PavlOS Dokumentace pro vývojáře

Obsah

[Začátek 1](#_Toc57291592)

[Design 2](#_Toc57291593)

[Přehled 2](#_Toc57291594)

[Ukázka designování v JSONu 3](#_Toc57291595)

[Podrobný popis objektů ve stránce 5](#_Toc57291596)

[Hodnoty 5](#_Toc57291597)

[Window 5](#_Toc57291598)

[Text 6](#_Toc57291599)

[Button 6](#_Toc57291600)

[Text box 6](#_Toc57291601)

[Border 7](#_Toc57291602)

[Check box 7](#_Toc57291603)

[Text box multiline 7](#_Toc57291604)

[Web browser 8](#_Toc57291605)

[Styly 8](#_Toc57291606)

[Vytvoření konfiguračního souboru pro styly 8](#_Toc57291607)

[příkazy v tlačítcích 8](#_Toc57291608)

[Programování 9](#_Toc57291609)

[Přehled 9](#_Toc57291610)

[Typy proměnných 9](#_Toc57291611)

[Zápis proměnných 10](#_Toc57291612)

[Využití proměnných 10](#_Toc57291613)

[Knihovny 10](#_Toc57291614)

[Nastavení hodnot proměnných 10](#_Toc57291615)

[Opakování (for loop) 12](#_Toc57291616)

[Práce se systémem 12](#_Toc57291617)

[Print do logu 13](#_Toc57291618)

[Uložení proměnné 13](#_Toc57291619)

[Podmínka (if) 13](#_Toc57291620)

# Začátek

Pro vytváření aplikací v PavlOS budete muset mít znalosti **JSONu** a našeho jazyka **Pavl+**. Pokud nevíte, jak se v Pavl+ programuje, tak nezoufejte, my Vás to naučíme.

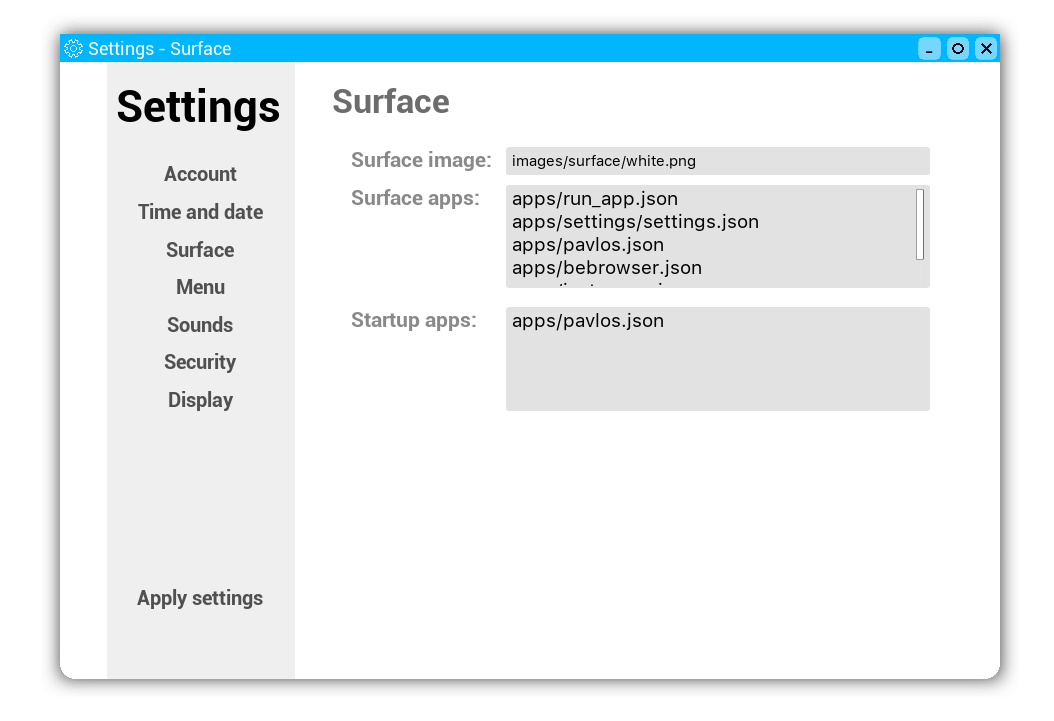
Budete potřebovat nějaký textový editor. My doporučujeme **Notepad** nebo **Visual Studio Code**. Možná už máme v této době náš vlastní editor nebo compiler pro jazyk Pavl+ a JSON design, tak můžete využít jej.

# Design

## Přehled

Design se vytváří v datových souborech JSON. Pokud jste již vytvářeli programy ve Win UI v XAML, tak náš systém v JSONu je velice podobný. V další části této kapitoly si ukážeme, jak vypadá vytváření designu aplikace „PavlOS Introduction“. Také se můžete podívat do souborů aplikací a podívat se na jejich strukturu, ať uvidíte použítí jiných objektů pro rychlejší pochopení.  
  


Aplikace PavlOS Introduction



Aplikace Settings

Naše designování v JSONu se skládá z pár sekcí.

První objekt neboli „sekce“ znamená číslo strany. Aplikace může mít nekonečný počet stran. Ve stranách jsou potom různé objekty, které tvoří samotnou strukturu stránky, o kterých se dozvíte níže.

**POZOR!** 1. strana musí mít vždy název „page\_1“!

## Ukázka designování v JSONu

Barvy označují objekty v kódu níže.

Základní objekt, ve kterém se nachází celá aplikace.

Strana aplikace.

Slouží pro design nebo funkci stránky. Je několik typů (postupně přibydou další).

* **Window** – Slouží k definování velikosti, názvu a dalších věcí pro okno aplikace.
* **Text** – Zobrazí text na určitém definovaném místě.
* **Button** – Stisknutí tlačítka může udělat nějaké akce.
* **Text box** – Textové pole.
* **Border** – Blok s jednou barvou (může být nastaven také obrázek)
* **Check box** – Box pro zaškrtnutí.
* **Text box multiline** – Textové pole s více řádky.
* **Web browser** – Webový prohlížeč.

Podrobný popis těchto parametrů se dozvíte v další části.

{ “pages”: [  
  
 { ”page\_1”: [  
  
 { ”window”:  
 { ”title”: ”Welcome to PavlOS!”, ”app\_name”: ”PavlOS Introduction”, ”size\_x”: 850, ”size\_y”: 425, ”icon”: ”images/pavlos\_logo\_reworked\_transparent.png” }  
 },  
  
 { ”text”: [  
 { ”string”: ”Welcome to PavlOS!”, ”pos\_x”: 170, ”pos\_y”: 70, ”scale\_x”:true, ”scale\_y”:true, ”font\_size”: 40, ”color”: ”R=1, G=1, B=1, A=1” },  
 { ”string”: ”We’re glad that you’ve downloaded the latest version of PavlOS!”, ”pos\_x”: 30, ”pos\_y”: 170, ”scale\_x”:true, ”scale\_y”:true, ”font\_size”: 20, ”color”: ”R=0.15, G=0.15, B=0.15, A=1” }  
 ]  
 },  
  
 { ”button”: [  
 { ”string”: ”NEXT”, ”pos\_x”: 0, ”pos\_y”: 310, ”size\_x”: 850, ”size\_y”: 40, ”font\_size”: 20, ”resize\_x”:true, ”scale\_y”:true, ”text\_color”: ”R=0.08. G=0.08, B=0.08, A=1”, ”color”: ”R=0.7, G=0.7, B=0.7”, ”style”: ”styles/button/modern\_style\_dark.ini”, ”command”: [{”cmd”: ”open\_page: pages[1].what\_is”} ] }  
 ]  
 },  
  
 { ”text\_box”: [  
  
 ]  
 },  
  
 { ”border”: [  
 { ”pos\_x”: 0, ”pos\_y”: 50, ”size\_x”: 850, ”size\_y”: 100, ”zorder”: -1, ”resize\_x”:true, ”scale\_y”:true, ”color”: ”R=0.2, G=0.2, B=0.2, A=1” },  
 { ”pos\_x”: 35, ”pos\_y”: 50, ”size\_x”: 100, ”size\_y”: 100, ”zorder”: 1, ”scale\_x”:true, ”scale\_y”:true, ”color”: ”R=1, G=1, B=1, A=1”, ”img”: ”images/PavlOS\_logo\_transparent” },  
 ]  
 },  
 { ”check\_box”: [  
  
 ]  
 },  
  
 { ”text\_box\_multiline”: [  
  
 ]  
 },  
  
 { ”web\_browser”: [  
  
 ]  
 },  
 ]  
},  
  
 { ”what\_is”: [  
  
 { ”window”:  
 { ”title”: ”Welcome to PavlOS! – What is Pavl?”, ”app\_name”: ”PavlOS Introduction”, ”size\_x”: 850, ”size\_y”: 425, ”icon”: ”images/pavlos\_logo\_reworked\_transparent.png” }  
 },  
  
. . . . . . . . Pokračování kódu s dalšími objekty . . . . . . . .  
  
 ]  
},  
 ]  
}

## Podrobný popis objektů ve stránce

Zde se dozvíte co můžou objekty obsahovat. Např. Window obsahuje title, app\_name atd.

### Hodnoty

* Růžová – String
* Zelená – Float
* Světle modrá – Integer
* Červená – Boolean
* Tmavě modrá – Vektor (např. R=1, G=1, B=1, A=1)

### Window

* title – Nadpis okna.
* app\_name – Název aplikace.
* icon – Ikona aplikace (musí se vepsat cesta k ikoně)
* size\_x – Velikost okna na ose X.
* size\_y – Velikost okna na ose Y.
* min\_size\_x – Nejmenší možná velikost okna na ose X.
* min\_size\_y – Nejmenší možná velikost okna na ose Y.
* allow\_nav\_btn – Povolení navigačních tlačítek.
* bg\_color – Nastaví barvu pozadí.

### Text

* string – Text (možnost použít „\n“ pro funkci SHIFT+ENTER – přeskočení na nový řádek).
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* font\_size – Velikost fontu.
* scale\_x – Mění pozici podle velikost okna na ose X.
* scale\_y – Mění pozici podle velikost okna na ose Y.
* color – Nastaví barvu textu.

### Button

* string – Text v tlačítku.
* style – Cesta k style config souboru (např. „styles/button/modern\_style\_dark.ini“). Více v kapitole „Styly“.
* command – Array s objekty s příkazy. Více v kapitole „Příkazy v tlačítcích“.
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* size\_x – Velikost na ose X.
* size\_y – Velikost na ose Y.
* font\_size – Velikost fontu.
* scale\_x – Mění pozici podle velikost okna na ose X.
* scale\_y – Mění pozici podle velikost okna na ose Y.
* resize\_x – Mění velikost podle velikosti okna na ose X.
* resize\_y – Mění velikost podle velikosti okna na ose Y.
* text\_color – Nastaví barvu textu.
* color – Nastaví barvu tlačítka.

### Text box

* text – Text v textovém poli.
* text\_ID – ID pro propojení s Pavl+ a příkazy (např. získání textu z text boxu).
* hint – Nápověda v textovém poli (pokud bude textové pole prázdné, tak se zobrazí nápověda.
* config\_file – Cesta k konfiguračnímu souboru, který chceme editovat text boxem (při použití této hodnoty musí být config\_type s hodnotou **true**).
* section – Sekce, kterou chceme editovat v konfiguračním souboru (při použití této hodnoty musí být config\_type s hodnotou **true**).
* key – Klíč k datu v určené sekci (při použití této hodnoty musí být config\_type s hodnotou **true**).
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* size\_x – Velikost na ose X.
* size\_y – Velikost na ose Y.
* font\_size – Velikost fontu.
* scale\_x – Mění pozici podle velikost okna na ose X.
* scale\_y – Mění pozici podle velikost okna na ose Y.
* resize\_x – Mění velikost podle velikosti okna na ose X.
* resize\_y – Mění velikost podle velikosti okna na ose Y.
* config\_type – Jestli se bude text box využívat pro nastavení konfiguračního souboru .ini.

### Border

* img – Pokud se vloží cesta k obrázku, tak se zobrazí obrázek místo borderu.
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* size\_x – Velikost na ose X.
* size\_y – Velikost na ose Y.
* zorder – Posunutí do popředí nebo do pozadí pomocí čísel vrstvy (-1 dozadu, 1 dopředu – můžete použít jakákoliv čísla).
* scale\_x – Mění pozici podle velikost okna na ose X.
* scale\_y – Mění pozici podle velikost okna na ose Y.
* resize\_x – Mění velikost podle velikosti okna na ose X.
* resize\_y – Mění velikost podle velikosti okna na ose Y.
* color – Nastaví barvu borderu/obrázku.

### Check box

* text – Text v check boxu.
* config\_file – Cesta k konfiguračnímu souboru, který chceme editovat text boxem.
* section – Sekce, kterou chceme editovat v konfiguračním souboru.
* key – Klíč k datu v určené sekci.
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* size\_x – Velikost na ose X.
* size\_y – Velikost na ose Y.
* zorder – Posunutí do popředí nebo do pozadí pomocí čísel vrstvy (-1 dozadu, 1 dopředu – můžete použít jakákoliv čísla).
* text\_color – Nastaví barvu textu.
* color – Nastaví barvu check boxu.

### Text box multiline

* text – Text v textovém poli.
* text\_ID – ID pro propojení s Pavl+ a příkazy (např. získání textu z text boxu).
* hint – Nápověda v textovém poli (pokud bude textové pole prázdné, tak se zobrazí nápověda.
* config\_file – Cesta k konfiguračnímu souboru, který chceme editovat text boxem (při použití této hodnoty musí být config\_type s hodnotou **true**).
* section – Sekce, kterou chceme editovat v konfiguračním souboru (při použití této hodnoty musí být config\_type s hodnotou **true**).
* key – Klíč k datu v určené sekci (při použití této hodnoty musí být config\_type s hodnotou **true**).
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* size\_x – Velikost na ose X.
* size\_y – Velikost na ose Y.
* font\_size – Velikost fontu.
* scale\_x – Mění pozici podle velikost okna na ose X.
* scale\_y – Mění pozici podle velikost okna na ose Y.
* resize\_x – Mění velikost podle velikosti okna na ose X.
* resize\_y – Mění velikost podle velikosti okna na ose Y.
* config\_type – Jestli se bude text box využívat pro nastavení konfiguračního souboru .ini.

### Web browser

* url – URL adresa webu, který chcete načíst (Lze načíst z text boxu pomocí jeho text\_ID – get.**vložte\_jeho\_text\_ID**).
* pos\_x – Pozice na ose X.
* pos\_y – Pozice na ose Y.
* size\_x – Velikost na ose X.
* size\_y – Velikost na ose Y.
* zorder – Posunutí do popředí nebo do pozadí pomocí čísel vrstvy (-1 dozadu, 1 dopředu – můžete použít jakákoliv čísla).
* scale\_x – Mění pozici podle velikost okna na ose X.
* scale\_y – Mění pozici podle velikost okna na ose Y.
* resize\_x – Mění velikost podle velikosti okna na ose X.
* resize\_y – Mění velikost podle velikosti okna na ose Y.

## Styly

Styly se využívají pro vlastní nadesignování (zatím tlačítek) objektů – obrázky při stisknutí atd. Styly se píšou v konfiguračním souboru „.ini“ a cesta k tomuto souboru se píše do objektu jako „style“.

### Vytvoření konfiguračního souboru pro styly

Konfigurační soubor se (zde moderním) stylem vypadá takto.

[normal] (stav objektu)  
color=(R=0,G=0,B=0,A=0.075) (barva)  
img=images/styles/white.png (cesta k souboru s obrázkem)  
[hovered]  
color=(R=0,G=0,B=0,A=0.1)  
img=images/styles/white.png  
[pressed]  
color=(R=0,G=0,B=0,A=0.05)  
img=images/styles/white.png

Tlačítko před a po použití stylu

## příkazy v tlačítcích

Tlačítka jsou programovatelná podle vaší libosti. Existuje pár příkazů, které můžete použít. Každý z těchto příkazů se píše do „cmd“ arraye v objektu „button“ v JSON souboru.

run\_app: (aplikace v souboru .json s cestou k souboru – např. apps/test.json nebo text\_ID z textového pole pro získání textu)  
run\_app: test.json

close\_app (zavře aplikaci)

min\_app (minimalizuje aplikaci do spodního panelu)

open\_page: (název stránky aplikace, kterou chcete otevřít)  
open\_page: page\_1

notification: (text upozornění)  
notification: Upozornění funguje!

shutdown\_OS (vypne operační systém)

restart\_OS (restartuje operační systém)

refresh\_settings (obnoví nastavení systému – např. po změně nastavení systému – využívá se v tlačítku „apply settings“ v aplikaci „settings“)

play\_sound: (pustí zvuk ze souboru .wav)  
play\_sound: sounds/system/shutdown.wav

trust\_app= (název aplikace nebo „last app“ – systém bude této aplikaci věřit – pokud aplikace obsahuje neobvyklé požadavky, tak váš to už nebude systém upozorňovat – neobvyklé požadavky jsou např. restart\_OS, refresh\_settings)  
trust\_app=last\_app (bude věřit posledně otevřené aplikaci)  
trust\_app=apps/test.json (bude věřit určené aplikaci)

@ (pokud příkaz obsahuje jako první znak „@“, tak se automaticky použije compiler pro pavl+ kód – více v kapitole o programování)  
@set=number1=5+number2  
@int=number3=546

# Programování

## Přehled

Programuje se v našem jazyce Pavl+. Jazyk je jednoduchý na pochopení a má jednoduchou strukturu. V dalších verzích bude přibývat více funkcí, tak nám můžete napsat, jakou bychom mohli přidat. Jako v každém jazyce jsou zde proměnné, logické funkce atd.

Každá funkce se píše do jednoho řádku – kód se spouští řádek po řádku.

Díky programování můžete vytvořit jakékoliv aplikace a máte více prostoru pro kreativitu – např. oznámení může mít text z proměnné nebo máte jednodušší propojování kódu s JSON objekty.

## Typy proměnných

Jsou zde 4 typy proměnných – integer, float, string, boolean.

### Zápis proměnných

Proměnné se zapisují podobně jako v ostatních jazycích.

TYP PROMĚNNÉ=NÁZEV PROMĚNNÉ=HODNOTA PROMĚNNÉ

zkrácený zápis:

Int=Number1=54  
Flt=Number2=8.56  
Str=Text1=Hello  
Bool=Bool1=True

normální zápis:

Integer=Number1=54  
Float=Number2=8.56  
String=Text1=Hello  
Boolean=Bool1=True

### Využití proměnných

Proměnné se používají pro ukládání dat, které si aplikace potřebuje zapamatovat nebo s nimi nějak pracovat.

## Knihovny

### Nastavení hodnot proměnných

Je mnoho možností jak nastavit hodnotu proměnné. Matematika funguje velmi podobně jako např. u jazyka C. Zatím je k dispozici pár matematických úloh, později přibydou další (zatím nelze např. zároveň sčítat a násobit)

#### Nastavení hodnoty

SET=NÁZEV PROMĚNNÉ=data (data se vloží do proměnné)

Set=string1=Ahoj kámo! (VLOŽENÍ TEXTU DO STRINGU)

Set=Number1=1568 (VLOŽENÍ ČÍSLA DO INTEGERU)

Set=Number1=Number2 (VLOŽENÍ DATA DO PROMĚNNÉ Z JINÉ PROMĚNNÉ)

#### ZÍSKÁNÍ HODNOTY Z JSON OBJEKTU

GET=název proměnné (data z objektu se vloží do této proměnné)=TYP OBJEKTU=ID OBJEKTU (id objektu se nastaví v objektu v json souboru)

Get=string1=text\_box=txtbox1 (VLOŽENÍ TEXTU DO STRINGU)

Get=number1=text\_box=txtbox1 (VLOŽENÍ čísla do integeru/floatu)

Get=bool1=check\_box=chbox1 (vloží boolean do booleanu)

#### složený string z více proměnných či textů

append=název proměnné(proměnné a texty se vloží této proměnné)=proměnné a texty (text je v uvozovkách ‘text‘)

Append=string1=‘Výsledek: ‘+number1 (string v uvozovkách ‘text‘ a proměnná s číslem – výsledek: 48)

Append=string1=string2+number1+string3 (string s číslem a dalším stringem)

#### Sčítání

SET=NÁZEV PROMĚNNÉ(DO KTERÉ vložíme VÝSLEDEK)=sčítání

Set=Number1=10+5 (sčítání čísel a vložení výsledku do proměnné)

Set=Number1=Number1+2 (sčítání proměnné s číslem a vložení výsledku do proměnné)

Set=Number1=Number1+Number2 (sčítání dvou proměnných a vložení výsledku do proměnné)

#### Odčítání

SET=NÁZEV PROMĚNNÉ (DO KTERÉ vložíme VÝSLEDEK)=ODČÍTÁNÍ

Set=Number1=10-5 (ODČÍTÁNÍ čísel a vložení výsledku do proměnné)

Set=Number1=Number1-2 (ODČÍTÁNÍ proměnné s číslem a vložení výsledku do proměnné)

Set=Number1=Number1-Number2 (ODČÍTÁNÍ dvou proměnných a vložení výsledku do proměnné)

#### Násobení

SET=NÁZEV PROMĚNNÉ (DO KTERÉ vložíme VÝSLEDEK)=NÁSOBENÍ

Set=Number1=10\*5 (NÁSOBENÍ čísel a vložení výsledku do proměnné)

Set=Number1=Number1\*2 (NÁSOBENÍ proměnné s číslem a vložení výsledku do proměnné)

Set=Number1=Number1\*Number2 (NÁSOBENÍ dvou proměnných a vložení výsledku do proměnné)

#### Dělení

SET=NÁZEV PROMĚNNÉ (DO KTERÉ vložíme VÝSLEDEK)=dělení

Set=Number1=10-5 (dělení čísel a vložení výsledku do proměnné)

Set=Number1=Number1-2 (dělení proměnné s číslem a vložení výsledku do proměnné)

Set=Number1=Number1-Number2 (dělení dvou proměnných a vložení výsledku do proměnné)

### Opakování (for loop)

For loop se používá pro opakování určité funkce několikrát po sobě (určíte, kolikrát se bude opakovat).

for=kolikrát se bude opakovat (muze být i proměnná)=@kód, který se bude vykonávat („@“ vždy znamená – spustit kód)

For=7=@set=number1=number1+1 (7x se zopakuje příkaz)

For=number2=@append=number1+‘Ahoj‘ (příkaz se opakuje tolikrát, podle hodnoty proměnné „number2“)

### Práce se systémem

V systému se dá udělat nebo změnit hodně věcí, jako např. restartovat nebo vypnout systém, měnit datum a čas…

Sys=shutdown (vypne systém)

Sys=restart (restartuje systém)

Sys=close\_apps (zavře všechny aplikace)

Sys=min\_apps (minimalizuje všechny aplikace)

Sys=notification=‘Nadpis‘ (Může být proměnná – bez uvozovek)=‘Text‘ (Může být proměnná – bez uvozovek) (vytvoří ihned oznámení)

Sys=time\_and\_date=12 (nastaví 12 hodinový formát s mm/dd/yyyy)

Sys=time\_and\_date=24 (nastaví 24 hodinový formát s dd/mm/yyyy)

### Print do logu

Vepíše určený text do systémového logu (ještě není dokončen).

print=text nebo proměnná

Print=Tohle funguje! (printne větu „tohle funguje!“ do logu)

Print=number1 (printne do logu data z proměnné „number1“)

### Uložení proměnné

Uloží proměnnou na disk, aby se mohla později využít (např. při příštím použití aplikace).

SAVE=NÁZEV PROMĚNNÉ (KTEROU CHCEME ULOŽIT)

Save=number (Uloží proměnnou „number“)

Save=MyVar (uloží proměnnou „Myvar“)

### Podmínka (if)

Určí jestli se hodnoty rovnají nebo jsou větší nebo menší. V tomto příkazu se nedá využít @ v @.

if=‘text‘ nebo proměnná==(🡨 určování rovnosti)‘text‘ nebo proměnná=@pokud je to pravda, vykoná příkaz@pokud to není pravda, vykoná tento příkaz (else)

If=number1==‘5‘=@sys=notification=‘Je to pravda‘@sys=notification=‘Není to pravda‘ (pokud „number1“ se rovná „5“, tak se vypíše „Je to pravda“ a pokud se nerovná, tak se vypíše „Není to pravda“)

If=number1<=‘5‘=@sys=notification=‘Je to pravda‘@sys=notification=‘Není to pravda‘ (pokud „number1“ se rovná nebo je měnší než „5“, tak se vypíše „Je to pravda“ a pokud se nerovná, tak se vypíše „Není to pravda“)

If=number1>=‘5‘=@sys=notification=‘Je to pravda‘@sys=notification=‘Není to pravda‘ (pokud „number1“ se rovná nebo je větší neř „5“, tak se vypíše „je to pravda“ a pokud se nerovná, tak se vypíše „není to pravda“)

If=number1<‘5‘=@sys=notification=‘Je to pravda‘@sys=notification=‘Není to pravda‘ (pokud „number1“ je menší než 5, tak se vypíše „je to pravda“ a pokud se nerovná, tak se vypíše „není to pravda“)

If=number1>‘5‘=@sys=notification=‘Je to pravda‘@sys=notification=‘Není to pravda‘ (pokud „number1“ je větší než 5, tak se vypíše „je to pravda“ a pokud se nerovná, tak se vypíše „není to pravda“)

### Získání dat z json design objektů

Můžete získat např. text z text boxu a vložit jej do proměnné. Může se vložit kdekoliv a kód jej přemění na získanou hodnotu z JSON objetu.

&get.id objektu.název hodnoty, kterou chceme získat&

String=string1=**&get.txtbox1.text&** (Do stringu se načte text z textboxu)

If=**&get.txtbox1.text&**==‘ahoj‘=@sys=notification=‘Text box obsahuje „ahoj“.‘ (zjistí se, jestli je text z text boxu „ahoj“)