Basics

* free-form
* whitespace - spaces, tabs, blank lines
* scripts are made up of 'procedures' which are made up of 'statements'
* semi-colons terminate statements
* comments - // and /* .. */

/* this is a comment
that spans multiple lines */
// this is a one liner

Variables

* $..
* proper naming is essential to maintainability
* a variable has a name and a type, and holds data. Data in, data out.

* int
* float
* string
* vector
* matrix

- strings can be concatenated [single strings, NOT string arrays; see below for defn.of arrays]
- can +,-,*,/and % a matrix by a scalar [float or int]
- rows are separated by semicolons, columns by commas
- multiplication of matrices yields a new result matrix
- can negate a matrix
- indices start at 0, not 1
- to access an individual element, say '$f = $m[0][2]`,etc.
* arrays, ordered collections of primitive types [except matrices]
- can grow as needed [dynamic memory allocation]
- size($arrayname)
- clear() to delete
- access: [] to specify an index, eg. '$nextNum = $hugeArray[32];'
- indices start at 0,not 1
* can't add more, int/float/string/vector/matrix are the only available types

```
int $i; int $j=5;
 int $k, $l;
 int $i = 4.765; // will be truncated (NOT rounded) to 4

 float $f1;
 float $f1a, $f1b;
 float $f2 = 6.65;
 float $f3 = 1/3; // $f3 will be 0, NOT 0.3333333
 float $f4 = 1.0/3; // will be 0.333333333 as expected

 string $a;
 string $fnm = "Los ";
 string $lnm ="Angeles";
 string $fullName = $fnm + " " + $lnm; // concatenation;
also note the use of a space character

 vector $v1 = <<1.0,0.56,-0.96>>;
 print($v1.x); // NEED the ()
 $v1 = <<$v1.x, $v1.y,10.0>>; // NEED to reset all values,
can't just set $v1.z [see below]
 $v1.z = 10.0; // NOT allowed, syntax error

 matrix $n[2][2];
 matrix $i2[2][3] = <<1,2,3;-6,.78,.45>>; // 2 rows,3
columns. Note the use of ';' to separate rows
 $j2 = $i2; // $j2 is another matrix created with same
contents as $i2 [copied over]
 $k2 = -$i2; // $k2 is a new matrix with values from $i2
NEGATED and copied over

 int $a2[];
 int $b2[100];
```

```
// here's how to initialize an array [specify all values while creating the variable]
float $flArr[6]= {0.8,-.6,1.,14.,12.3,-7.6};
```

* arrays - for lists & temporary storage, size() command, clear() command

* global vs. local vars