



ŠKODA AUTO Vysoká škola

# Počítačová simulace logistických procesů

Programování v SimTalk

Jan Fábry

06.04.2023

# Programování v SimTalk

## Cíl přednášky

- Seznámit se základy jazyka SimTalk a práci s metodami.



ŠKODA AUTO Vysoká škola

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Struktura přednášky

- **Metoda.**
  - Struktura objektu metoda.
  - Syntaxe metody.
  - Klíčová slova.
  - Datové typy.
  - Operace.
  - Komentáře.
  - Konvence.
- **Debug.**
- **Watch Window.**

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Co rozumíme pod pojmem METODA

- Metody jsou **malé části programu**, srovnatelné s procedurou nebo funkcí, napsaném v programovacím jazyce (např. Basic, Pascal, C++).
- V Plant Simulation se programuje pomocí programovacího jazyka „**SimTalk**“:
  - SimTalk je odvozen od programovacího jazyka „**Eiffel**“ a je podobný ostatním programovacím jazykům.
- Metoda je základním prvkem Plant Simulation a je plně **integrována do všech jeho objektů**. Zdrojový kód se zpracovává při spuštění simulace (**Init**) či jejím ukončení (**EndSim**), nebo v průběhu simulačního běhu (pokud je iniciován).



# Programování v SimTalk

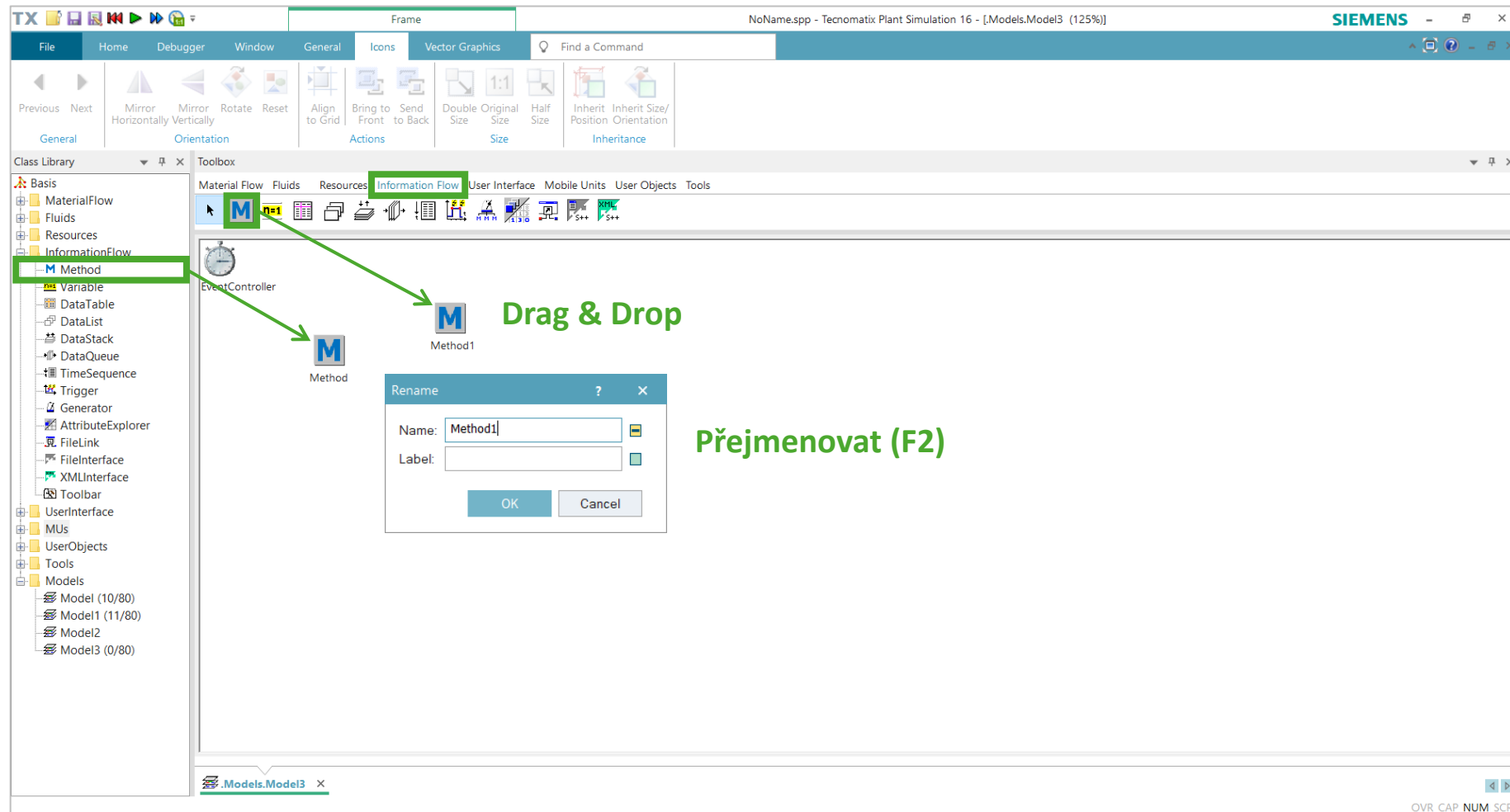
## Možnosti, využitelnost a přednosti metod

- Reagují v určitých událostech v **průběhu simulačního běhu**.
- Dotazují se na podmínky **definované uživatelem**.
- Stanovují **podmínky pro běh simulace**.
- Provádí **příkazy**.
- Mění a rozšiřují **chování jednotlivých prvků** simulačního modelu.
- Pomocí **dialogů** nabízí možnost vytvořit model i pro nezkušené uživatele.
- **Využitelnost** metod u všech simulačních modelů.
- Model se pomocí metod stává:
  - **Přizpůsobivým**.
  - **Plně ovladatelným**.
  - **Snadno měnitelným**.

# Programování v SimTalk



## Prvek Metoda

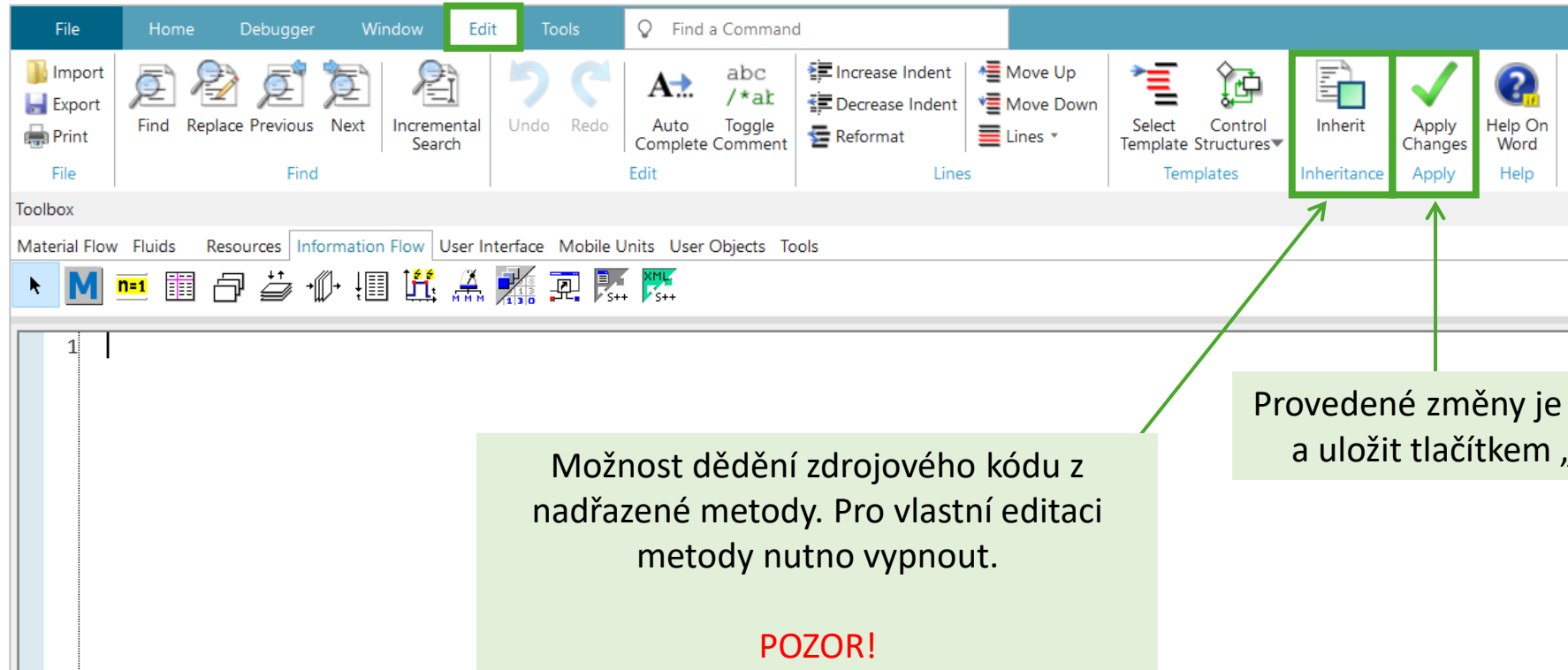


# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Základní okno metody, dědění, ukládání



Možnost dědění zdrojového kódu z nadřazené metody. Pro vlastní editaci metody nutno vypnout.

**POZOR!**

V případě, že po napsání vlastního zdrojového kódu aktivujete funkci dědění, vše se přepíše. Krok zpět není možný!

Provedené změny je vždy třeba potvrdit a uložit tlačítkem „Apply Changes“.

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Základní syntaxe metody

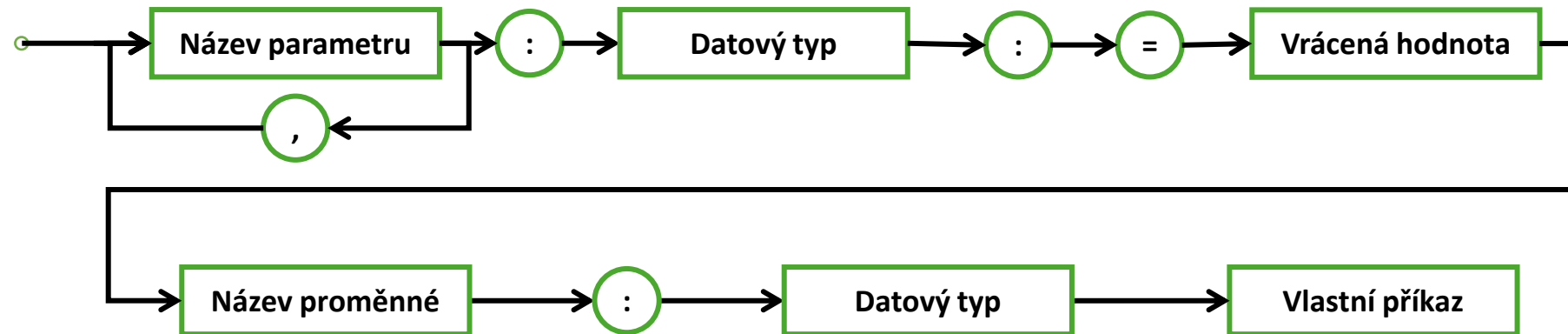
[PARAMETR]	→	<b>Dodatečné informace objektu</b> , který metodu vyvolal, metoda se stává <b>funkcí</b> , počet předaných a deklarováných parametrů i datové typy musí souhlasit.
[VÝSLEDEK FUNKCE]	→	<b>Výsledek funkce</b> (musí být při vrácení hodnoty přiřazen na vyvolávací objekt pomocí klíčového slova „RESULT“).
[LOKÁLNÍ PROMĚNNÉ]	→	Definují se a jsou k dispozici <b>pouze v uvedené metodě</b> . Po zpracování ztrácí svou hodnotu. Při každém novém vyvolání je znovu definována a musí se před použitím deklarovat (definují se pomocí slova „VAR“).
[VLASTNÍ PŘÍKAZY]	→	<b>Provádění</b> metody.

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Syntaxový diagram metody



# Programování v SimTalk



## Syntaxový diagram metody

- Příklad – funkce vrátí větší ze dvou zadaných hodnot:

```
1 param number1: integer, number2: integer    --datový typ argumentů
2 -> integer                                  --výsledný datový typ
3
4 if number1 >= number2 then                  --zjištění jaké z čísel je větší
5     result:= number1                       --pokud je hodnota 1 větší než hodnota 2 vrácení hodnoty 1
6 else
7     result:= number2                       --v opačném případě vrácení hodnoty 2
8 end
9
```

```
-> integer    --výsledný datový typ
result:= mGetMax(85,25)    --vrácení maxima ze zadaných hodnot pomocí vytvořené metody
```

*Volání z jiné metody*

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Klíčová slova SimTalk

---

acceleration	downto	method	result	to
and	else	mod	return	true
any	elseif	next	root	until
array	end	not	rootfolder	var
basis	exitloop	object	self	void
boolean	false	or	speed	wait
byref	for	param	stack	waitexpired
continue	forget	pi	stopuntil	waituntil
create	if	print	string	weight
current	integer	prio	switch	when
date	length	queue	table	while
datetime	list	real	then	
div	loop	repeat	time	

---

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Datové typy

- **Integer** celá čísla (-2 147 483 648; +2 147 483 647)
- **Real** reálná čísla
- **Time** čas (hh:mm:ss.ss)
- **Date** datum (1.1.1970 – 31.12.9999)
- **DateTime** datum včetně času (dd.MM.yyyy HH:mm:ss)
- **Boolean** pravda, nepravda
- **String** řetězec, uvádí se do horních uvozovek " "
- **Object** objekt v simulačním modelu
- **Table** tabulka



# Programování v SimTalk

## Datové typy

- **Length** přiřazená jednotka [m]\*
  - **Speed** přiřazená jednotka [m·sec<sup>-1</sup>]\*
  - **Acceleration** přiřazená jednotka [m·sec<sup>-2</sup>]\*  
použití u objektů „Line“ a „Transporter“
  - **Weight** přiřazená jednotka [kg]\*
  - **Money** obecná proměnná, využívá datový typ REAL
  - **List** sdílí vestavěné vlastnosti objektu CARDFILE
  - **Stack** obdobná funkce jako STACKFILE, využívá LIFO
  - **Queue** vestavěné vlastnosti objektu QueueFile, využívá FIFO
  - **Any** univerzální proměnná, může nabývat všech hodnot
- Rozsah hodnot:  
 $-8,9 \cdot 10^{307} \leq \text{real} \leq 8,9 \cdot 10^{307}$

!Datové typy TIME, LENGTH, WEIGHT, SPEED, ACCELERATION nejsou kompatibilní.

\* Při výstupu se jednotky převedou na jednotky nastavené uživatelem v sekci **Tools > Model Settings/Preferences > Units >**.

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Operace

- **Přiřazení hodnoty:**
  - `<Objekt>.<atribut> := <nová hodnota>`
- **Logické operátory:**
  - Symboly ( = , /= , > , >= , < , <= , ~= )
  - Hodnoty ( true , false )
  - Operátory pro spojení ( and , or , not )
- **Aritmetické operátory:**
  - Sčítání, odčítání, násobení, dělení ( + , - , \* , / )
  - Funkce (goniometrická, logaritmická, exponenciální)
- **Vstupní a výstupní operace:**
  - Slouží pro přenos dat ze vstupních a výstupních parametrů

# Programování v SimTalk



## Logické operátory

- Operátory pro datový typ **Integer**:

Symbol	Popis	Typ výsledku	Příklad
+	Sčítání	integer	
-	Odčítání	integer	
*	Násobení	integer	
//	Celočíselné dělení	integer	17//5=3
\%	Zbytek po dělení	integer	17\%5=2
/	Dělení	integer, real	
=	Roven	boolean	
/=	Není roven	boolean	
>	Větší	boolean	
<	Menší	boolean	
>=	Větší nebo roven	boolean	
<=	Menší nebo roven	boolean	

# Programování v SimTalk



## Logické operátory

- Operátory pro datový typ **Real** a **Boolean**:

Symbol	Popis	Typ výsledku
+	Sčítání	real
-	Odčítání	real
*	Násobení	real
/	Dělení	real
=	Roven	boolean
/=	Není roven	boolean
>	Větší	boolean
<	Menší	boolean
>=	Větší nebo roven	boolean
<=	Menší nebo roven	boolean
~=	Přibližně roven	boolean
<~=	Menší nebo přibližně roven	boolean
>~=	Větší nebo přibližně roven	boolean

# Programování v SimTalk



## Logické operátory

- Operátory pro datový typ **String**:

Symbol	Výraz	Popis	Typ výsledku	Syntaxe
<b>+</b>		Sčítání	string	
<b>=</b>		Roven	boolean	
<b>/=</b>		Není roven	boolean	
<b>==</b>		Přibližně roven (nerozlišují se malá a velká písmena)	boolean	
	<b>strToLower</b>	všechna velká písmena změní na malá	string	<i>strToLower(&lt;string&gt;)</i>
	<b>strToUpper</b>	všechna malá písmena změní na velká	string	<i>strToUpper(&lt;string&gt;)</i>
	<b>strCopy</b>	zkopírování části řetězce	string	<i>strcopy(&lt;string&gt;,&lt;integer1&gt;,&lt;integer2&gt;)</i>
	<b>strIncl</b>	vkładání textu do řetězce	string	<i>strincl(&lt;string1&gt;,&lt;string2&gt;,&lt;integer&gt;)</i>
	<b>strOmit</b>	mazání části řetězce	string	<i>stromit(&lt;string&gt;,&lt;integer1&gt;,&lt;integer2&gt;)</i>
	<b>strlen</b>	určí délku řetězce	integer	<i>strlen(&lt;string&gt;)</i>
	<b>strlpos</b>	určí pozici textu v řetězci	integer	<i>strpos(&lt;string1&gt;,&lt;string2&gt;)</i>

# Programování v SimTalk



## Logické operátory

- Výrazy datového typu **Boolean** (TRUE, FALSE):
  - Výsledek logické hodnoty při použití spojek **and**, **or**, **not**.

Proměnná	Spojka	Proměnná		Výsledek
TRUE	<b>and</b>	TRUE	→	TRUE
TRUE	<b>and</b>	FALSE	→	FALSE
FALSE	<b>and</b>	TRUE	→	FALSE
FALSE	<b>and</b>	FALSE	→	FALSE
TRUE	<b>or</b>	TRUE	→	TRUE
TRUE	<b>or</b>	FALSE	→	TRUE
FALSE	<b>or</b>	TRUE	→	TRUE
FALSE	<b>or</b>	FALSE	→	FALSE
	<b>not</b>	TRUE	→	FALSE
	<b>not</b>	FALSE	→	TRUE

# Programování v SimTalk



## Funkce

- Funkce pro datový typ **String**:
  - **strToLower**
    - `print strToLower("ABCDEF")`      -- vrátí „abcdef“
  - **strToUpper**
    - `print strToUpper("abcdef")`      -- vrátí „ABCDEF“
  - **strCopy**
    - `print strCopy("abcdef",2,3)`      -- vrátí „bcd“
    - `print strCopy("abcdef",4,10)`      -- vrátí „def“
    - `print strCopy("abcdef",-1,4)`      -- vrátí „ab“ (-1 a 0 jsou počítané)

# Programování v SimTalk



## Funkce

- Funkce pro datový typ **String**:
  - **strIncl**
    - `print strIncl("XYZ","abcdef",3)` -- vrátí „abXYZcdef“
    - `print strIncl("XYZ","abcdef",-1)` -- vrátí „XYZabcdef“
    - `print strIncl("XYZ","abcdef",1)` -- vrátí „XYZabcdef“
    - `print strIncl("XYZ","abcdef",20)` -- vrátí „abcdefXYZ“
  - **strOmit**
    - `print strOmit("abcdef",3,2)` -- vrátí „abef,“
    - `print strOmit("abcdef",0,3)` -- vrátí „cdef“ (0 je počítaná)
  - **strlen**
    - `print strlen("abcd")` -- vrátí 4
  - **strpos**
    - `print strlpos("b","abcdfb")` -- vrátí 2

# Programování v SimTalk



## Funkce

- Změna **datového typu**:

Funkce	Argument	Výsledek
to_str	(<...>)	string
num_to_bool	(<integer>)	boolean
bool_to_num	(<boolean>)	real
str_to_num	(<string>)	real
str_to_bool	(<string>)	boolean
str_to_time	(<string>)	time
str_to_date	(<string>)	date
str_to_datetime	(<string>)	datetime
str_to_length	(<string>)	length
str_to_weight	(<string>)	weight
str_to_speed	(<string>)	speed
str_to_obj	(<string>)	object

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Komentář v metodě

- Slouží jako **poznámky** na formulované pokyny v metodě. Pokyny se tak stávají srozumitelnějšími i pro jinou osobu a v samotné metodě tak **zlepšují orientaci**.
- Text je v metodě odlišen **zelenou barvou**.
- Tvorba komentáře:
  - Jednořádkové** komentáře „ -- text “.
  - Víceřádkové** komentáře „/\* text \*/“.

The screenshot shows the SimTalk IDE interface. The top menu bar includes File, Home, Debugger, Window, Edit, and Tools. The Tools menu is open, showing options like Undo, Redo, and Auto Complete. The code editor displays the following code:

```
1
2  --jednořádkový komentář
3
4  /* víceřádkový
5     komentář */
6
```

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Konvence v metodě

- V nápovědě k Plant Simulation platí pro usnadnění čtení následující konvence týkající se pravopisu:
  - **Formulace** v metodách jsou psány **proporčním písmem**.
  - **Názvy atributů** začínají **velkým** písmenem (např. `BufFe.Capacity`).
  - **Názvy metod** začínají **malým** písmenem, skládá-li se metoda ze dvou slov, začíná druhé slovo velkým písmenem (např. `liSOVna.Removeobserver`).
  - **Klíčová slova** jsou psána **modrým** písmem.
  - **Komentáře** jsou psány **zeleným** písmem.

# Programování v SimTalk



## Debug v metodě

- **Okamžité zastavení** v aktuálním kroku vykonávané metody z důvodů hlášení.
- **Důvody hlášení Debug:**
  - Automaticky při chybě, která brání vykonání metody.
  - Manuálně pro vybrané sledování a hledání chyby.
  - Účelově pro hlášení nevykonané akce (např. `if @.Move() then else debug; end;`).
- **Aktivace Debug:**
  - Automaticky při výskytu chyby během doby simulace.
  - Nastavení zarážky (class/instance breakpoint).
  - Manuálně během provádění metody (v nabídce menu, funkční klávesou F8).
  - Přímým napsáním „Debug“ do vlastního příkazu metody.
- Při aktivaci Debug se **otevře okno** zobrazující metodu, která tento stav vyvolala.

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Okno Debug

- Definování lokálních proměnných
- Zarážka (Class Breakpoint)
- Vlastní příkazy
- Aktuální pozice zpracování metody

The screenshot shows the SimTalk IDE's debug window. The main area displays code with a yellow highlight on the `debug` line. A red dot on the left margin indicates a breakpoint. Below the code editor is a 'Watch window' with a table of variables and their values.

```
var a: real := 5
var b: real := 10
var S: real := a * b
debug
print S
```

Name	Value
a	5.0
b	10.0
S	50.0

Sledovací okno (Watch window)

# Programování v SimTalk



## Sledovací okno (Watch window)

- **Informuje** o aktuálním inicializačním a volajícím prvku, stavu proměnných apod.
- Podokno „**Variables**“ a „**Anonymous Identifiers**“:
  - Zobrazení proměnných veličin.
- Sloupec „**Value**“:
  - Zobrazení aktuálního statusu, resp. hodnoty proměnných veličin.
  - „**VOID**“ – status/hodnota není k dispozici (prázdná veličina).

Name	Value
@	VOID
?	VOID
current	.Models.Model
self	.Models.Model.Method
root	.Models.Model
rootfolder	VOID

Name	Value
a	5.0
b	10.0
S	50.0



# Programování v SimTalk

## Sledovací okno (Watch window)

- „@“
  - Anonymní identifikátor pro pohybující se prvky (prvky knihovny MU).
  - Slouží jako iniciátor, tj. ten, který metodu spouští.
- „?“
  - Anonymní identifikátor pro statické prvky (prvky knihovny Material Flow).
  - Obvykle je to volající prvek, tj. ten, kterým je metoda volána.
- „S, a, b“
  - Definované lokální proměnné, které metoda zpracovává.

Name	Value
a	5.0
b	10.0
S	50.0



ŠKODA AUTO Vysoká škola

# Děkuji za pozornost

Jan Fábry

Katedra řízení výroby, logistiky a kvality

✉ [fabry@savs.cz](mailto:fabry@savs.cz)

🌐 [www.janfabry.cz](http://www.janfabry.cz)

[www.savs.cz](http://www.savs.cz)