



ŠKODA AUTO Vysoká škola

# Počítačová simulace logistických procesů

Programování v SimTalk

Jan Fábry

10.12.2024

# Programování v SimTalk

## Cíl přednášky

- Seznámit s možnostmi práce v cyklech a tabulkami.



ŠKODA AUTO Vysoká škola

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Struktura přednášky

- Podmíněné příkazy s **INSPECT**.
- Příkazy s **opakujícím se během**:
  - FOR...NEXT,
  - REPEAT...UNTIL,
  - WHILE...LOOP.
- **Tabulka**.

# Programování v SimTalk



## Podmíněné příkazy s Inspect ...

- Typy:
  - `inspect <výraz>`
    - `when <hodnota> then`
    - `<obecný příkaz>;`
    - `end;`
  - `inspect <výraz>`
    - `when <hodnota> then`
    - `<obecný příkaz>;`
    - `else`
    - `<obecný příkaz>;`
    - `end;`
- Syntaxe „when <hodnota> then <obecný příkaz>“ může být použita i **vícekrát po sobě**.
- **Hodnota/výraz** může být datového typu **integer**, **real** nebo **string**.

# Programování v SimTalk



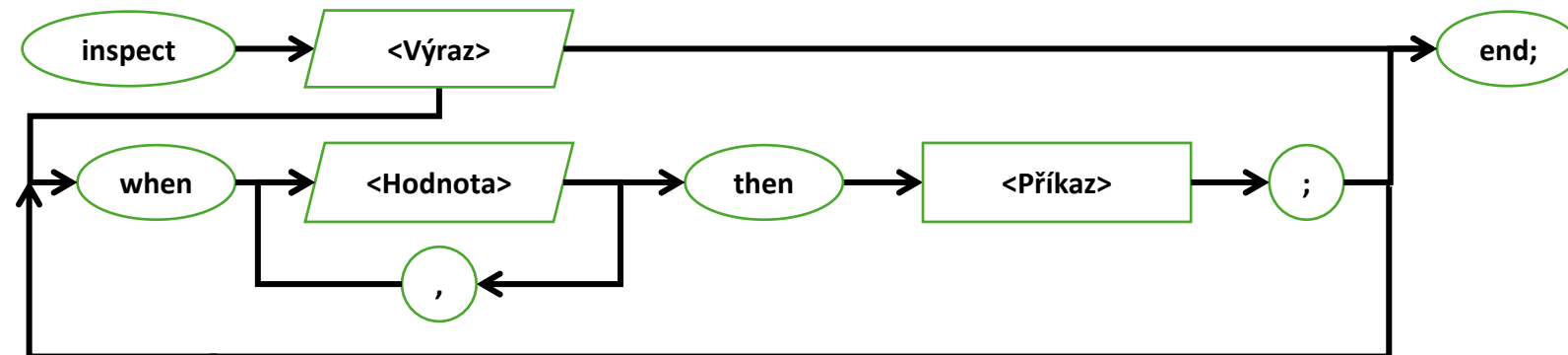
## Podmíněné příkazy s Inspect ...

- Typy:

- `inspect <výraz>`
  - `when <hodnota 1> then`
    - `<obecný příkaz 1>;`
  - `when <hodnota 2> then`
    - `<obecný příkaz 2>;`
- `end;`

```
M.Models.Frame.Method *
File Edit Navigate Run Template View Tools Help
1: is
2:   («expression» : data_type; -- integer, real, string)
3:   -- v prípade nutnosti definovat dátový typ
4: do
5:   inspect «expression»
6:   when «value_list» then
7:     «statements»
8:   when «value_list» then
9:     «statements»
10:  end;
11: end;
```

- Syntaxový diagram:



# Programování v SimTalk



## Podmíněné příkazy s Inspect ...

- Příklad

```
is
  Model : string;
do
  inspect @.Model
    when "Fabia" then
      FabiaCelkem := FabiaCelkem + 1;
    when „Yeti" then
      YetiCelkem := YetiCelkem + 1;
    else
      debug; -- neznámy model
  end;
end;
```

```
.Models.Frame.Method
File Edit Navigate Run Template View Tools Help
1: is
2:   Model : string;
3: do
4:   inspect @.Model
5:     when "Fabia" then
6:       FabiaCelkem := FabiaCelkem + 1;
7:     when "Yeti" then
8:       YetiCelkem := YetiCelkem + 1;
9:     else
10:      debug; -- neznámy model
11:   end;
12: end;
```

# Programování v SimTalk

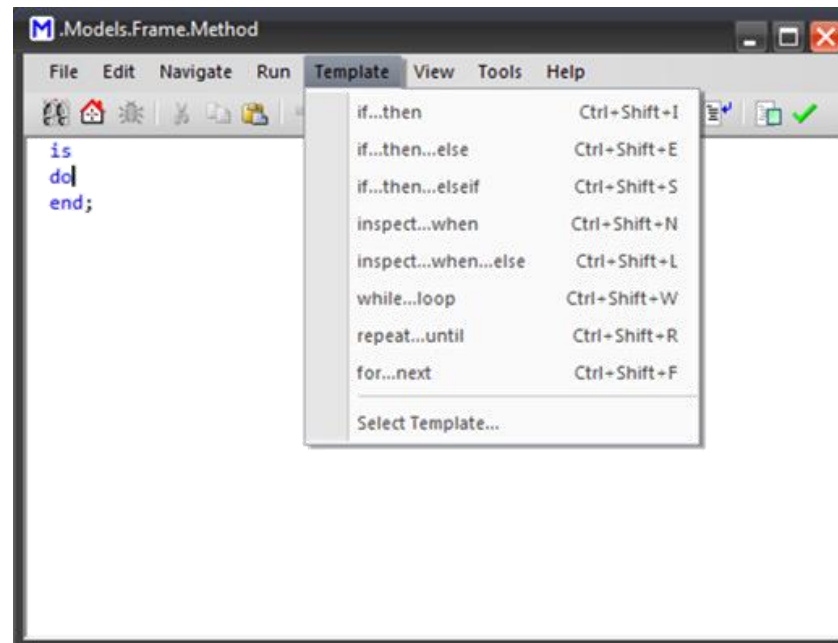


## Příkazy s opakujícím se během

- Cyklus je definován **řídící proměnnou**.
- Příkazy obsažené v cyklu se provádějí, pokud řídící proměnná **nepřesáhne konečnou hodnotu** nebo pokud je **splněna určitá podmínka** nebo naopak dokud **nenastane určitá podmínka**.

- Základní **typy cyklů**:

- for ... next;
- repeat ... until;
- while ... loop;



# Programování v SimTalk



## Příkazy s opakujícím se během s for..next

- Typy:
  - `for` <řídící proměnná := počáteční hodnota> `to` <konečná hodnota> `loop`  
<Sled příkazů ve smyčce>;  
`next`;
    - počáteční hodnota < konečná hodnota
  - `for` <řídící proměnná := počáteční hodnota> `downto` <konečná hodnota> `loop`  
<Sled příkazů ve smyčce>;  
`next`;
    - počáteční hodnota > konečná hodnota

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Příkazy s opakujícím se během s for..next

- Definování řídicí proměnné (`loop_variable`):
  - Řídicí proměnná musí být datového typu `integer`.
  - Definovat se může v `lokálních proměnných` nebo přímo ve vlastním příkaze pomocí klíčového slova „`local`“.
  - Využitím „`local`“ v příkazech s opakujícím se během se řídicí proměnná stává `viditelná` jenom v rámci smyčky daného příkazu.

```
1: is
2:   «loop_variable» : integer;
3: do
4:   for «loop_variable» := «start_value» to «end_value» loop
5:     «loop_statements»
6:   next;
7: end;
```

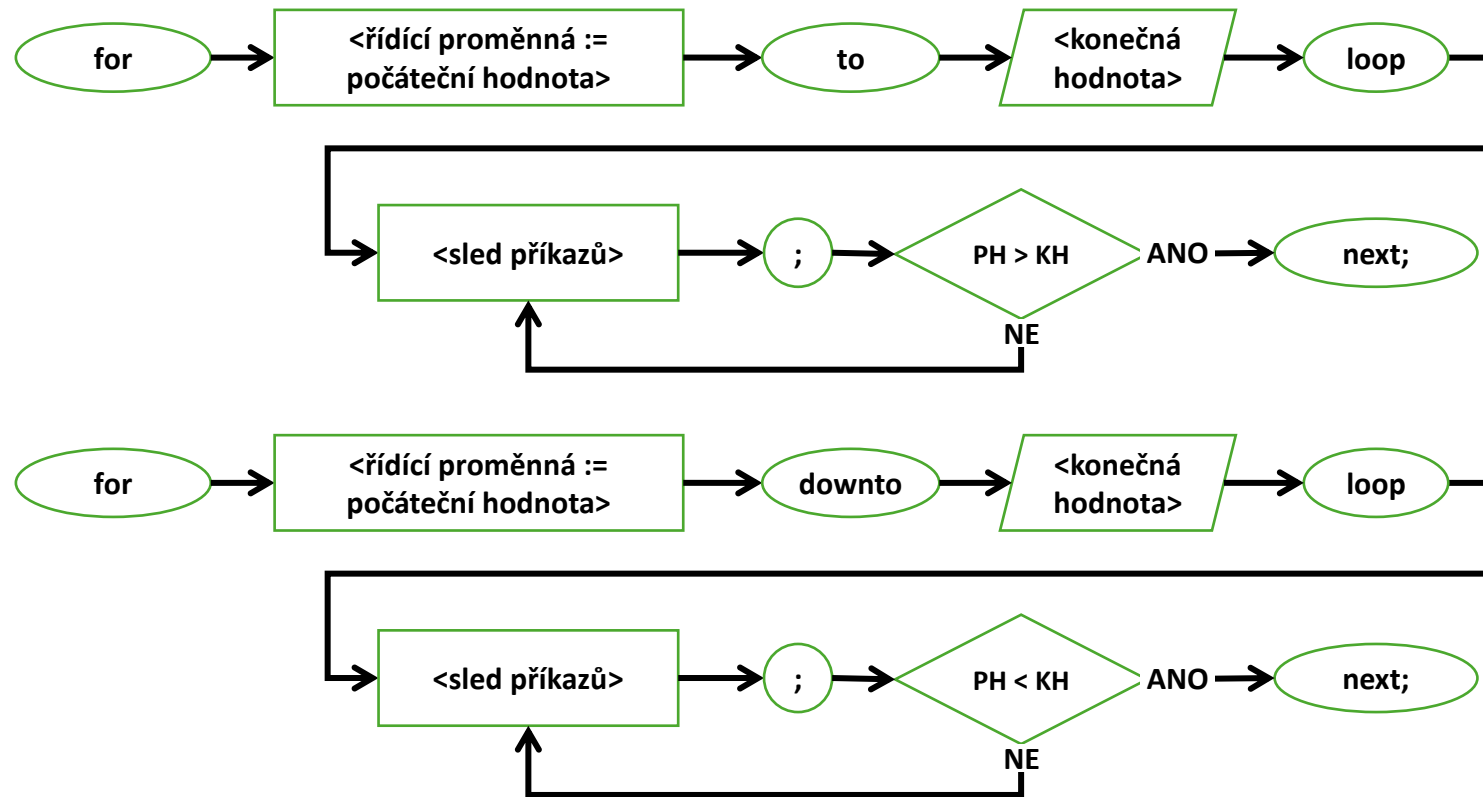
```
1: is
2: do
3:   for local «loop_variable» := «start_value» to «end_value» loop
4:     «loop_statements»
5:   next;
6: end;
```

# Programování v SimTalk



## Příkazy s opakujícím se během s for..next

- Syntaxový diagram:



# Programování v SimTalk



## Příkazy s opakujícím se během s for..next

- Příklad

```
is
  x : integer;
do
  for x := 1 to 3 loop
    Tabulka[x,1] := x;
  next;
  for local y := 1 to 10 loop
    Tabulka[1,y] := y;
  next;
end;
```

The screenshot shows a SimTalk editor window titled ".Models.Frame.Method". The code in the editor is as follows:

```
1: is
2:   x : integer;
3: do
4:   -- to a definovaná řídicí proměnná v lokálních proměnných
5:   for x := 1 to 3 loop
6:     Tabulka[x,1] := x;
7:   next;
8:   -- downto a definování řídicí proměnné pomocí "local"
9:   for local y := 1 to 10 loop
10:    Tabulka[1,y] := y;
11:   next;
12: end;|
```

Below the code, a table is displayed with the following data:

	string 0	integer 1	integer 2	integer 3
string		y		
1	x	1	2	3
2		2		
3		3		
4		4		
5		5		
6		6		
7		7		
8		8		
9		9		
10		10		

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Příkazy s opakujícím se během s repeat ... until

- **repeat**  
    <Sled příkazů ve smyčce>;
- **until** <Podmínka pro ukončení smyčky>;

```
M .Models.Frame.Method *
File Edit Navigate Run Template View Tools Help
1: is
2:   «loop_variable» : integer;
3: do
4:   «loop_variable» := «start_value»;
5:   repeat
6:     «loop_statements»;
7:     «loop_variable» := «loop_variable» + <numerical_value>;
8:   until «exit_condition»;
9: end;
```

# Programování v SimTalk



## Příkazy s opakujícím se během s repeat ... until

### ■ Příklad

```
is
  x : integer;
do
  x := 1;
repeat
  Tabulka[x,1] := x;
  x := x + 1;
until x > 3;
end;
```

The screenshot shows a SimTalk IDE window titled ".Models.Frame.Method1". The code editor contains the following code:

```
1: is
2:   x : integer;
3: do
4:   x := 1;
5:   repeat
6:     TableFile[x,1] := x;
7:     x := x + 1;
8:   until x > 3;
9: end;|
```

Below the code editor is a table with 10 rows and 4 columns. The columns are labeled "string 0", "integer 1", "integer 2", and "integer 3". The first row (index 1) contains the values "x", "1", "2", and "3", which are highlighted in yellow. The rest of the table is empty.

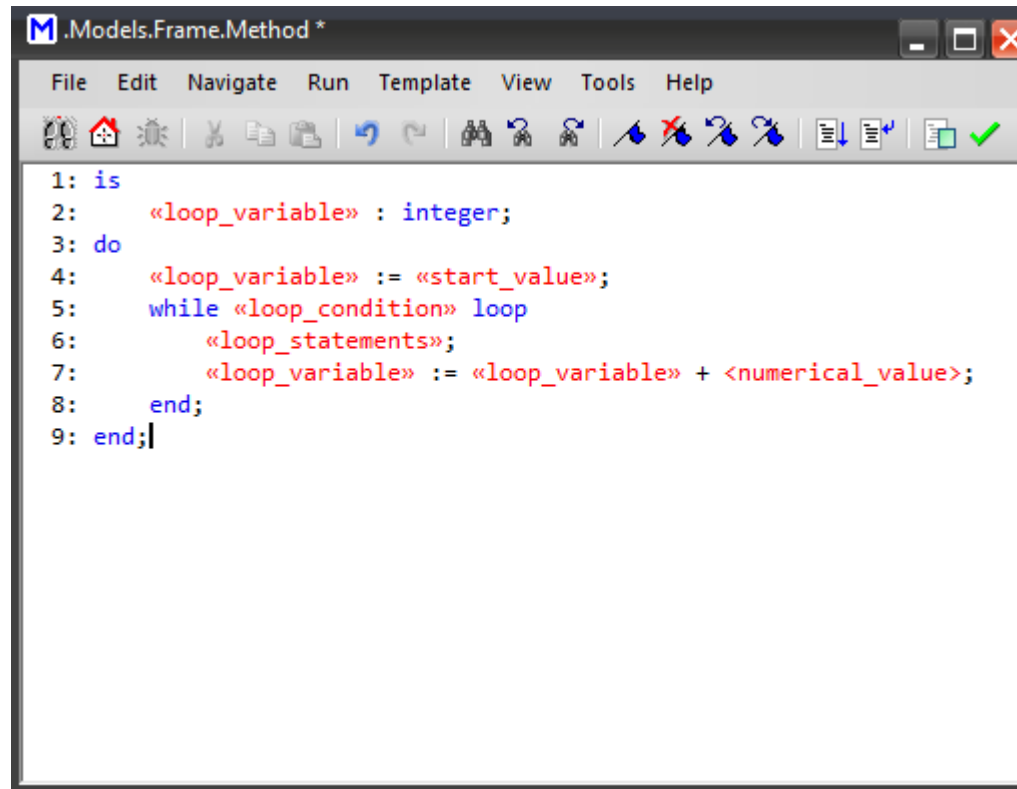
	string 0	integer 1	integer 2	integer 3
1	x	1	2	3
2				
3				
4				
5				
6				
7				
8				
9				
10				

# Programování v SimTalk



## Příkazy s opakujícím se během s while ... loop

- **while** <Podmínka pro běh smyčky> **loop**  
    <Sled příkazů ve smyčce>;  
**end;**



```
.Models.Frame.Method *
File Edit Navigate Run Template View Tools Help
1: is
2:   «loop_variable» : integer;
3: do
4:   «loop_variable» := «start_value»;
5:   while «loop_condition» loop
6:     «loop_statements»;
7:     «loop_variable» := «loop_variable» + <numerical_value>;
8:   end;
9: end;
```

# Programování v SimTalk



## Příkazy s opakujícím se během s while ... loop

- Příklad

```
is
  x : integer;
do
  x := 1;
  while x <= 3 loop
    Tabulka[x,1] := x;
    x := x + 1;
  end;
end;
```

```
.Models.Frame.Method1 *
File Edit Navigate Run Template View Tools Help
1: is
2:   x : integer;
3: do
4:   x := 1;
5:   While x <= 3 loop
6:     TableFile[x,1] := x;
7:     x := x + 1;
8:   end;
9: end;|
```

	string 0	integer 1	integer 2	integer 3
string				
1	x	1	2	3
2				
3				
4				
5				
6				
7				
8				
9				
10				

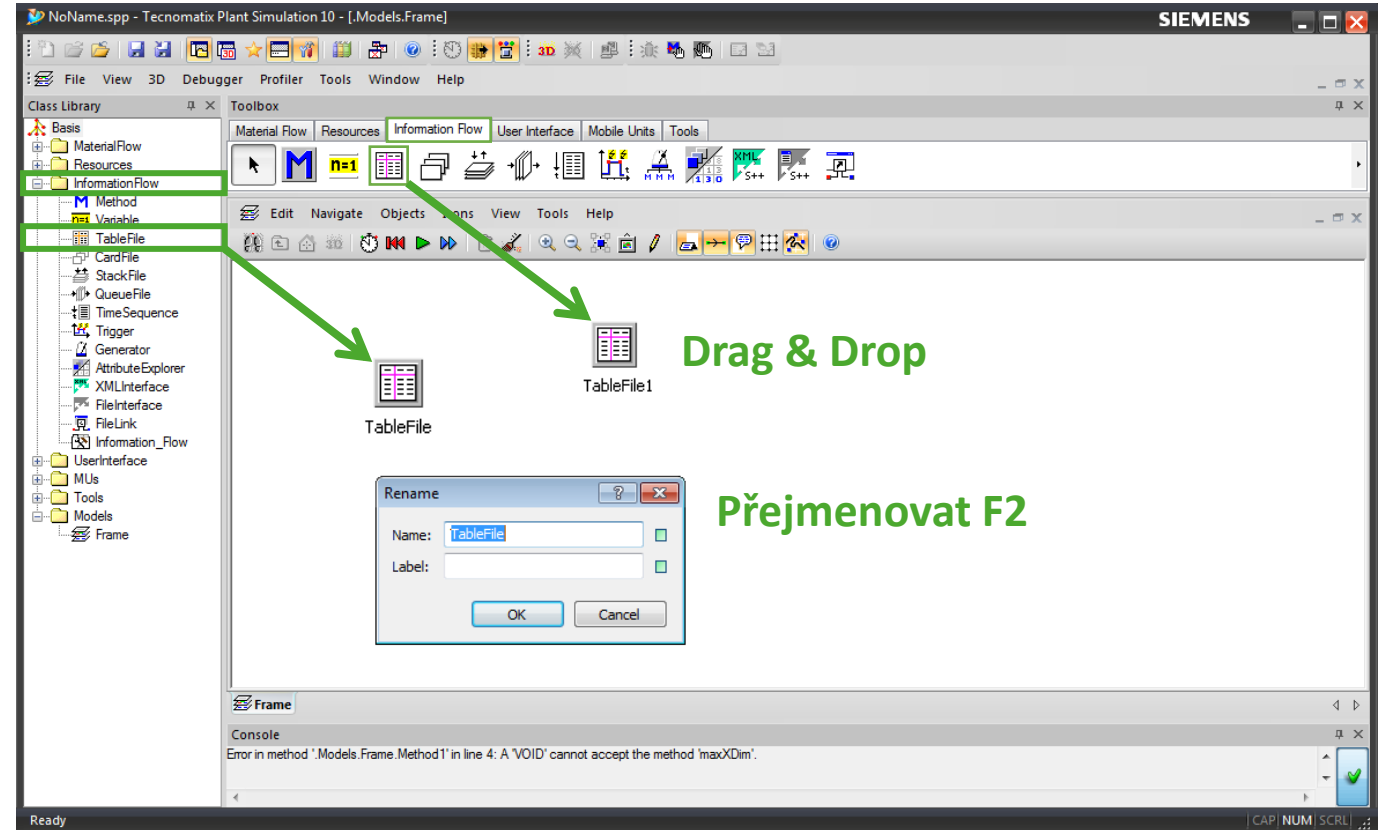
# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka

- Tabulka je **objekt**, složen z **řádků** a **sloupců**.
- Slouží pro **zápis** nebo **čtení** hodnot/výrazů.
- Hodnoty/výrazy se můžou **doplňovat** nebo **mazat** i v průběhu simulačního běhu.



# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka



TableFile

Hlavní nabídka

Přednastavený datový typ pro sloupec

Textové pole zobrazuje/edituje hodnotu/výraz aktivní buňky

	string 1	string 2	string 3	string 4
1				
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				
17				
18				
19				
20				

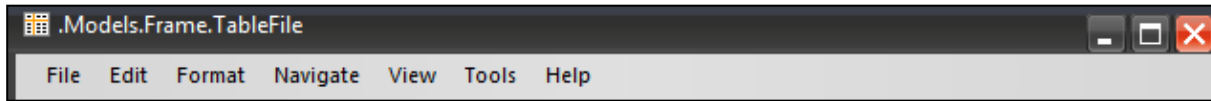
Nástrojová lišta

Tělo tabulky

# Programování v SimTalk



## Tabulka



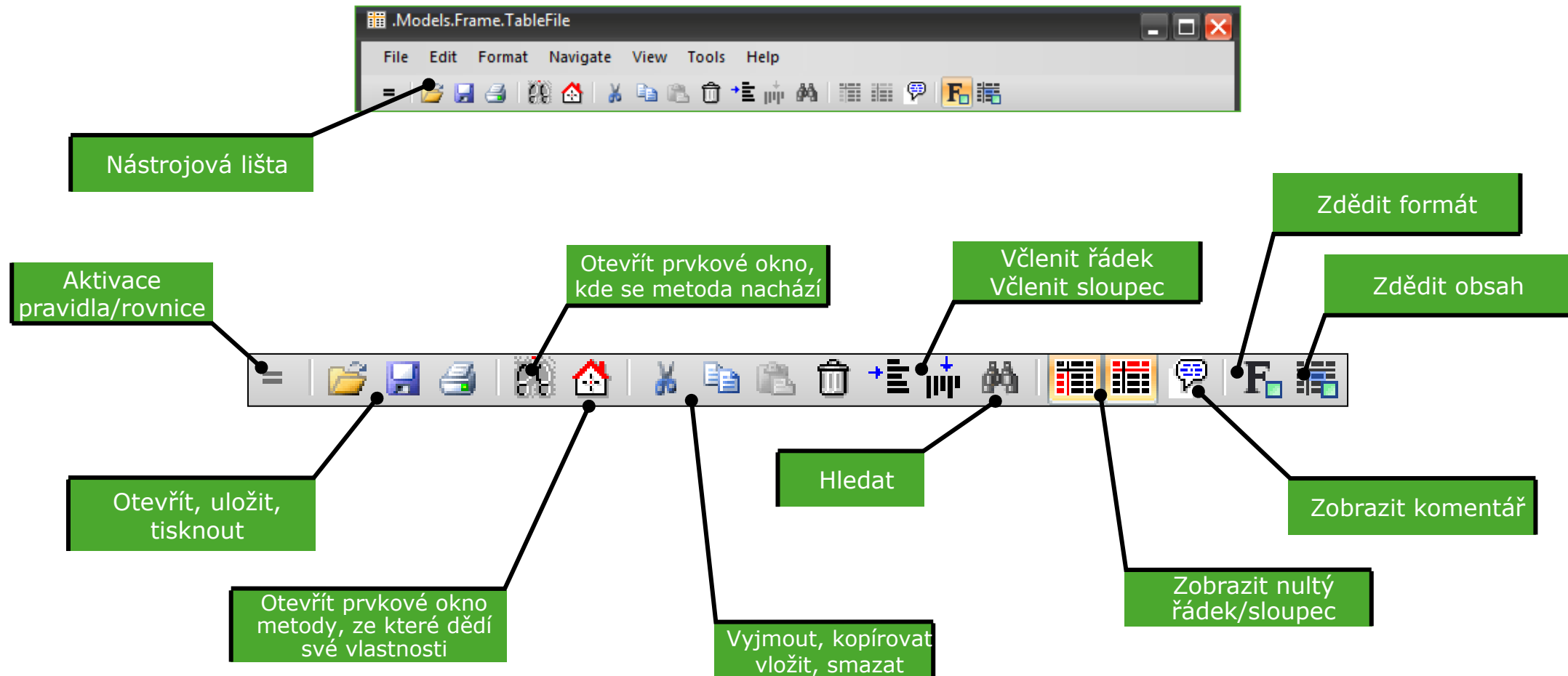
FILE	EDIT	FORMAT	NAVIGATE	VIEW	TOOLS	HELP
SOUBOR	UPRAVIT	FORMÁTOVÁNÍ	NAVIGACE	ZOBRAZENÍ	NÁSTROJE	POMOCNÍK
> otevřít *.obj	> vyjmout	> formátování	> otevřít okno	> přepočítat	> otevřít objekt	
> uložit jako text	> kopírovat	> index sloupců	obsahující	příkazy	> zvolit příkazy	
> uložit jako objekt	> vložit	> index řádků	danou	> zobrazit	> zvolit	
> uložit jako soubor Excel	> smazat	> dědění	tabulku	komentář	pozorovatele	
> uložit jako soubor XML	> označit	formátu	> otevřít	> zobrazit	> uživatelem	
> formát	všechno	> dědění	původní	nástrojovou	definované	
> tisk	> vložit řádek	formátu a	> jít původní	lištu	atributy	
> tisk formátu	> vložit sloupec	obsahu	> otevřít třídu	> zobrazit	> 3D	
> zavřít	> najít	> dědění	> jít na třídy	datový typ	> přejmenovat	
	> nahradit	komentáře		> změna		
	> seřadit			kontrastu		
	vzestupně			buněk		
	> seřadit			> jít na buňku		
	sestupně			> ukázat		
	> příkazy			atributy a		
				objekty		

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka



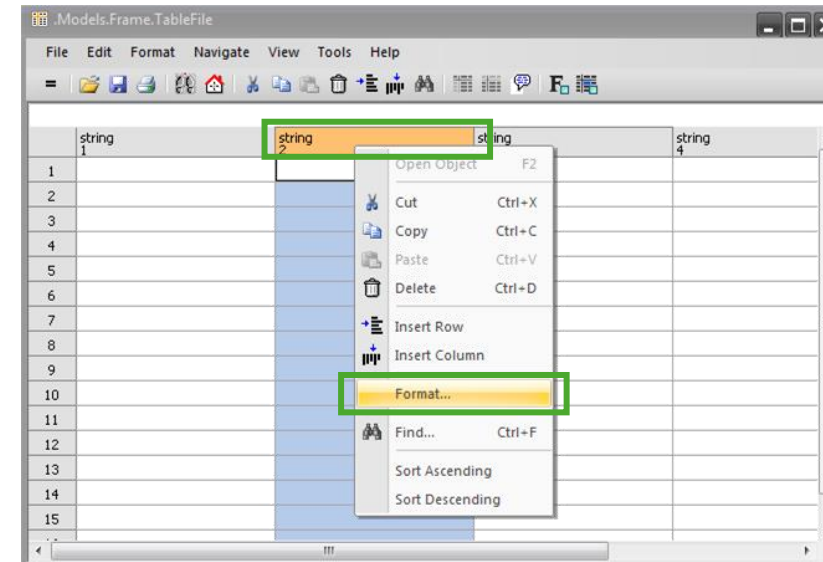
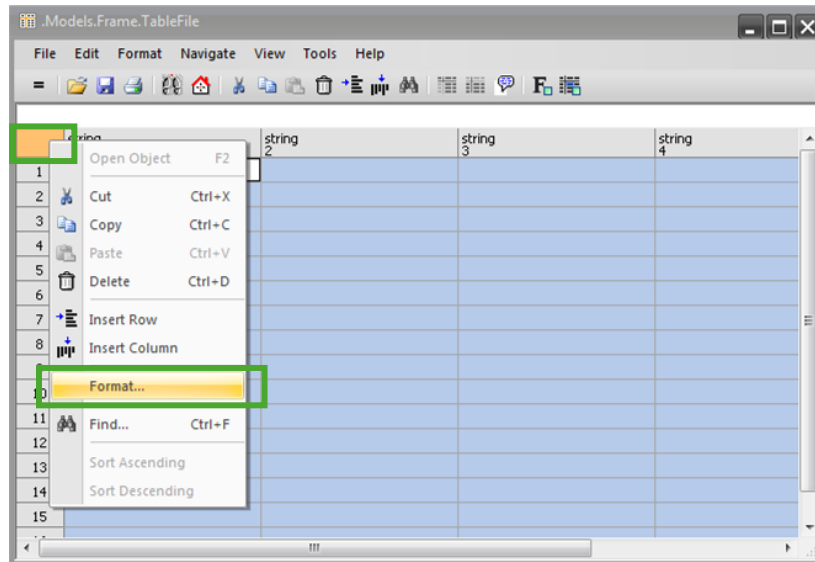
# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka

- Formátování tabulky
  - Možnost formátování celé tabulky, pouze řádků, pouze sloupců.



- Před úpravou je nutno vypnout funkce Dědění!
- Po aktivování funkce Dědění se přeformátuje celá tabulka = ZTRÁTA ÚDAJŮ!

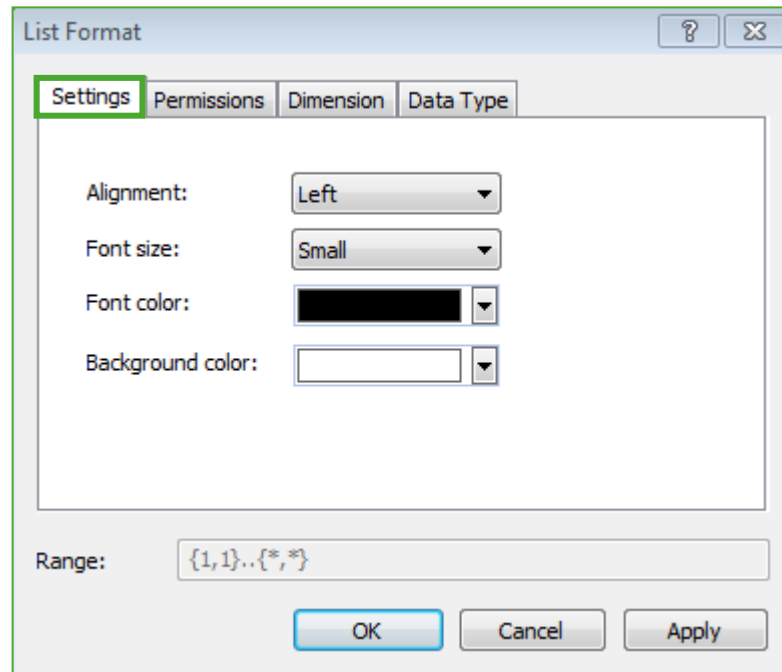
# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka

- **Formátování** tabulky
  - Nastavení zarovnání, velikosti a barvy písma, barva buněk.



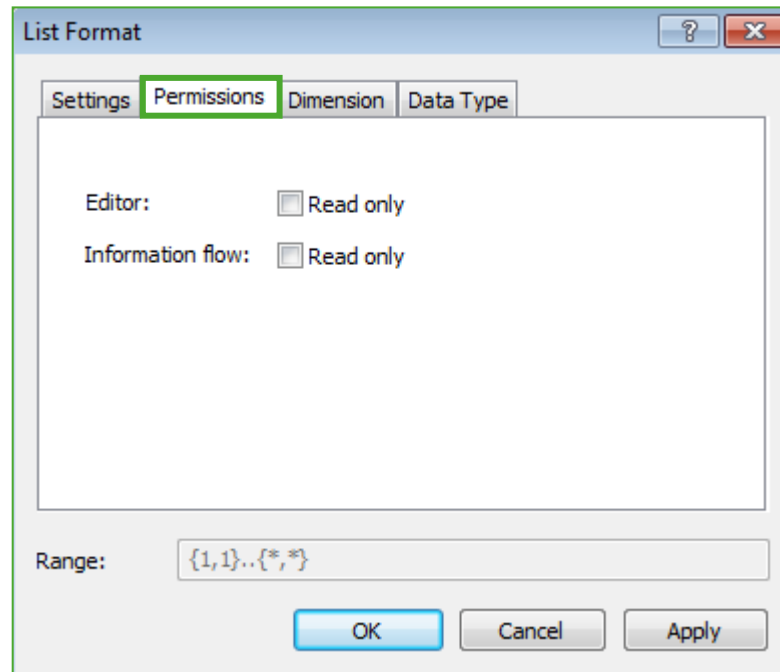
# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka

- **Formátování** tabulky
  - Nastavení povolení pro čtení a zápis.

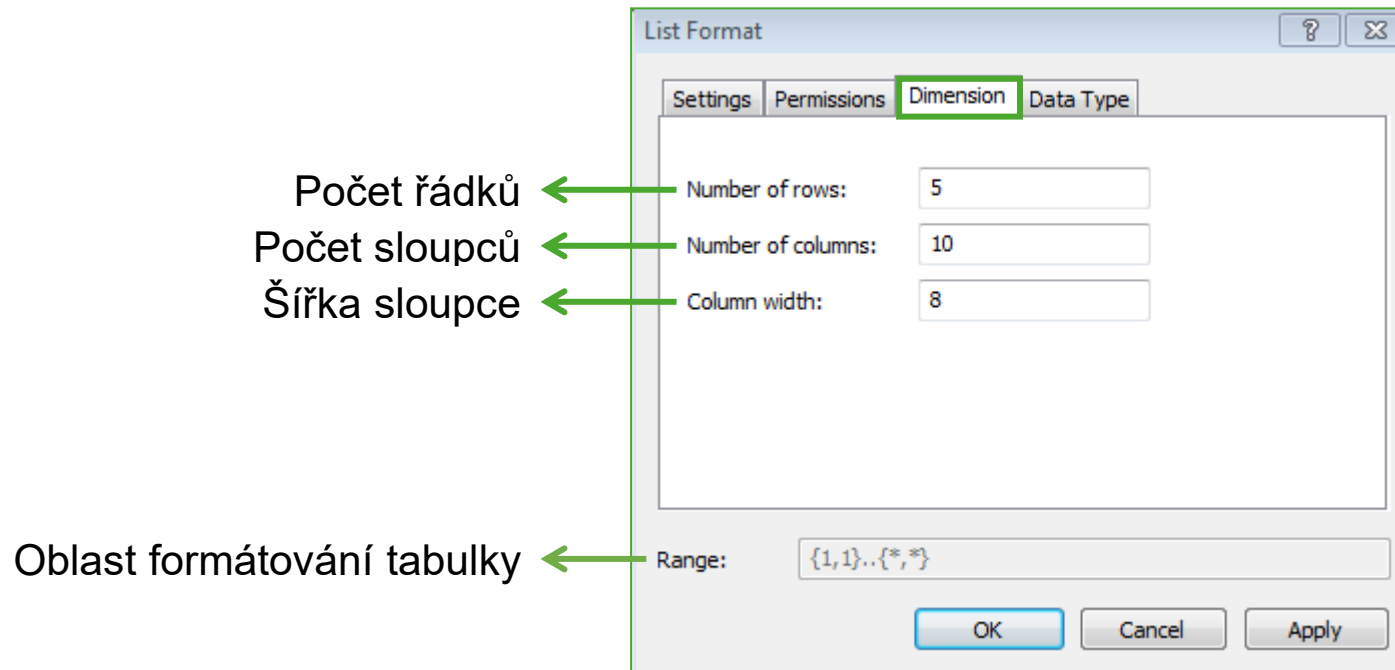


# Programování v SimTalk



## Tabulka

- Formátování tabulky
  - Úprava počtu řádků/sloupců.



# Programování v SimTalk



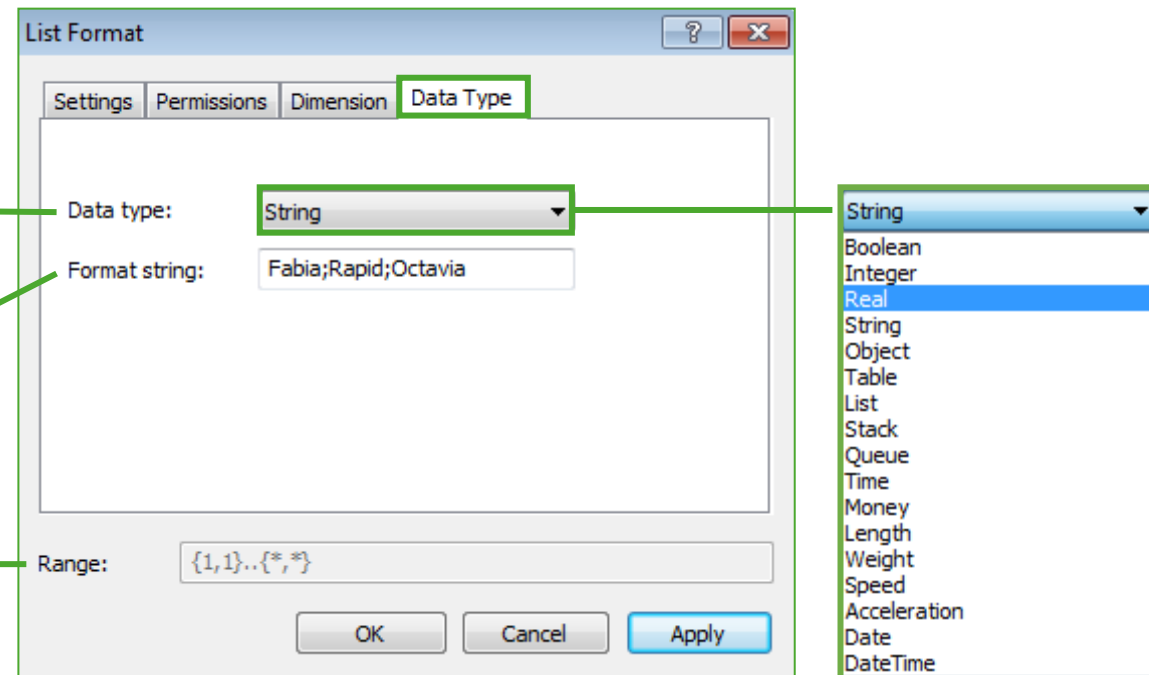
## Tabulka

- Formátování tabulky
  - Úprava datového typu.

Volba datového typu  
pro vybranou oblast formátování

Restrikce pro zadávání textu  
nebo vytvoření rolovací nabídky

Oblast formátování tabulky

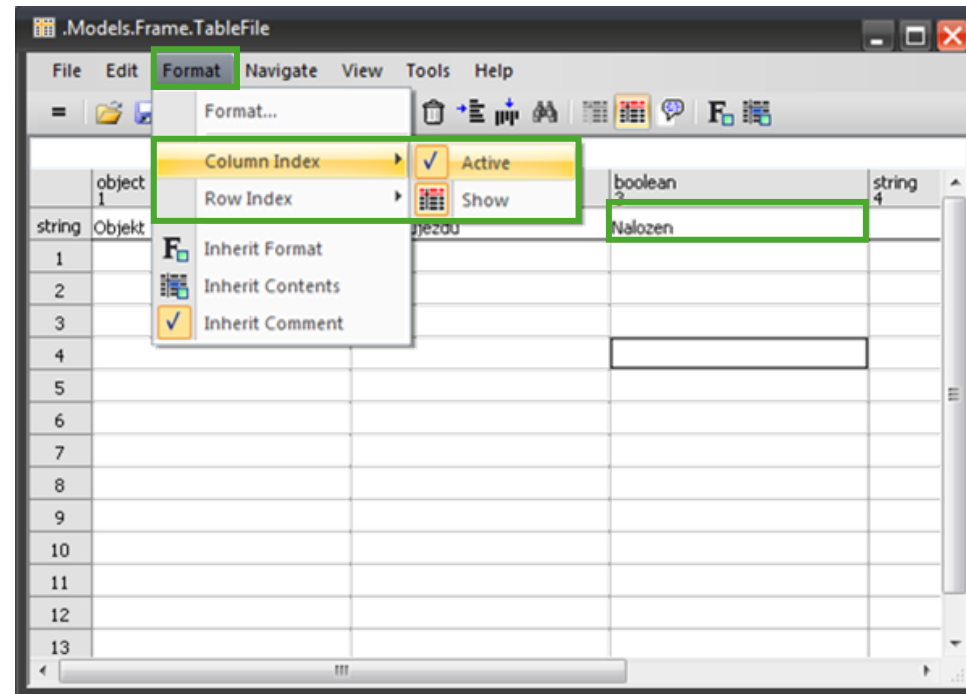


# Programování v SimTalk



## Tabulka

- **Formátování** tabulky
  - Řádkový a sloupcový index
    - Jedná se o **nultý** sloupec, řádek.
    - Slouží pro **lepší orientaci** v tabulce (popis).



# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka

- **Formátování** tabulky
  - Formátování **tabulky** pomocí **metody**
    - Počet sloupců, řádků
      - `<název_tabulky>.MaxxDim := 3;`
      - `<název_tabulky>.MaxyDim := 10;`
  - Formátování **SUBtabulky** pomocí **metody**
    - Počet sloupců, řádků
      - `<název_tabulky>[sloupec,radek].MaxxDim := 2;`
      - `<název_tabulky>[sloupec,radek].MaxyDim := 4;`

# Programování v SimTalk



ŠKODA AUTO Vysoká škola

## Tabulka

- Čtení z tabulky
  - `print <název_tabulky>[<sloupec>,<řádek>];`
- Obecný zápis do tabulky
  - `<název_tabulky>[<sloupec>,<řádek>]:= <hodnota>;`
- Změna vlastnosti prvku z tabulky pomocí metody
  - `<Objekt>.<atribut> := <název_tabulky>[<sloupec>,<řádek>];`
- Mazání obsahu tabulky
  - `<název_tabulky>.delete;`
  - `<název_tabulky>.delete({<sloupec>,<řádek>}..{<sloupec>,<řádek>});`
- Při aktivním nultém sloupci a řádku a jejich vyplnění je možno zadat jako cestu k buňce popis
  - `<název_tabulky>["Objekt","Cas_prujezdu"];`

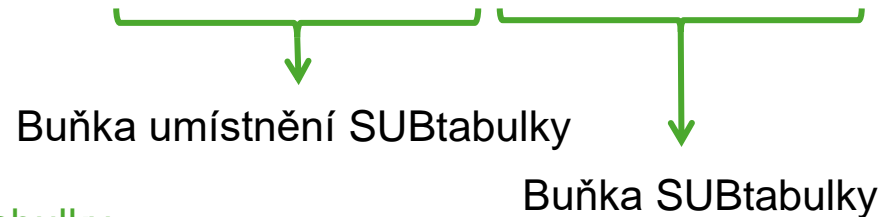
# Programování v SimTalk



## Tabulka

- Čtení ze SUBtabulky

- `print <název_tabulky>[<sloupec>,<řádek>][<sloupec>,<řádek>];`



- Obecný zápis do SUBtabulky

- `<název_tabulky>[<sloupec>,<řádek>][<sloupec>,<řádek>] := <hodnota>;`

- Změna vlastnosti prvku ze SUBtabulky pomocí metody

- `<Objekt>.<atribut> := <název_tabulky>[<sloupec>,<řádek>];`

- `print <název_tabulky>[<sloupec>,<řádek>];`

# Programování v SimTalk



## Tabulka

- $xDim$ ,  $yDim$ 
  - Vrací číslo posledního sloupce, řádku v tabulce, který obsahuje záznam. Číslo nezahrnuje nultý sloupec a řádek.
  - Hodnoty jsou datového typu `integer`.

	integer 1	integer 2	string 3
1	1	2	
2	2		
3	3		
4	4		
5	5		
6			
7			
8			
9			
10			

$xDim = 2$   
 $yDim = 5$

# Programování v SimTalk



## Tabulka

- **MaxxDim, MaxyDim**
  - Vrací **absolutní počet sloupců, řádků** v tabulce. Číslo nezahrnuje nultý sloupec a řádek.
  - Hodnoty jsou datového typu **integer**.

	integer 1	integer 2	string 3
1	1	2	
2	2		
3	3		
4	4		
5	5		
6			
7			
8			
9			
10			

MaxxDim = 3  
MaxyDim = 10



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)**