

# Programátorská příručka

k programu

# OpenControl

Autor: Kamil Ježek  
Email: [opencontrol.info@gmail.com](mailto:opencontrol.info@gmail.com)  
Verze dokumentu 1.0

Vznik dokumentu: 2006  
Poslední modifikace: 2009

## Poznámka k vydání

Tento dokument vznikl pro první verzi programu Control (nyní pojmenovaný OpenControl) a proto se ovládání může v budoucích verzích mírně lišit. Předpokládají se však změny rozšiřující funkčnost, které nemusí být v tomto dokumentu popsány, avšak ovládání původní funkčnosti by mělo zůstat zachováno.

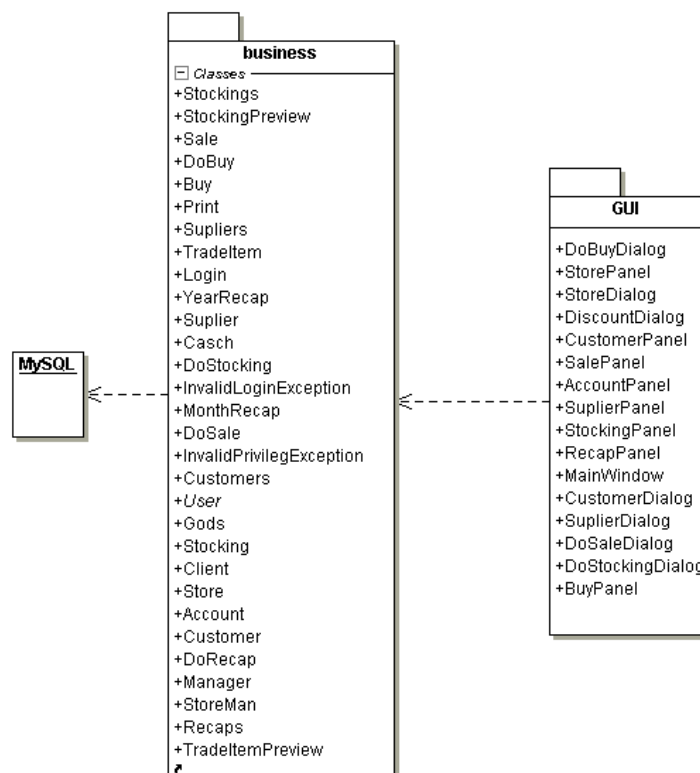
## Poznámka k licenci

Tento dokument (stejně tak jako program OpenControl samotný) je možné volně šířit a modifikovat.

Cílem této dokumentace je být co nejvíce užitečná a je poskytována „tak jak je“. Není však poskytována žádná záruka především, avšak nejenom, na správnost a úplnost údajů zde uvedených.

## 1. Základní princip

Program bude tvořen třetí vrstvou architektury. Datovou vrstvu zajistí databáze MySQL. Aplikační a prezentační vrstva bude naprogramována v jazyce Java. Přičemž prezentační vrstvu tvoří grafické uživatelské rozhraní vytvořené v JFC Swing.



Created by Borland® Together® Designer Community Edition

## 2. Datová vrstva

Následuje přehled tabulek v databázi.

*Vysvětlivky: první řádek tabulky označuje název sloupce, druhý datový typ, třetí význam sloupce. Nad tabulkou je vždy uvede její název. Šípkou je naznačena relace s jinou tabulkou.*

### 2.1 Skladová karta

Uchovává informace o zboží na skladě

Tabulka: gods

gods_id	name	type	dph	unit	EAN	NC	PC_A	PC_B	PC_C	PC_D	quantity
varchar(20) unique	varchar(256)	integer		char(5)	char(13)	decimal(12, 2)					
jednoznačné skladové číslo	jméno zboží pro snadnější orientaci	typ zboží	daň s přidané hodnoty [%]	množstevní jednotka [ks, m, kg, ...]	čárový kód	nákupní cena bez daně[Kč]	prodejní ceny bez daně [Kč]				množství zboží na skladě

### 2.2. Karta dodavatele

Uchovává informace o dodavatelích zboží

Tabulka: supplier

sup_id	name	person	street	city	PSC	tel	fax
integer unique	varchar(256)	varchar(50)	varchar(30)	char(5)	varchar(20)		
jednoznačné číslo dodavatele	název dodavatele	kontaktní osoba	adresa			telefon	fax
			ulice	město	směrovací číslo		

mail	web	ICO	DIC	is DPH	account
varchar(50)	varchar(100)	varchar(20) unique	varchar(20)	boolean	varchar(20)
e-mail	internetová adresa	IČO	DIČ	údaj zda je plátce DPH	bankovní účet

Zadavatel: AKJ servis spol. s r. o.

Zpracoval: Kamil Ježek

POZN.: Jedná se o jednu dlouhou tabulku

## 2.3 Karta odběratele

Uchovává informace o odběrateli zboží

Tabulka: customer

cust_id	name	person	send_street	send_city	send_PSC	pay_street	pay_city
integer unique	varchar(256)	varchar(50)		varchar(30)	char(5)	varchar(50)	varchar(30)
jednoznačné číslo odběratele	název odběratele	kontaktní osoba	dodací adresa			fakturační adresa	
			ulice	město	směrovací číslo	ulice	město
pay_PSC	tel	fax	mail	web	ICO	DIC	account
char(5)	varchar(20)		varchar(50)	varchar(100)	varchar(20) unique	varchar(20)	varchar(20)
fakturační adresa	telefon	fax	e-mail	internetová adresa	IČO	DIČ	bankovní účet
směrovací číslo							

POZN.: Jedná se o jednu dlouhou tabulku

## 2.4 Tabulky výdejových dokladů – výdejky

Obsahuje doklady o prodeji zboží. Výdejky jsou rozděleny na dvě tabulky. První obsahuje přehled všech výdeků. Druhá obsahuje detailní rozpis zboží se kterým se obchodovalo. Některé sloupce ve výdejce (v tabulce „sales“) jsou shodné jako na kartě zboží. Přesto není možné tyto sloupce vynechat a přechíst údaje z karty zboží, neboť karty zboží je možno editovat, případně smazat. Na výdejce však musí zůstat zachovány informace o proběhnutém prodeji. Obdobně je možno prodat za jinou cenu, než která je na kartě zboží a opět je nutno zachovat údaj za kolik se prodalo (buňka „PC“).

Tabulka: sales\_listing

id_sale_listing	number	date	cust_id	total_PC_DPH	total_PC	reduction	author	user_id
integer primary key	integer	date	integer foreign key	decimal(20,2)			varchar(50)	integer foreign key
identifikační číslo výdejky	číslo výdejky, určené podle číselné řady	datum vystavení výdejky	jednoznačné číslo odběratele, kterému bylo prodáno	prodejní cena za kterou bylo prodáno (ovlivněna případnou slevou) [Kč]		sleva, nebo přírůžka poskytnutá na zboží [Kč]	jméno uživatele, který provedl výdej	jednoznačné číslo uživatele

Tabulka: sales

id_sale	id_sales_listing	goods_id	save_goods_id	name	dph	PC	quantity	unit
integer primary key	integer foreign key	varchar(20) foreign key	varchar(20)	varchar(256)	integer	decimal(12,2)	integer	char(5)
jednoznačné číslo výdejky	číslo výdejky ke které tento řádek náleží	jednoznačné skladové číslo zboží	skladové číslo, slouží pro uchování informace při výmazu zboží z databáze	jméno prodaného zboží (může být rozdílné od karty zboží – karty je možno měnit)	daň [%]	cena za kterou bylo prodáno bez daně (může být rozdílná od ceny na kartě zboží – ceny je možno měnit) [Kč]	množství prodaného zboží	množství jednotka [ks, m, kg, ...]

## 2.5 Tabulka příjmových dokladů – příjemky

Obsahuje archivaci dokladů o nákupu zboží. Struktura tabulek a systém je stejný jako u výdeků.

Tabulka: buy\_listing

id_buy_listing	number	date	sup_id	total_NC_DPH	total_NC	reduction	author	user_id
integer primary key	integer	date	integer	decimal(12,2)			varchar(50)	integer
identifikační číslo výdejky	jednoznačné číslo příjemky, určené podle číselné řady	datum vystavení příjemky	jednoznačné číslo dodavatele, od kterého bylo nakoupeno	nákupní cena s DPH za kterou bylo nakoupeno [ovlivněna případnou slevou]		sleva, nebo přírůžka poskytnutá na zboží [Kč]	jméno uživatele, který provedl příjem	jednoznačné číslo uživatele

Tabulka: buy

id_buy	id_buy_listing	goods_id	save_goods_id	name	dph	NC	quantity	unit
integer primary key	integer foreign key	varchar(20) foreign key	varchar(20)	varchar(256)	integer	decimal(20,2)	integer	char(5)
jednoznačné identifikační číslo příjemky	číslo příjemky, ke které tento řádek náleží	jednoznačné skladové číslo zboží	skladové číslo, slouží pro uchování informace při výmazu zboží z databáze	jméno přijmutoho zboží (může být rozdílné od karty zboží – karty je možno měnit)	daň [%]	cena za kterou bylo nakoupeno bez daně (může být rozdílná od ceny na kartě zboží – ceny se mohou měnit)	množství přijmutoho o zboží	množství jednotka [ks, m, kg, ...]

## 2.6. Tabulka inventur

Uchovává výsledky provedených inventur. Inventury jsou uloženy ve dvou tabulkách. První tabulka obsahuje přehled provedené inventury. Druhá tabulka poskytuje podrobné informace o zboží, které bylo spočítáno při inventuře. Stejně jako v předchozích tabulkách, i zde jsou potřeba buňky tabulky shodně pojmenované jako na kartě zboží, aby byly tyto údaje zachovány při případné změně karty zboží.

Tabulka: stocking listing

id	number	date	difer NC	author	user id	text	is lock
integer primary key	integer	date	decimal(12,2)	varchar(50)	integer foreign key	varchar(256)	boolean
jednoznačné číslo inventury	číslo inventury, určené podle číselné řady	datum provedení inventury	zjištěný celkový rozdíl v nákupních cenách [Kč]	jméno uživatele, který provedl inventuru	jednoznačné číslo uživatele	textová poznámka k inventuře	označuje, zda se má období před inventurou uzamknout

Tabulka: stocking

id stocking	id stocking listing	goods id	name	DPH	NC	difer	unit
integer primary key	integer foreign key	varchar(20) foreign key	varchar(256)	integer	decimal(12,2)	integer	char(5)
jednoznačné identifikační číslo	číslo inventury, ke které náleží tento řádek	jednoznačné skladové číslo zboží	jméno zboží	daň [%]	nákupní cena zboží bez daně	zjištěný rozdíl v množstevních jednotkách [ks, m, kg, ...]	množstevní jednotka [ks, m, kg, ...]

## 2.7. Tabulka rekapitulací

Obsahuje údaje o provedených měsíčních a ročních rekapitulacích. Rekapitulace jsou uloženy ve dvou tabulkách. V první se nachází měsíční rekapitulace. V druhé roční rekapitulace. Každá roční rekapitulace je souhrnem dvanácti měsíců jednoho roku z tabulky měsíčních rekapitulací.

Tabulka: recap month

date	start	profit	release	end	text	author	user id
date primary key	decimal(20,2)				varchar(256)	varchar(50)	integer foreign key
jednoznačné datum určující kterého měsíce se rekapitulace týká	počáteční finanční stav z daní (určený z předchozí rekapitulace) [Kč]	zisk za měsíc z daní (určený z příjemeck za příslušný měsíc) [Kč]	výdej za měsíc z daní (určený z výdejeck za příslušný měsíc) [Kč]	výsledek za měsíc z daní (určený jako $end = start + profit - release$ ) [Kč]	textová poznámka připojená k rekapitulaci	uživatel, který provedl rekapitulaci	jednoznačné číslo uživatele, který provedl rekapitulaci

Tabulka: recap year

datum	begin	profit	release	end	text	author	user id
date primary key	decimal(20,2)				varchar(256)	varchar(50)	integer foreign key
jednoznačné datum určující kterého roku se rekapitulace týká	počáteční finanční stav z daní (určený z předchozí rekapitulace) [Kč]	zisk za rok z daní (určený z příjemeck za příslušné měsíce v roce) [Kč]	výdej za rok z daní (určený z výdejeck za příslušné měsíce v roce) [Kč]	výsledek za rok z daní (určený jako $end = start + profit - release$ ) [Kč]	textová poznámka připojená k rekapitulaci	uživatel, který provedl rekapitulaci	jednoznačné číslo uživatele, který provedl rekapitulaci

## 2. 8. Tabulka uživatelských účtů

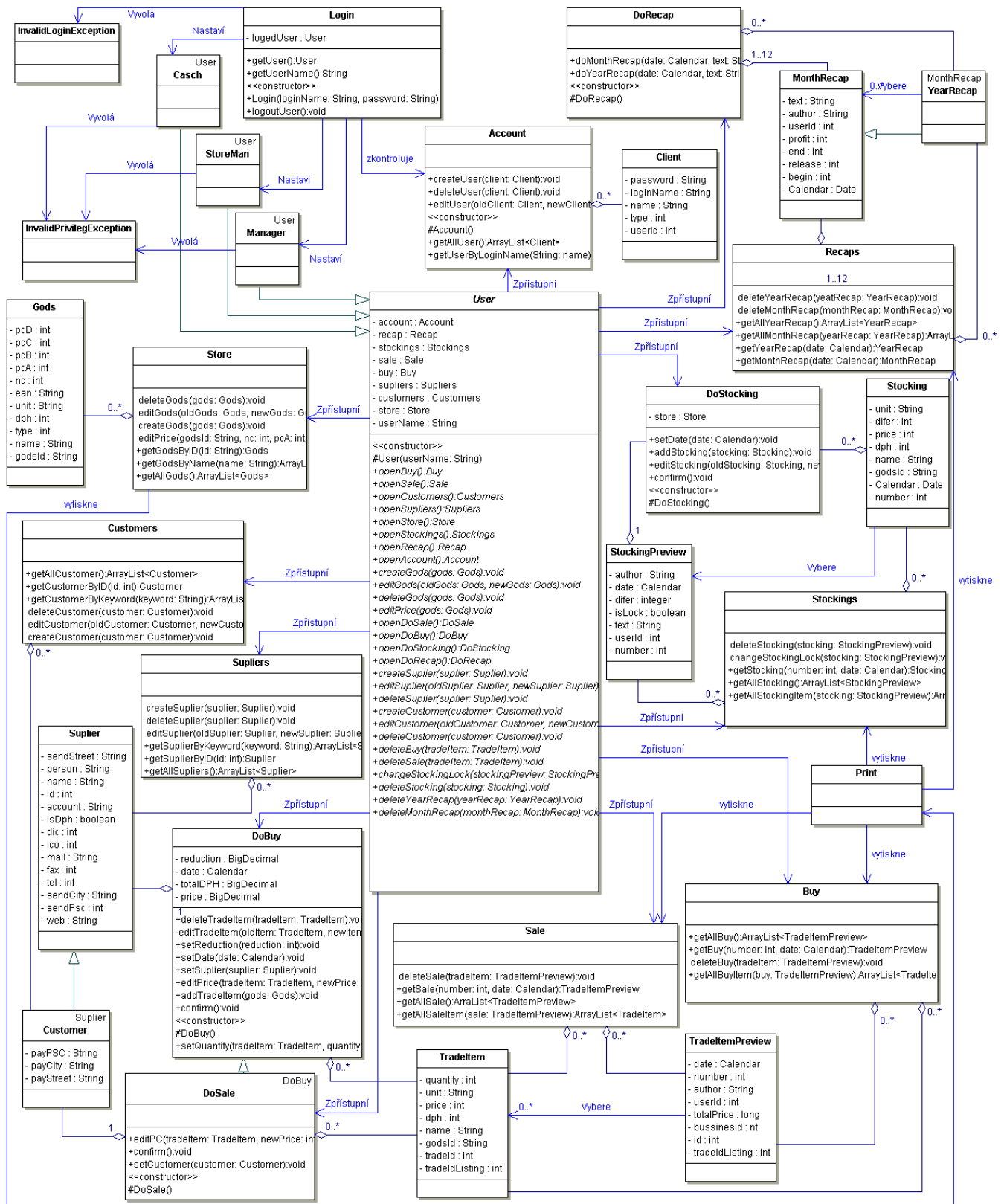
Obsahuje informace o jednotlivých uživateli systému

Tabulka: account

user id	type	name	login name	password
integer primary key	integer	varchar(50) unique	varchar(50)	char(20)
jednoznačné identifikační číslo uživatele	typ uživatele 1 – vedoucí 2 – prodavač 3 – skladník	jméno uživatele	přihlašovací jméno uživatele	přihlašovací heslo uživatele

### 3. Aplikační vrstva

Návrh tříd ukazuje následující diagram



Created by Borland® Together® Designer Community Edition

Diagram obsahuje klíčovou třídu **User**. Jedná se o abstraktní třídu. Tato třída zpřístupňuje jednotlivé funkce uživatelům. Od ní dědí třídy **Manager** (Vedoucí), **StoreMan** (Skladník) a **Casch** (Prodávač). Tyto tři třídy implementují abstraktní metody předka. A to tak, že buď provedou potřebný kód, nebo vyvolají výjimku **InvalidPrivilegeException** podle toho, zda má uživatel oprávnění příslušnou akci provést.

**Zadavatel:** AKJ servis spol. s r. o.

**Zpracoval:** Kamil Ježek

Propojení příslušného uživatele z prezenční vrstvy do aplikační provede třída **Login**, která přihlásí uživatele a vrátí příslušnou instanci třídy **User**. Vyvolá vyjímku **InvalidLoginException**, jestliže byly zadány špatné přihlašovací údaje.

Přihlášení může vypadat například takto:

```
Login login;  
try {  
    login = new Login(„jmeno“, „heslo“);  
    catch (InvalidLoginException e) {  
        // chybné přihlášení - chybné heslo, nebo jméno  
    }  
    User user = login.getUser(); // Automaticky vrátí příslušnou instanci, podle přihlašovacích údajů  
    // Nyní je možné přistoupit k funkcím, které jsou dostupné přihlášenému uživateli
```

Dále třída obsahuje celkem tři typy tříd, které je možno volat z prezenční vrstvy:

### 1. Třídy přístupné jen vybraným uživatelům

Takové třídy obsahují konstruktor přístupný pouze v aplikační vrstvě a instanci třídy lze získat pouze přes třídu **User**. Metody třídy jsou již veřejné.

Takto fungují třídy v diagramu začínající předponou „Do“ – jsou to třídy zajišťující určitou akci: **DoBuy** (Příjem zboží), **DoSale** (Výdej zboží), **DoStocking** (Provedení inventury), **DoRecap** (Provedení rekapitulace) a také třída **Account** (Uživatelské účty).

Princip použití takové třídy:

```
DoSale doSale;  
try {  
    doSale = user.OpenDoSale();  
    catch (InvalidPrivilegeException e) {  
        // uživatel nemá oprávnění používat tuto třídu  
    }  
    doSale.editCustomer(.....); //Metody třídy jsou již přístupné a lze je volat přímo přes instanci
```

### 2. Třídy s veřejně přístupnými i omezeně přístupnými metodami

Takové třídy obsahují veřejný konstruktor. Lze tedy volně vytvořit instanci. Dále třída obsahuje veřejné metody, které lze přes tuto instanci volat. Mimo to ovšem třída obsahuje metody přístupné pouze z aplikační vrstvy. Ty opět zpřístupňuje třída **User**.

V diagramu takto fungují třídy sloužící pro přehled uložených dat: **Store** (Sklad), **Customers** (Odběratelé), **Suppliers** (Dodavatele), **Sale** (Příjemky), **Buy** (Výdejky), **Stockings** (Inventury), **Recaps** (Rekapitulace).

Princip použití takové třídy:

```
Sale sale  
try {  
    sale = user.OpenSale();  
    catch (InvalidPrivilegeException e) {  
        // uživatel nemá oprávnění používat tuto třídu  
    }  
    sale.getSale(.....); // metoda přístupná pro všechny uživatele  
    try {  
        user.deleteSale(....); //mazat může jenom oprávněný uživatel, proto je třeba volat metodu třídy User  
    } catch (InvalidPrivilegeException e) {  
        // uživatel nemá oprávnění používat tuto třídu  
    }  
}
```

### 3. Finální neměnné třídy s veřejným přístupem

Takové třídy jsou libovolně přístupné. Obsahují pouze konstruktor nastavující instanční proměnné a metody pro získání těchto hodnot („gettry“). Neobsahují žádné nastavovací metody („settry“). Představují jednotlivé položky údajů. Jsou využity k uchování položek načtených z databáze.

Jedná se o třídy: **Goods** (Karta zboží), **Supplier** (Karta dodavatele), **Customer** (Karta odběratele), **TradeItem** (Položka z jedné příjemky, nebo výdejky), **TradeItemPreview** (Přehled příjemky, nebo výdejky), **Stocking** (Položka z jedné inventury), **StockingPreview** (Přehled inventury), **MonthRecap** (Měsíční rekapitulace), **YearRecap** (Roční rekapitulace), **Client** (Karta uživatele).

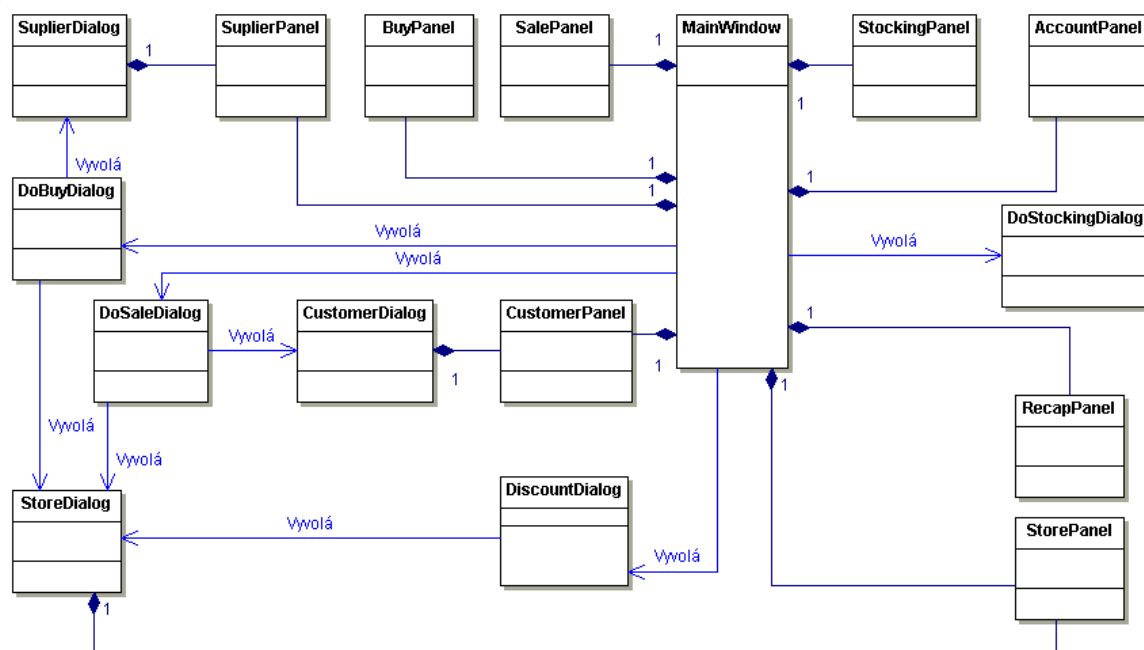
Použit lze prostě:

```
Goods gods = new Goods(.... );  
gods.getQuantity();
```



## 4. Prezenční vrstva

Návrh tříd:



Created by Borland® Together® Designer Community Edition

Prezenční vrstva vytvoří hlavní okno (třída **MainWindow**), to bude obsahovat hlavní menu, nástrojovou lištu a sadu záložek (**JTabbedPane**). Jednotlivé záložky představují jednotlivé moduly programu – *Sklad*, *Dodavatele*, *Odběratele*, *Příjemky*, *Výdejky*, *Inventury*, *Uzávěrky* a *Uživatelské účty*.

Obsah každé záložky bude tvořit příslušný **JPanel**, konkrétně: Sklad – **StorePanel**, Dodavatel – **SupplierPanel**, Odběratel – **CustomerPanel**, Příjemky – **BuyPanel**, Výdejky – **SalePanel**, Inventury – **StockingPanel**, Uzávěrky – **RecapPanel** a Uživatelské účty – **AccountPanel**.

Dále je možno z hlavního okna vyvolávat dialogy pro provedení určitých akcí a to: *Příjem zboží* – **DoBuyDialog**, *Výdej zboží* – **DoSaleDialog**, *Provedení inventury* – **DoStockingDialog**. Poslední dvě akce, které program umožňuje – *Rekapitulace* a *Nastavení uživatelských účtů* – se provádějí přímo z hlavního okna a neotvírá se žádné dialogové okno. Dále z dialogových oken *Výdej zboží* a *Příjem zboží* je možno otevřít další dialog *Sklad* a *Odběratele* nebo *Dodavatele*. Obsah těchto dialogových oken je však tvořen opět panely **StorePanel**, **CustomerPanel** a **SupplierPanel** a je tedy shodný s odpovídajícím obsahem záložky v hlavním okně. Vše zobrazuje diagram.