

Funkce a pravidla

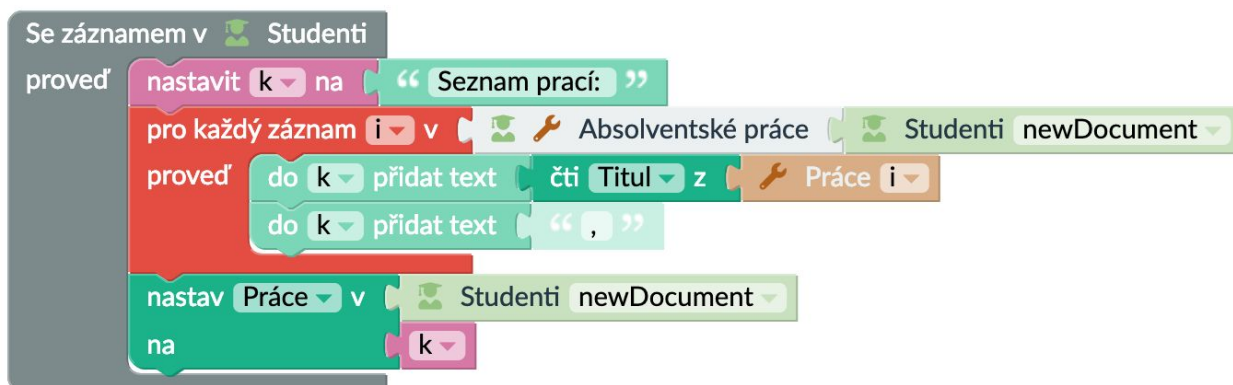
Poslední aktualizace 30. srpna 2019

ÚVOD

Funkce a pravidla jsou v Lumeerovi přizpůsobené tak, aby byly velmi snadné na použití. Nastavují se v grafické podobě, která je přirozená a umožňuje snadné pochopení smyslu té které funkce či pravidla. Funkce a pravidla používají shodné principy a jsou mezi nimi jen minimální rozdíly, na které vždy upozorníme. Zjednodušeně lze říci, že Pravidla dokáží totéž, co Funkce, a ještě něco navíc.

SKLÁDAČKA

Na Obrázku 1 můžeme vidět příklad poměrně komplexního pravidla. Jak si můžete všimnout, podobá se skládačce (puzzle).



Obrázek 1.

Každý jeden dílek skládačky se nazývá **Blok**. Bloky mají ouška (na svojí levé straně), šipky (na spodním okraji), kulaté otvory pro ouška (na pravo) a trojúhelníkové otvory (na horním okraji). Jako u běžné skládačky, žádné otvory ani ouška nejsou na okraji (na obvodu) skládačky. Celý obvod je krásně rovný.

TOK INFORMACÍ

Vytváření vlastních Funkcí a Pravidel je jednodušší, pokud porozumíte *toku informací* mezi jednotlivými Bloky.

Hodnoty (tedy obsah jednotlivých buněk v *Tabulkové Perspektivě*) jsou vždy sbírání na jednom řádku Bloků zprava doleva. Následující Blok poskytuje hodnotu (*Záznam*) na svojí levé straně

(proto má vlevo ouško) a musí být připojen k Bloku, který má na své pravé straně příslušný otvor (tedy k Bloku, který očekává vstup).



Obrázek 2.

Jak jsme již zmínili, otvor na pravé straně Bloku očekává připojení nějakého vstupu. Na následujícím příkladu můžete vidět Blok, který očekává seznam Hodnot, transformuje je na jejich součet a tento vrací na své levé straně.



Obrázek 3.

Ve Funkcích nejsou k dispozici žádné Bloky, které mají šipku nebo otvor na své horní nebo spodní straně. Proto je následující částí platná pouze pro Pravidla.

PŘÍKAZY

Tato část se vztahuje pouze k Pravidlům.

Existuje druhý typ toku a tím je postup v čase, neboli tok času, který v Pravidlech postupuje shora dolů. Jednotlivé řádky Bloků tvoří příkazy, které se provádějí jeden po druhém shora dolů.

To můžeme pozorovat například na Obrázku 1. Postup je následující:

1. Je proveden růžový Blok **nastavit k na**. Tento Blok nastaví proměnnou *k* na hodnotu Bloku, který je připojen na jeho pravé straně. V tomto případě se jedná o text "Seznam prací:". Více o proměnných naleznete dále v této kapitole.
2. Následně je spuštěn červený Blok **pro každý záznam i v**. Tento Blok přijímá na svém vstupu seznam *Záznamů* a spouští své vnitřní Bloky opakovaně pro každý *Záznam* ze vstupního seznamu zvlášť. Na každý *Záznam* je tedy aplikována následující posloupnost kroků:
 - a. Připoj Hodnotu Atributu *Titul* k proměnné *k*.
 - b. Připoj text " , " k proměnné *k*.
3. Nyní máme seznam všech titulů ze všech *Záznamů* v *Tabulce Práce*, které jsou připojené k hlavnímu *Záznamu* v *Tabulce Studenti*. Tento seznam jsme vytvořili v proměnné *k*. Následující Blok **nastav Práce v** očekává dva vstupy. Nejdříve *Záznam*, se kterým budeme pracovat, a dále *Hodnotu* (obsah proměnné *k*), kterou vložíme do *Atributu Záznamu*, který vybereme v rozbalovací nabídce (tedy *Atribut Práce*).

PROMĚNNÉ

Tato část se vztahuje pouze k pravidlům, protože ve Funkcích nejsou proměnné k dispozici.

Existují příkazy, které mohou číst a zapisovat proměnné. Proměnnou si můžete představit jako poznámku na papírku vedle vašeho počítače, kam si zapíšete poznámku - obsah proměnné. Každý

papírek představuje jednu proměnnou a každý papírek má svůj název/pojmenování, abychom dokázali papírky od sebe snadno odlišit. Proměnné tak zjednodušují zápis Pravidel, abychom nemuseli stejnou *Hodnotu* získávat opakovaně, pokaždé, když ji potřebujeme. Změna proměnné nemění původní *Hodnotu* v *Záznamu*.

OBÁLKA

I Funkce i Pravidlo potřebuje svůj obalovací Blok. Vlevo můžeme vidět obalovací Block pro Pravidlo, vpravo pak pro funkci.



Obrázek 4.

Všechny Bloky, které umístíte na plochu musí být ve výsledku spojené k obalovacímu Bloku (buď přímo, nebo přes jiné Bloky). Bloky, které zůstanou rozmístěné a nepřipojené, způsobí, že Funkce nebo Pravidlo nebudou fungovat.

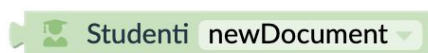
PRÁCE S ÚDAJI

POJMY

Abychom snadno porozuměli způsobu práce s údaji, je dobré se shodnout na společných pojmech a jejich významu. Pokud již neznáte pojmy jako *Tabulka*, *Záznam*, *Atribut*, *Hodnota* apod. doporučujeme vám si nejdříve přečíst kapitolu [Práce s údaji v příručce Základní informace](#).

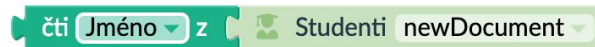
BLOKY SE ZÁZNAMY

Nejllepší způsob, jakým začít tvořit vaši Funkci nebo Pravidlo, je s Blokem, který představuje aktuální *Záznam* (také označován jako *Dokument*). V případě Pravidel máte k dispozici dva takové Bloky. Jeden pojmenovaný *newDocument* a druhý *oldDocument*. Tyto Bloky představují stav *Záznamu* před změnou a po změně, aby je bylo možné například porovnat a zjistit, jestli se sledovaný *Atribut* změnil. V případě Funkcí je k dispozici jen jeden Blok s názvem *thisDocument*. Pokud Pravidlo reaguje na vytvoření *Záznamu* je *oldDocument* prázdný, pokud reaguje na smazání *Záznamu*, je *newDocument* prázdný.



Obrázek 5.

Blok na Obrázku 5 představuje jeden celý *Záznam* (tj. řádek v *Tabulkové Perspektivě*). Tento *Záznam* je dostupný na levé straně Bloku. Abychom z Bloku zjistili *Hodnotu* konkrétního *Atributu*, musíme použít Blok na čtení *Atributů*.



Obrázek 6.

Nyní jsme získali jednu *Hodnotu* z *Atributu* "Jméno", která je nyní k dispozici na levé straně Bloku pro čtení. Zkusme něco jiného a podívejme se, jak můžeme získat připojené *Záznamy*. K tomuto účelu slouží Blok s *Vazbou*, který připojíme k Bloku se *Záznamem*.

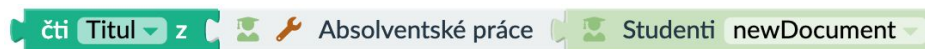


Obrázek 7.

Všimněme si nyní důležitého rozdílu mezi situacemi na Obrázku 6 a 7. Na levé straně šedého Bloku *Vazby* na Obrázku 7 se nyní nachází seznam *Záznamů*. Je to z toho důvodu, že každý *Záznam* může být připojen na libovolné množství dalších *Záznamů*. Všimněte si také, jak tok informací postupuje zprava doleva. Je to přesně opačný směr, než jak údaje vidíme v propojených tabulkách.

Tip: Jaký je důvod opačného směru? Pokud bychom chtěli v Blocích dodržet stejný směr toku informací jako v *Tabulkové Perspektivě*, museli bychom také otočit obalovací Bloky. Pokud by byly tyto obalovací Bloky obráceně, působily by velmi nepřirozeně a pochopení smyslu *Pravidel* a *Funkcí* by bylo mnohem náročnější. Funkce by také musely fungovat přesně opačně než v matematice (psali bychom $f(y)=x$ namísto $y=f(x)$).

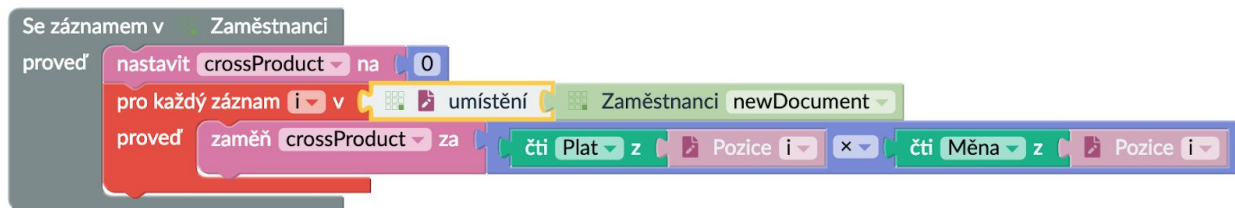
Skutečnost, že Blok *Vazby* získává více *Záznamů* ovlivňuje fungování další navázaných Bloků. Připojme Blok pro čtení *Atributu*.



Obrázek 8.

Protože Blok pro čtení začíná se seznamem *Záznamů*, vrací také seznam, avšak seznam *Hodnot* vybraného *Atributu* ("Titul") ze všech *Záznamů*. Můžeme nyní použít různé Bloky pro spojení *Hodnot* do jednoho dlouhého textu, nebo pro výpočet sumy apod.

Je zde však jeden nedostatek. Pokud bychom použili tento přístup pro získání několika seznamů *Hodnot*, nedokážeme zjistit, které dvě *Hodnoty* z jednoho a druhého seznamu pocházejí původně z jednoho *Záznamu* (kdybychom například násobili počet kusů zboží s jeho cenou). Pro tento účel existuje Blok pro procházení seznamů *Záznamů*. Uvnitř tohoto Bloku můžeme přistupovat ke všem *Atributům* jednoho *Záznamu* pěkně popořadě (viz Obrázek 9).



Obrázek 9.

Tato situace je patrná na Obrázku 9. Pravidlo nejdříve nastaví nulu do proměnné `crossProduct`. Poté prochází všechny Záznamy připojené k Záznamu z Tabulky "Zaměstnanci" pomocí Vazby se jménem "umístění". Bloky uvnitř Bloku pro procházení seznamu (**pro každý záznam...**) se provádí pro každý připojený Záznam. Bloky přičtou (přidají) výsledek násobení Atributů "Plat" a "Měna" do proměnné `crossProduct`. Proměnná `i` představuje jeden aktuální Záznam ze seznamu Záznamů, takže když čteme dva různé Atributy této proměnné, vždy jsou ze stejného Záznamu.

Na posledním Obrázku (Obrázek 10) je vizuálně znázorněn význam jednotlivých Bloků a toho, jak pracují s údaji.

Záznam
Tento Blok představuje Záznam (řádek) z Tabulky Projects.

Čtení Hodnot
Tento Blok čte Hodnotu zvoleného Atributu (? = nenastaveno).

Jméno/Typ Vazby
Šedé Bloky s ikonami dvou tabulek a Jménem Vazby zjistí všechny Záznamy připojené k Záznamu na pravé straně Bloku.

Číst a Zapsat Hodnotu Atributu
Dva propojené Bloky výše zjistí konkrétní Hodnotu (označená žlutě).
Tři Bloky níže nastaví Hodnotu Atributu "Name" v Záznamu z Tabulky "Projects" na řetězec "Hello".

Procházení připojených Záznamů
Zde si můžete všimnout obráceného toku informací. Začínáme Záznamem z Tabulky "Projects", načteme všechny Záznamy připojené Vazbou se Jménem "Project assignments" a spustí Blok příkazů (na obrázku ještě prázdný) s každým navázaným Záznamem zvlášť. Proměnná `i` tedy představuje vždy jeden Záznam z Tabulky Employees.

Projects				Employees					
Name	Description	Started	State	Code	First Name	Last Name	Department	JobRole	Ed
1	Resistant Chic	Nutritionally V	Completed	CHPEA	Salvador	Godinez	Research & D	Healthcare Ex	M
2					Marta	Cagley	Research & D	Research Dire	Lif
3	Micronutrient	A Mini Review	Outdated	MMOLY	Laurena	Drayer	Research & D	Laboratory Te	M
4					Iluminada	Pardee	Research & D	Laboratory Te	Lif
5	Synapsin III	Synapsin III in	In Progress	SYNIII					
6	Papillary Thyr	The Role of C	Proposed	PTCAN	Wendolyn	Carruthers	Research & D	Laboratory Te	M
7	Integron isolat	Study of Class	In Progress	NTGR	Iluminada	Pardee	Research & D	Laboratory Te	Lif
8					Alexis	Gauntt	Research & D	Manufacturin	Lif

Workflow blocks: `čti ? z`, `čti Name z`, `pro každý záznam i v`, `proved`

Obrázek 10.

