Create a new sorted string using two strings with random characters in them. Strings contains only letters from ***a*** to ***z***.

class NewString{

    NewString(){

    }

    [string]MakeString([string]$s1, [string]$s2){

        [string]$st1 = $this.MakeOne($s1, $s2)

        [string]$st2 = ""

        for( [int]$i = 0; $i -lt $st1.length; $i++ ){

            [int]$k = 0

            for( [int]$j=0; $j -lt $st2.length; $j++ ){

                if ($st1[$i] -eq $st2[$j]){

                    $k++

                }

            }

            if ($k -eq 0){

                $st2 += $st1[$i]

            }

        }

        return $this.MakeSort($st2)

    }

    [string]MakeOne([string]$s1, [string]$s2){

        [int]$i=0

        for($i; $i -lt $s2.length; $i++){

            $s1 += $s2[$i]

        }

        return $s1

    }

    [string]MakeSort([string]$str){

        [int]$fl = 0

        [array]$sn = $str.ToCharArray()

        Do {

            $fl = 0

            for ([int]$i=0; $i -lt $sn.length -1; $i++){

                if ($sn[$i] -ge $sn[$i+1]){

                    [char]$max = $sn[$i]

                    $sn[$i] = $sn[$i+1]

                    $sn[$i+1] = $max

                    $fl++

                }

            }

        }while($fl -ne 0)

        $str=""

        for ([int]$i=0; $i -lt $sn.length; $i++){

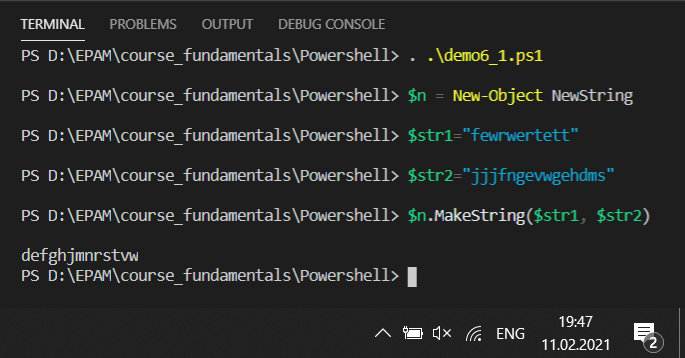
            $str+=$sn[$i]

        }

        return $str

    }

}



#------------------------------------------------------

Write a script that it converts dash/underscore delimited words into camel and pascal casing.

class Converse{

    Converse(){

    }

    [string]MakeString([string]$str){

        [array]$s=$str.ToCharArray()

        [string]$ns=""

        for([int]$i=0; $i -lt $s.length; $i++){

            if( $s[$i] -ne "\_" -and $s[$i] -ne "-" ){

                $ns += $s[$i]

            }else{

                $i++

                $ns += ($s[$i]).ToString().ToUpper()

            }

        }

    return $ns

    }

}

