ASCII Dots
Brief Documentation

Dots
A dot (.) is what begins your program, your dot is where its happening

Paths
A path (| and -) carries the dot, .----$"Printing..." Once your dot hits that $ following the path, the $ will prompt your dot to print whatever follows $# prints the value of your dot, and $a# prints the ASCII value of your dot $_# prints without adding a new line to the end

Special Paths
Some paths do more than just carry a dot, thats why they are special. Special paths include: >, this one takes a dot coming from the left, and dots coming veritcally and outputs them all to the right. < does the same but takes it from the right and outputs from the left. v and ^ do this aswell for vertical outputs. (and) bounce a dot back teh way it came. \ and / act as mirrors so .-/ exits going upwards

Warps (Invisible Path)
.-A A-$"This executes"-&-(The & kills a dot so .--&--$"I am unexecutable")

Operators
[+] takes a input vertically and a horozontal input and outputs to the other vertical direction the sum of the ttwo dot values {+} does the same but outputs to the other horozontal side. You can substitute any operator into this.

~
Receives a value horozontally, and vertically. If the vert. value is not 0 it continues the horz. dot to the other vert. direction. if it is 0; however, we continue horozontally.

Inspirations for the language
ASCII Art
ASCII Art was a driving inspiration for this language, meant to look aestethic and artistic to programmers and non-programmers alike, unlike other languages where we just look at a bunch of operations and indentations with a bunch of words randomly thrown in.

Befunge
Another wonky eso-lang, famously the first two-dimmmensional programming language. This language also runs along 'paths' but on a larger scale. A link for more reading on the language can be found at https://en.wikipedia.org/wiki/Befunge

Mechanical Computers
Mechanical computers are all very logical, as is this language. Data flow is very visual and connected as a mechanical computer is in its many gears.

Sources
Repo
https://github.com/aaronjanse/asciidots?tab=readme-ov-file

Documentation
https://ajanse.me/asciidots/language/

Library Documentation
https://ddorn.github.io/asciidots/docs/libs/

ESO-Langs Article
https://esolangs.org/wiki/AsciiDots