagram for `OnlineOrder`:



Collaboration diagram for `OnlineOrder`:



### Public Member Functions

- `OnlineOrder` (std::string name, std::string address, std::string phone, int maxPizzasCount, int deliveryPrice)  
*Constructor.*
- `~OnlineOrder` ()



*Destructor.*

- virtual int [getPrice](#) ()

*Get delivery price.*

- int [getDeliveryPrice](#) ()

*Set delivery price.*

### 4.10.1 Detailed Description

The [OnlineOrder](#) class.

### 4.10.2 Constructor & Destructor Documentation

#### 4.10.2.1 OnlineOrder()

```
OnlineOrder::OnlineOrder (  
    std::string name,  
    std::string address,  
    std::string phone,  
    int maxPizzasCount,  
    int deliveryPrice )
```

Constructor.

Parameters

<i>name</i>	Name of customer
<i>address</i>	Address of customer
<i>phone</i>	Phone of customer
<i>maxPizzasCount</i>	Maximum number of pizzas
<i>deliveryPrice</i>	Delivery price

### 4.10.3 Member Function Documentation

#### 4.10.3.1 getDeliveryPrice()

```
int OnlineOrder::getDeliveryPrice ( )
```

Set delivery price.

Parameters

<i>deliveryPrice</i>	Delivery price
----------------------	----------------

#### 4.10.3.2 getPrice()

```
int OnlineOrder::getPrice ( ) [virtual]
```

Get delivery price.

##### Returns

Delivery price

Reimplemented from [Order](#).

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

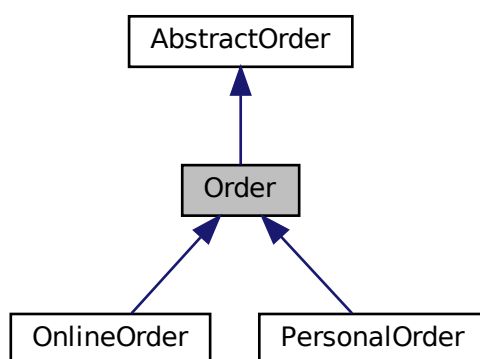
- [OnlineOrder.h](#)
- [OnlineOrder.cpp](#)

## 4.11 Order Class Reference

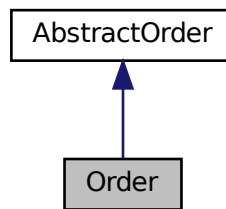
[Order](#) class.

```
#include <Order.h>
```

Inheritance diagram for Order:



Collaboration diagram for Order:



## Public Member Functions

- [Order](#) (std::string name, std::string address, std::string phone, int maxPizzasCount)  
*Constructor.*
- [~Order](#) ()  
*Destructor.*
- virtual int [getPrice](#) ()  
*Get price.*
- void [addPizza](#) ([Pizza](#) \*pizza)  
*Add pizza.*
- void [removePizza](#) (int index)  
*Remove pizza.*
- [Pizza](#) \*\* [getPizzas](#) ()  
*Get pizzas.*
- std::string [getName](#) ()  
*Get name.*
- std::string [getAddress](#) ()  
*Get address.*
- std::string [getPhone](#) ()  
*Get phone.*
- int [getPizzasCount](#) ()  
*Get pizzas count.*

### 4.11.1 Detailed Description

[Order](#) class.

### 4.11.2 Constructor & Destructor Documentation

#### 4.11.2.1 Order()

```
Order::Order (
    std::string name,
    std::string address,
    std::string phone,
    int maxPizzasCount )
```

Constructor.

##### Parameters

<i>name</i>	Name of customer
<i>address</i>	Address of customer
<i>phone</i>	Phone of customer
<i>maxPizzasCount</i>	Maximum number of pizzas

### 4.11.3 Member Function Documentation

#### 4.11.3.1 addPizza()

```
void Order::addPizza (
    Pizza * pizza )
```

Add pizza.

##### Parameters

<i>pizza</i>	Pointer of pizza to add
--------------	-------------------------

#### 4.11.3.2 getAddress()

```
std::string Order::getAddress ( )
```

Get address.

##### Returns

Address

#### 4.11.3.3 getName()

```
std::string Order::getName ( )
```

Get name.

**Returns**

Name

#### 4.11.3.4 getPhone()

```
std::string Order::getPhone ( )
```

Get phone.

**Returns**

Phone

#### 4.11.3.5 getPizzas()

```
Pizza ** Order::getPizzas ( )
```

Get pizzas.

**Returns**

Array of pizzas

#### 4.11.3.6 getPizzasCount()

```
int Order::getPizzasCount ( )
```

Get pizzas count.

**Returns**

Pizzas count

#### 4.11.3.7 getPrice()

```
int Order::getPrice ( ) [virtual]
```

Get price.

##### Returns

Price

Implements [AbstractOrder](#).

Reimplemented in [OnlineOrder](#).

#### 4.11.3.8 removePizza()

```
void Order::removePizza (
    int index )
```

Remove pizza.

##### Parameters

<i>index</i>	Index of pizza to remove
--------------	--------------------------

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

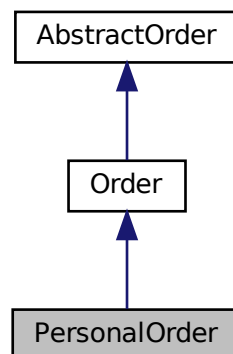
- Order.h
- Order.cpp

## 4.12 PersonalOrder Class Reference

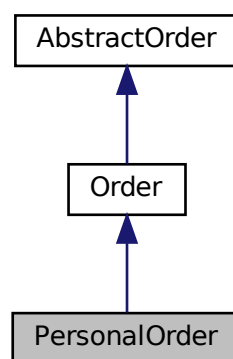
The [PersonalOrder](#) class.

```
#include <PersonalOrder.h>
```

Inheritance diagram for PersonalOrder:



Collaboration diagram for PersonalOrder:



## Public Member Functions

- [PersonalOrder](#) (std::string name, std::string address, std::string phone, int maxPizzasCount)  
*Constructor.*
- [~PersonalOrder](#) ()  
*Destructor.*

### 4.12.1 Detailed Description

The [PersonalOrder](#) class.

## 4.12.2 Constructor & Destructor Documentation

### 4.12.2.1 PersonalOrder()

```
PersonalOrder::PersonalOrder (
    std::string name,
    std::string address,
    std::string phone,
    int maxPizzasCount )
```

Constructor.

#### Parameters

<i>name</i>	Name of the customer
<i>address</i>	Address of the customer
<i>phone</i>	Phone of the customer
<i>maxPizzasCount</i>	Maximum number of pizzas

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

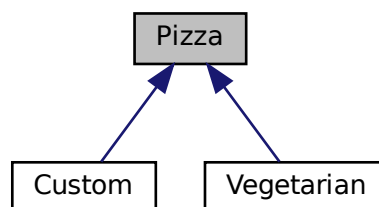
- PersonalOrder.h
- PersonalOrder.cpp

## 4.13 Pizza Class Reference

The [Pizza](#) class.

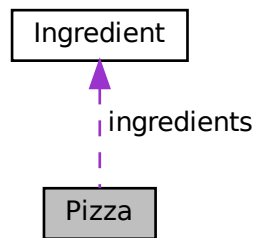
```
#include <Pizza.h>
```

Inheritance diagram for Pizza:





Collaboration diagram for Pizza:



## Public Member Functions

- [Pizza](#) (std::string name, int maxIngredientsCount)  
*Constructor.*
- virtual [~Pizza](#) ()  
*Destructor.*
- std::string [getName](#) ()  
*Get name of pizza.*
- int [getIngredientsCount](#) ()  
*Get count of ingredients.*
- [Ingredient](#) \* [getIngredient](#) (int index)  
*Get pointer to ingredient.*
- void [addDough](#) (int quantity)  
*Add dough.*
- void [addSauce](#) (int quantity)  
*Add sauce.*
- void [removeIngredient](#) (int index)  
*Remove ingredient on index.*
- virtual void [addDefaultIngredients](#) ()  
*Add default ingredients.*
- void [setPrice](#) (int price)  
*Set price.*
- int [getPrice](#) ()  
*Get price.*

## Static Public Member Functions

- static int [getClassCount](#) ()  
*Get number of pizza classes.*

## Protected Attributes

- `Ingredient ** ingredients`  
*Array of ingredients.*
- `int ingredientsCount`  
*Count of ingredients.*

### 4.13.1 Detailed Description

The `Pizza` class.

### 4.13.2 Constructor & Destructor Documentation

#### 4.13.2.1 `Pizza()`

```
Pizza::Pizza (
    std::string name,
    int maxIngredientsCount )
```

Constructor.

##### Parameters

<i>name</i>	Name of pizza
<i>maxIngredientsCount</i>	Maximum number of ingredients

### 4.13.3 Member Function Documentation

#### 4.13.3.1 `addDough()`

```
void Pizza::addDough (
    int quantity )
```

Add dough.

##### Parameters

<i>quantity</i>	Quantity of dough
-----------------	-------------------

#### 4.13.3.2 addSauce()

```
void Pizza::addSauce (
    int quantity )
```

Add sauce.

##### Parameters

<i>quantity</i>	Quantity of sauce
-----------------	-------------------

#### 4.13.3.3 getClassCount()

```
int Pizza::getClassCount ( ) [static]
```

Get number of pizza classes.

##### Returns

Number of pizza classes

#### 4.13.3.4 getIngredient()

```
Ingredient * Pizza::getIngredient (
    int index )
```

Get pointer to ingredient.

##### Parameters

<i>index</i>	Index of ingredient
--------------	---------------------

##### Returns

Pointer to ingredient

#### 4.13.3.5 getIngredientsCount()

```
int Pizza::getIngredientsCount ( )
```

Get count of ingredients.

**Returns**

Count of ingredients

**4.13.3.6 getName()**

```
std::string Pizza::getName ( )
```

Get name of pizza.

**Returns**

Name of pizza

**4.13.3.7 getPrice()**

```
int Pizza::getPrice ( )
```

Get price.

**Returns**

Price

**4.13.3.8 removeIngredient()**

```
void Pizza::removeIngredient (
    int index )
```

Remove ingredient on index.

**Parameters**

<i>index</i>	Index of ingredient
--------------	---------------------

**4.13.3.9 setPrice()**

```
void Pizza::setPrice (
    int price )
```

Set price.

## Parameters

<i>price</i>	Price
--------------	-------

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

- Pizza.h
- Pizza.cpp

## 4.14 Pizzeria Class Reference

Class that represents a pizzeria.

```
#include <Pizzeria.h>
```

### Public Member Functions

- [Pizzeria](#) (std::string name, int maxEmployeesCount, int maxOrderCount)  
*Constructor.*
- [~Pizzeria](#) ()  
*Destructor.*
- std::string [getName](#) ()  
*Get name of pizzeria.*
- void [setName](#) (std::string name)  
*Set name of pizzeria.*
- void [addEmployee](#) ([Employee](#) \*employee)  
*Add employee to pizzeria.*
- void [addOrder](#) ([Order](#) \*order)  
*Add order to pizzeria.*
- [Employee](#) \* [getEmployee](#) (int index)  
*Get employee from pizzeria.*
- [Order](#) \* [getOrder](#) (int index)  
*Get order from pizzeria.*
- void [removeEmployee](#) (int index)  
*Remove employee from pizzeria.*
- void [removeOrder](#) (int index)  
*Remove order from pizzeria.*
- std::string [printEmployees](#) ()  
*Get employees as string.*
- std::string [printOrders](#) ()  
*Get orders as string.*

#### 4.14.1 Detailed Description

Class that represents a pizzeria.

## 4.14.2 Constructor & Destructor Documentation

### 4.14.2.1 Pizzeria()

```
Pizzeria::Pizzeria (
    std::string name,
    int maxEmployeesCount,
    int maxOrderCount )
```

Constructor.

#### Parameters

<i>name</i>	Name of pizzeria
<i>maxEmployeesCount</i>	Maximum number of employees
<i>maxOrderCount</i>	Maximum number of orders

## 4.14.3 Member Function Documentation

### 4.14.3.1 addEmployee()

```
void Pizzeria::addEmployee (
    Employee * employee )
```

Add employee to pizzeria.

#### Parameters

<i>employee</i>	Pointer of employee to add
-----------------	----------------------------

### 4.14.3.2 addOrder()

```
void Pizzeria::addOrder (
    Order * order )
```

Add order to pizzeria.

#### Parameters

<i>order</i>	Pointer of order to add
--------------	-------------------------

#### 4.14.3.3 `getEmployee()`

```
Employee * Pizzeria::getEmployee (
    int index )
```

Get employee from pizzeria.

##### Parameters

<i>index</i>	Index of employee to get
--------------	--------------------------

##### Returns

Pointer of employee

#### 4.14.3.4 `getName()`

```
std::string Pizzeria::getName ( )
```

Get name of pizzeria.

##### Returns

Name of pizzeria

#### 4.14.3.5 `getOrder()`

```
Order * Pizzeria::getOrder (
    int index )
```

Get order from pizzeria.

##### Parameters

<i>index</i>	Index of order to get
--------------	-----------------------

##### Returns

Pointer of order

#### 4.14.3.6 printEmployees()

```
std::string Pizzeria::printEmployees ( )
```

Get employees as string.

##### Returns

Employees as string

#### 4.14.3.7 printOrders()

```
std::string Pizzeria::printOrders ( )
```

Get orders as string.

##### Returns

Orders as string

#### 4.14.3.8 removeEmployee()

```
void Pizzeria::removeEmployee (
    int index )
```

Remove employee from pizzeria.

##### Parameters

<i>index</i>	Index of employee to remove
--------------	-----------------------------

#### 4.14.3.9 removeOrder()

```
void Pizzeria::removeOrder (
    int index )
```

Remove order from pizzeria.

##### Parameters

<i>index</i>	Index of order to remove
--------------	--------------------------



#### 4.14.3.10 setName()

```
void Pizzeria::setName (
    std::string name )
```

Set name of pizzeria.

##### Parameters

<i>name</i>	Name of pizzeria
-------------	------------------

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

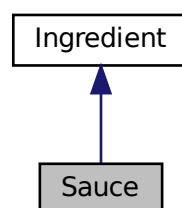
- Pizzeria.h
- Pizzeria.cpp

## 4.15 Sauce Class Reference

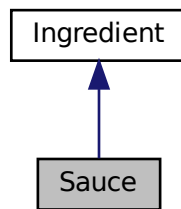
The [Sauce](#) class.

```
#include <Sauce.h>
```

Inheritance diagram for Sauce:



Collaboration diagram for Sauce:



## Public Member Functions

- `Sauce` (int quantity)  
*Constructor.*
- `~Sauce` ()  
*Destructor.*

### 4.15.1 Detailed Description

The `Sauce` class.

### 4.15.2 Constructor & Destructor Documentation

#### 4.15.2.1 Sauce()

```
Sauce::Sauce (  
    int quantity )
```

Constructor.

#### Parameters

<i>quantity</i>	Quantity of sauce
-----------------	-------------------

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

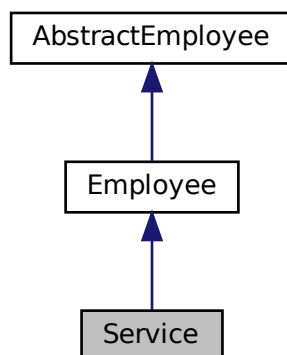
- `Sauce.h`
- `Sauce.cpp`

## 4.16 Service Class Reference

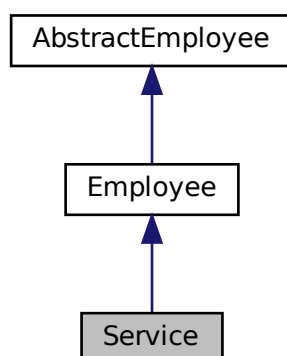
The [Service](#) class.

```
#include <Service.h>
```

Inheritance diagram for Service:



Collaboration diagram for Service:



### Public Member Functions

- [Service](#) (std::string name, std::string surname, int age)  
*Constructor.*
- [~Service](#) ()  
*Destructor.*

### 4.16.1 Detailed Description

The [Service](#) class.

### 4.16.2 Constructor & Destructor Documentation

#### 4.16.2.1 Service()

```
Service::Service (
    std::string name,
    std::string surname,
    int age )
```

Constructor.

##### Parameters

<i>name</i>	Name of the service
<i>surname</i>	Surname of the service
<i>age</i>	Age of the service

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

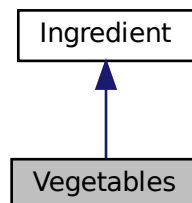
- Service.h
- Service.cpp

## 4.17 Vegetables Class Reference

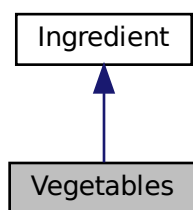
The [Vegetables](#) class.

```
#include <Vegetables.h>
```

Inheritance diagram for Vegetables:



Collaboration diagram for Vegetables:



## Public Member Functions

- `Vegetables` (int quantity)  
*Constructor.*
- `~Vegetables` ()  
*Destructor.*

### 4.17.1 Detailed Description

The `Vegetables` class.

### 4.17.2 Constructor & Destructor Documentation

#### 4.17.2.1 Vegetables()

```
Vegetables::Vegetables (  
    int quantity )
```

Constructor.

Parameters

<i>quantity</i>	Quantity of vegetables
-----------------	------------------------

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

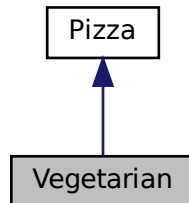
- Vegetables.h
- Vegetables.cpp

## 4.18 Vegetarian Class Reference

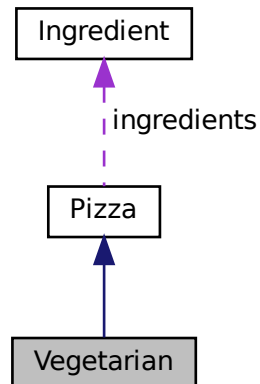
The [Vegetarian](#) pizza class.

```
#include <Vegetarian.h>
```

Inheritance diagram for Vegetarian:



Collaboration diagram for Vegetarian:



### Public Member Functions

- [Vegetarian](#) ()  
*Constructor.*
- [~Vegetarian](#) ()  
*Destructor.*
- virtual void [addDefaultIngredients](#) ()  
*Add default ingredients.*
- void [addVegetables](#) (int quantity)  
*Add cheese to pizza.*

## Additional Inherited Members

### 4.18.1 Detailed Description

The [Vegetarian](#) pizza class.

### 4.18.2 Member Function Documentation

#### 4.18.2.1 addVegetables()

```
void Vegetarian::addVegetables (
    int quantity )
```

Add cheese to pizza.

#### Parameters

<i>quantity</i>	Quantity of cheese
-----------------	--------------------

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

- Vegetarian.h
- Vegetarian.cpp





# Index

- AbstractEmployee, [7](#)
- AbstractOrder, [8](#)
- addCheese
  - Custom, [16](#)
- addDough
  - Pizza, [36](#)
- addEmployee
  - Pizzeria, [40](#)
- addMeat
  - Custom, [16](#)
- addOrder
  - Pizzeria, [40](#)
- addPizza
  - Order, [30](#)
- addSauce
  - Pizza, [36](#)
- addVegetables
  - Custom, [17](#)
  - Vegetarian, [49](#)
- assingOrder
  - Chef, [13](#)
- Cheese, [8](#)
  - Cheese, [9](#)
  - getType, [10](#)
  - setType, [10](#)
- Chef, [11](#)
  - assingOrder, [13](#)
  - Chef, [12](#)
  - getMaxOrderCount, [13](#)
  - getOrder, [13](#)
  - removeOrder, [14](#)
  - setMaxOrderCount, [14](#)
- Custom, [14](#)
  - addCheese, [16](#)
  - addMeat, [16](#)
  - addVegetables, [17](#)
  - Custom, [16](#)
- Dough, [17](#)
  - Dough, [18](#)
- Employee, [19](#)
  - Employee, [20](#)
  - getAge, [20](#)
  - getName, [20](#)
  - getSurname, [21](#)
  - setAge, [21](#)
  - setName, [21](#)
  - setSurname, [22](#)
- getAddress
  - Order, [30](#)
- getAge
  - Employee, [20](#)
- getClassCount
  - Pizza, [37](#)
- getDeliveryPrice
  - OnlineOrder, [27](#)
- getEmployee
  - Pizzeria, [41](#)
- getIngredient
  - Pizza, [37](#)
- getIngredientsCount
  - Pizza, [37](#)
- getMaxOrderCount
  - Chef, [13](#)
- getName
  - Employee, [20](#)
  - Ingredient, [23](#)
  - Order, [30](#)
  - Pizza, [38](#)
  - Pizzeria, [41](#)
- getOrder
  - Chef, [13](#)
  - Pizzeria, [41](#)
- getPhone
  - Order, [31](#)
- getPizzas
  - Order, [31](#)
- getPizzasCount
  - Order, [31](#)
- getPrice
  - OnlineOrder, [28](#)
  - Order, [31](#)
  - Pizza, [38](#)
- getQuantity
  - Ingredient, [23](#)
- getSurname
  - Employee, [21](#)
- getType
  - Cheese, [10](#)
- Ingredient, [22](#)
  - getName, [23](#)
  - getQuantity, [23](#)
  - Ingredient, [23](#)
  - setName, [23](#)
  - setQuantity, [24](#)
- Meat, [24](#)

- Meat, [25](#)
- OnlineOrder, [26](#)
  - getDeliveryPrice, [27](#)
  - getPrice, [28](#)
  - OnlineOrder, [27](#)
- Order, [28](#)
  - addPizza, [30](#)
  - getAddress, [30](#)
  - getName, [30](#)
  - getPhone, [31](#)
  - getPizzas, [31](#)
  - getPizzasCount, [31](#)
  - getPrice, [31](#)
  - Order, [29](#)
  - removePizza, [32](#)
- PersonalOrder, [32](#)
  - PersonalOrder, [34](#)
- Pizza, [34](#)
  - addDough, [36](#)
  - addSauce, [36](#)
  - getClassCount, [37](#)
  - getIngredient, [37](#)
  - getIngredientsCount, [37](#)
  - getName, [38](#)
  - getPrice, [38](#)
  - Pizza, [36](#)
  - removeIngredient, [38](#)
  - setPrice, [38](#)
- Pizzeria, [39](#)
  - addEmployee, [40](#)
  - addOrder, [40](#)
  - getEmployee, [41](#)
  - getName, [41](#)
  - getOrder, [41](#)
  - Pizzeria, [40](#)
  - printEmployees, [41](#)
  - printOrders, [42](#)
  - removeEmployee, [42](#)
  - removeOrder, [42](#)
  - setName, [43](#)
- printEmployees
  - Pizzeria, [41](#)
- printOrders
  - Pizzeria, [42](#)
- removeEmployee
  - Pizzeria, [42](#)
- removeIngredient
  - Pizza, [38](#)
- removeOrder
  - Chef, [14](#)
  - Pizzeria, [42](#)
- removePizza
  - Order, [32](#)
- Sauce, [43](#)
  - Sauce, [44](#)
- Service, [45](#)
  - Service, [46](#)
- setAge
  - Employee, [21](#)
- setMaxOrderCount
  - Chef, [14](#)
- setName
  - Employee, [21](#)
  - Ingredient, [23](#)
  - Pizzeria, [43](#)
- setPrice
  - Pizza, [38](#)
- setQuantity
  - Ingredient, [24](#)
- setSurname
  - Employee, [22](#)
- setType
  - Cheese, [10](#)
- Vegetables, [46](#)
  - Vegetables, [47](#)
- Vegetarian, [48](#)
  - addVegetables, [49](#)