

# Windows PowerShell Funktionen

## 1 Theorie

### 1.1 Definition und Ausführung einer Funktion

Funktionen erlauben Programmlogik oder komplexe Aufrufe von Cmdlets für eine spätere Wiederverwendung zu speichern.

Funktionen müssen immer am Anfang eines Skripts stehen.

```
# Definition der Funktion
function IstTeilbar ($z1, $z2) {
    if ( $z1 % $z2 -eq 0 ) {
        return "teilbar"
    } else {
        return " nicht teilbar"
    }
}

# Ausführung Variante 1
$a = IstTeilbar 21 6
Write-Host $a

# Ausführung Variante 2
$z1 = 20
$z2 = 4
Write-Host "Die Zahl $z1 ist durch $z2 $(IstTeilbar $z1 $z2)"
```

## 1.2 Parameterblöcke

```
# Definition der Funktion
function IstTeilbar {
    param (
        [Parameter(mandatory=$true)]
        [int]$z1,
        [Parameter(mandatory=$false)]
        [int]$z2=2
    )

    if ( $z1 % $z2 -eq 0 ) {
        return "teilbar"
    } else {
        return " nicht teilbar"
    }
}

# Ausführung Variante 1
$a = IstTeilbar -z1 21 -z2 6
Write-Host $a

# Ausführung Variante 2
$z1 = 20
Write-Host "Die Zahl $z1 ist durch 2 $(IstTeilbar -z1 $z1)"
```

## 2 Übungen

### Aufgabe 1

Erstellen Sie eine Funktion, welche °Celsius in °Fahrenheit umrechnet. Die Umrechnungsformel lautet: °**Fahrenheit** = (°**Celsius** \* 9) / 5 + 32.

### Aufgabe 2

Erstellen Sie eine Funktion, welche das aktuelle Datum in einer Hashtable zurückgibt. Die Hashtable weist folgende Schlüssel-Wertepaare auf: d=Tag, m=Monat, y=Jahr.

### Aufgabe 3

Erstellen Sie eine Funktion, welche die Eigenschaften **Id**, **ProcessName**, **Path** der "sichtbaren" Prozesse zurückgibt. Dies ist der Fall, wenn die Eigenschaft **MainWindowTitle** nicht leer ist. Falls der Parameter "**ShowGrid**" **\$true** ist, soll die Aufgabe in einer **GridView** erfolgen.