HeinzoScript MANUAL

A guide how to use HeinzoScript and HeinzoScript reader.

CONTENTS

1. WHAT IS HEINZOSCRIPT?

HeinzoScript is my own idea how to create a file that contains text and images without using Word, or any other text editor. All you really need is to be able to create .txt file.

If you ask: Why did you make a new text editing software instead of using literally anything that exists and is free or/and opensource? Well, my only answer is: I made it coz I just wanted to. HeinzoScript will be supported in some of my future projects.

Of course after writing HeinzoScript text you need to generate it somehow and this is the job of HeinzoScript reader which is a bunch of if-else blocks packed into a .exe file. Really you will cry while reading the sourcecode.

2. HOW TO USE HEINZOSCRIPT?

There are few important files to be aware of while using HeinzoScript:

2.1. command4HSreader.txt

This file needs to be located in the same folder as HeinzoScript reader. It contains three lines of text, for example:

```
HS_init_path = input/hamster.HS.txt

HS_output_path = output/hamster.HS

HS_output_ext = .pdf
```

The first line tells HS reader where the file of initializer is located.

Both relative and absolute path can be used, so instead of "input/hamster.HS.txt", one can use "C:/Users/Jerry/HeinzoScript_reader/input/hamster.HS.txt". But it is the best to use "/" instead of Windows' "\" as code may have troubles with with the later one.

The second line tells HS reader where to put generated file.

Third line tells what should be the extension of file to generate. Supported extensions are: .pdf, .jpg (or .jpeg) and .png. Should the output file contains more than one page, one raster file per each page will be generated. The resolution of raster files is set to 600 PPI.

2.2. Initializer

HS needs an initializer file that contains parameters of output, for example:

```
file_folder = hamster.HS

filename = hamster.txt

width = 200

rotate = 0

font_size = 18

numbering = 2
```

```
numbering_font_size = 18

page_number = 1
```

The first line contains name of our main HS folder and second one tells the name of .txt file which will be generated.

Third line tells the width of page working area. Most of text editing softwares use margins for this purpose, but in HS I took opposite way. Working page is A4 format. Width is stated in milimetres and maximum value is 200 mm for page in portrait mode and 287 for page in landscape mode (it leaves 5 mm margin of each side of A4 page).

Fourth line, called „rotate" tells if the working page is in portrait or landscape mode. For portrait set 0, for landscape set 1.

Next line sets default font size measured in points (1/72nd of an inch). It is not permament decision, for font size can be set to each line later. Note that in HS there are only two fonts: Inconsolata Condensed as main one and Inconsolata-LGC as secondary one.

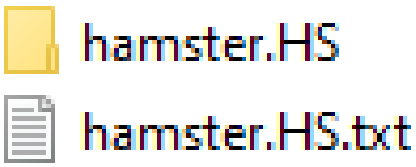
The next one starts with „numbering = " and it tells HS reader how we want to numer lines in our document. If we set 0, no line will be numbered, therefore any numbering must be included in text itself. If we set 1, a margin will be added at the left side of our page and all of the lines will be numbered in auto mode. Yet we can always tell HS reader which page we do not want to number. If we set 2, a margin will be added same as before and lines will be numbered in manual mode. This means we must tell HS reader which lines we actually want to be numbered.

Next line sets font size for numbers of lines. This one cannot be changed manually later in .txt file.

The last line tells if we want our pages to be counted. Should you set 0, no page number will be attached. In case of setting 1, a margin will be added at the bottom of page and each page number will contain two informations: current page no. and number of all pages.

2.3. Main HS folder

Main HS folder must be located in the same directory as initializer file.



Example of location of main HS folder and initializer file

While initializer file tells HS reader basic parameters of output file, the folder contains all files that are later combined to create output file.

2.4. Text file

Text file must be located inside main HS folder. It is the actual file which will be used as a canva to generate output file.

Text file uses few codes to tell HS reader how to deal with lines of text:

`<<C>` Place text in centre of line.

`<<R>` Place text at right side of the line.

If no of the above is used, or both are used at the same time, the text will be placed at the left side of the line.

`<<N>` This code has two meanings. If numbering is set to 1, it tells that this line should not be numbered. If numbering is set to 2, it tells that this line should be numbered.

`<<n>` Resets line number counting back to 1.

`<<I>` Tells that this line contains an image. This line is never numbered.

`<<c>` Tells that this line contains caption to an image. This line is never numbered, but the caption itself is numbered in following way: „fig. no. yourtexthere”. Remember that only caption lines are being numbered, not the image lines.

`<<U>` The next line will be placed at the same position as the current one. It is important to know that these two lines will be generated on top of each other, so

if you position the text incorrectly, they will blend together. This line is never numbered.

`<<G>` The font of this line will be changed from default „Inconsolata condensed" to „Inconsolata-LGC". While I prefer the first one, unfortunately it does only support latin alphabet. So if your text includes Greek or Russian, this code must be used. Note that it does not affect font of line number.

Codes can be mixed and their order is irrelevant. For example: `<<NCU>` places text in centre, tells HS reader to number (or not) this line and places the following line on top of this one.

Codes also may contain a number. In this case, number should be preceded by „-" sign. Number in code has two meanings. If it is used in line of text, it sets manually font size in points for this particular line. If it is used in line of image, it sets width of an image in milimetres. One must know, that image cannot be set to any size. If width or hight of the image is greater than working width or hight of page (including size of margins), it will be scaled down.

`<<-25>` Set current line font size to 25.

`<<R-30>` Place text at right side of the line and set its font size to 30.

`<<IC-80>` This line contains an centered image with a width of 80 mm.

Remember that numbers must be place at right side within line code.

The text can be also modified in terms of marking. Following marking signs are used:

`**Bold**`	`--Underline--`	`__Italic__`
Bold	<u>Underline</u>	*Italic*

Marking signs can be joined together like: `**__--BoldItalicUnderline--__**` will give you: ***<u>BoldItalicUnderline</u>***. Note that default font „Inconsolata Condensed" does not support italic forms. If you want to use italic text, switch to `<<G>` code.

If you want to include marking signs in your text, but do not want them to modify it, you must use „\"sign before marking signs. Like this: `\**Bold\**` or in case of multiple

signs: `\**\__BoldItalic\__\**`. So you can make for example bold text with two minuses around: `**\--Bold\--**` which looks like: **`--Bold--`**.

3. HEINZOSCRIPT IN ACTION!

As a school homework we have to write essay about animal chosen by us. Let's say we want to describe hamster. Let's start with creating empty folder named „hamster.HS" and empty text file named „hamster.HS.txt". We must first describe parameters for output file. We know that our page must be in portrait mode and font size to be 18 points. Quite big, but our essay is short, as we are lazy. We don't want to number each line, but we also want our pages to be numbered at some point. Now we must describe path to our initializer and our output file in „command4HSreader.txt".

Now we can create „hamster.txt" file inside our folder. And let the writing begin!

At first we must write our name and date. We don't want to waste two lines on that, so we write it in a single line:

```
Jerry Bonion                    30.01.2025
```

But wait! There is more elegant solution.

```
<<U>Jerry Bonion
<<R>30.01.2025
```

This will put both lines in a single line, but the lower line will be placed at right side of page. Now we want to include some sort of heading.

```
<<U>Jerry Bonion
<<R>30.01.2025
<<C-20>Essay: My favourite animal.
```

Now let's generate what we have so far.

Jerry Bonion 30.01.2025

Essay: My favourite animal.

So far so good. Now we must put some informations about our animal. Let's start with species. For latin species name it is correct to use italic. So we can write our line as:

```
<<G>Species: Winter White Dwarf Hamster (__Phodopus sungorus_)
```

But let’s use one trick this time.

```
<<U>Species: Winter White Dwarf Hamster
<<G>(__Phodopus sungorus_)
```

This text will be generated as:

Species: Winter White Dwarf Hamster (*Phodopus sungorus*)

Now we could just play with space to achieve:

```
<<U>Species: Winter White Dwarf Hamster
<<G>                                (__Phodopus sungorus__)
```

Species: Winter White Dwarf Hamster (*Phodopus sungorus*)

This approach is probably overengineered, but maybe there are some situations where it is just right.

Now let us attach an image of a hamster. We must put our image in hamster.HS folder (same that contains our hamster.txt file) and then we must attach it to our line. Then we can add a caption.

So far our text look like this:

```
<<U>Jerry Bonion
<<R>30.01.2025
<<C-20>**Essay: My favourite animal.**

<<U>Species: Winter White Dwarf Hamster
<<G>                                (__Phodopus sungorus__)

<<IC-80>hamster.png
<<Cc>My pet hamster
```

Jerry Bonion 30.01.2025

Essay: My favourite animal.

Species: Winter White Dwarf Hamster (*Phodopus sungorus*)

fig. 1: My pet hamster

Note that image is auto-numbered by code. Or more correct is to say, that the caption is being auto-numbered. If we included two images, one after another and only one caption, it would still be counted as „fig. 1”. Now we include some text.

Jerry Bonion 30.01.2025

Essay: My favourite animal.

Species: Winter White Dwarf Hamster (*Phodopus sungorus*)

fig. 1: My pet hamster

Hamsters are rodents that are native to Europe and Asia. They are solitary, which means they do not live in groups. They are nocturnal to protect themselves from predators that hunt during the day. They also like to burrow underground in burrows that they have made themselves.

The text we added is not created with four lines. Instead it is only one line in .txt file, but it is distributed so it looks like a whole paragraph. You must remember about it, if you

wanted to number each line of this text. Now we've added second line of text, which also turns out to render as a whole paragraph.

Jerry Bonion 30.01.2025

Essay: My favourite animal.

Species: Winter White Dwarf Hamster (*Phodopus sungorus*)

fig. 1: My pet hamster

Hamsters are rodents that are native to Europe and Asia. They are solitary, which means they do not live in groups. They are nocturnal to protect themselves from predators that hunt during the day. They also like to burrow underground in burrows that they have made themselves.

Winter White Dwarf Hamsters live in region of Asia which is placed between Russia and Kazakhstan. Their natural environments are steppes and semi-deserts. They are chubby animals, but they can run very fast for their sizes.

Now let's add second image. This time we will put it at right side of page. And include more text, this time numbered.

```
<<RI-60>apple.png

<<cR>Apple I feed my hamster


Hamsters are omnivores, which means they can eat a lot of stuff:

<<N>Seeds

<<N>Fruits

<<N>Vegetables
```

```
<<N>Insects

<<N>Meat
```

Winter White Dwarf Hamsters live in region of Asia which is placed between Russia and Kazakhstan. Their natural environments are steppes and semi-deserts. They are chubby animals, but they can run very fast for their sizes.

fig. 2: Apple I feed my hamster

Hamsters are omnivores, which means they can eat a lot of stuff:

1 Seeds
2 Fruits
3 Vegetables
4 Insects

5 Meat

Our document already has two pages and they are numbered correctly. Now anytime we use `<<N>` code, number will increase by one. If we want to make second list, we need to use `<<n>` code.

```
<<C>Reasons to have a hamster:

<<nNC>They are --funny--

<<NC>They are --cute--

5 Meat
```

Reasons to have a hamster:

1 They are funny
2 They are cute

Note that line numbers are not centered. They only number lines, not list.

Have fun using HeinzoScript for your own documents!