```
#!\Progra~1\Perl\bin\perl.exe -w

#use diagnostics;
#use strict;
use LWP::Simple;
use XML::DOM;

my $xmldocument = new XML::DOM::Document;
my $keplerian_elements = $xmldocument->createElement("keplerian-elements");
$xmldocument->appendChild($keplerian_elements);

my $urlBase = "http://www.amsat.org/amsat/ftp/keps/current/nasa.all";

my $webPage = get("$urlBase") or die $!;

my $orbital_parameters = $webPage;

# "Skip" ahead past the following text
if ($orbital_parameters =~ /TO ALL RADIO AMATEURS BT/)
{

}

my %orbital_elements_hash;

while($orbital_parameters =~ /(.{2,5})\s+(\d)\s+(\d{5}\w)\s+(\d{5}\w)\s+(\d{5}\.\d{
8})\s+(\.\d{8})\s+(\d{5}\-\d)\s+(\d{5}\-\d)\s+(\d)\s+(\d{4,5})\s+(\d)\s+(\d{5})\s+(\
d{2,3}\.\d{4})\s+(\d{3}\.\d{4})\s+(\d{7})\s+(\d{2,3}\.\d{4})\s+(\d{3}\.\d{4})\s+(\d{
2}\.\d{14})\s+/g)
{
        $orbital_elements_hash{$1} = [ ($1, $3, $4, $5, $6, $7, $8, $9,
                                        $10, $12, $13, $14, $15, $16, $17,
                                        $18, "Reserved") ];
}

###
#
#

foreach $array (values %orbital_elements_hash)
{
   my $index = 0;
   my $orbital_element = $xmldocument->createElement("orbital-element");

   foreach $dataitem (@$array)
   {
      $keplerian_elements->appendChild($orbital_element);

      if($index == 0)
      {
        print "$dataitem ";
      }
      if($index == 1)
      {
        my $nasa_catalog_number = $xmldocument->createElement("nasa-catalog-number"
);
        $orbital_element->appendChild($nasa_catalog_number);
        $nasa_catalog_number->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 2)
      {
        my $launch_information = $xmldocument->createElement("launch-information");
        $orbital_element->appendChild($launch_information);
        $launch_information->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 3)
```

```
      {
        my $epoch_time = $xmldocument->createElement("epoch-time");
        $orbital_element->appendChild($epoch_time);
        $epoch_time->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 4)
      {
        my $decay_rate = $xmldocument->createElement("decay-rate");
        $orbital_element->appendChild($decay_rate);
        $decay_rate->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 6)
      {
        # Drag
      }
      elsif($index == 8)
      {
        my $element_set = $xmldocument->createElement("element-set");
        $orbital_element->appendChild($element_set);
        $element_set->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 10)
      {
        my $inclination = $xmldocument->createElement("inclination");
        $orbital_element->appendChild($inclination);
        $inclination->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 11)
      {
        my $ra_of_node = $xmldocument->createElement("right-ascension-of-node");
        $orbital_element->appendChild($ra_of_node);
        $ra_of_node->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 12)
      {
        my $eccentricity = $xmldocument->createElement("eccentricity");
        $orbital_element->appendChild($eccentricity);
        $eccentricity->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 13)
      {
        my $arg_of_perigee = $xmldocument->createElement("argument-of-perigee");
        $orbital_element->appendChild($arg_of_perigee);
        $arg_of_perigee->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 14)
      {
        my $mean_anomaly = $xmldocument->createElement("mean-anomaly");
        $orbital_element->appendChild($mean_anomaly);
        $mean_anomaly->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 15)
      {
        my $mean_motion = $xmldocument->createElement("mean-motion");
        $orbital_element->appendChild($mean_motion);
        $mean_motion->appendChild($xmldocument->createTextNode($dataitem));
      }
      elsif($index == 16)
      {
         # revolution-at-epoch
         # first-derivative-of-mean-motion
      }
      $index++;
   }
   $xmldocument->printToFile("Satellites.xml");
}
```