Advanced Scripting   
Classes

Last Updated: 11/8/2021 3:55 PM Version 1  
Document Prepared for: CIT361 Student

# Name Zach Lucas ID 895649438

# Instructions

Save a copy of this document. Answer all questions directly in this document. You will save and upload this completed document as your homework submission.

# Overview

In this exercise you will create a simple class used to import metal information from our metals file this will provide an object with appropriate types for the data.

# Requirements

PowerShell

Class sample files <http://cf.esage.com/psfiles.zip>

# Setup

# Task 1—Explore the metals data

## Steps

1. Import the data from Metals.csv file, use your path to the Metals.csv file.  
   import-csv Metals.csv|Format-Table
2. Examine the columns and complete the following table with the Column Name and the data type that would be the most appropriate to store the data in.

|  |  |
| --- | --- |
| Column | DataType |
| Symbol | String |
| Name | String |
| MeltingPoint | Int |
| SpecificGravity | Int |

# Task 2—Create A Metal Class

Create a class to hold the metal data. I’ll get you started you will need to complete the properties

## Steps

1. Create a file named MetalImport.ps1
2. Create a class to hold metal data that contains a type restricted property for each column in the Metals file, the property name should match the Column name:  
   Class Metal{  
    [string]$Symbol  
    #Add other Properties as appropriate  
   }
3. Create an instance of the class three ways
   1. Call the new method  
      $m1=[Metal]::New()
      1. Set some properties  
         $m1.Symbol='Au'  
         $m1.Name='Gold'  
         $m1.MeltingPoint=1945  
         $m1.SpecificGravity=19.3
      2. Take a look at the results  
         $m1
   2. Use New-Object cmdlet  
      $m2=New-Object Metal  
      $m2
   3. Initialize with a hashtable  
      $m3=[Metal]@{Symbol='Au';Name='Gold'}  
      $m3
4. Now import the data and cast to your new Metal Type (use an appropriate path to your file)  
   $Metals=import-csv Metals.csv|ForEach-Object {[Metal]$\_}
5. View the type returned  
   $Metals[0].GetType()
6. Copy your completed script here:

class Metal {  
 [string]$Symbol  
 [string]$Name  
 [Int32]$MeltingPoint  
 [Int32]$SpecificGravity  
}  
  
$m1=[Metal]::new()  
 $m1.Symbol='Au'  
 $m1.Name='Gold'  
 $m1.MeltingPoint=1945  
 $m1.SpecificGravity=19.3  
  
$m1  
  
$m2=New-Object Metal  
$m2  
  
$m3=[Metal]@{Symbol='Au';Name='Gold'}  
$m3  
  
$Metals=Import-Csv ./Metals.csv|ForEach-Object{[Metal]$\_}  
$Metals[0].GetType()

# Deliverable

Upload this document with completed answers to i-learn.