W02 Walidacja Parametrów Funkcji / Skryptu

W PowerShellu parametryzacja skryptów oraz walidacja danych to kluczowe zagadnienia pozwalające tworzyć solidne, bezpieczne i łatwe w utrzymaniu skrypty.

Parametryzacja skryptów/Funkcji – możliwości

- Blok param() – Definiuje parametry wejściowe w skryptach i funkcjach.
- Atrybuty typu ([string], [int] itd.) – Wymuszają typ danych.
- Parametry opcjonalne z wartościami domyślnymi – np. $x = 10.
- Parametry obowiązkowe ([Parameter(Mandatory=$true)]) – Wymuszają podanie wartości.
- Pozycjonowanie parametrów (Position = 0) – Umożliwia przekazywanie bez nazw.
- Walidacja danych (poniżej szczegóły).

[ValidateNotNull]

Zapobiega przekazaniu wartości null (np. $null). Skrypt zakończy się błędem, jeśli parametr nie zostanie przekazany lub będzie równy null.

```
[ValidateNotNull()]
[string]$Imie
```

[ValidateNotNullOrEmpty]

Nie zezwala na przekazanie wartości null ani pustego ciągu ("").

```
[ValidateNotNullOrEmpty()]
[string]$Nazwisko
```

[ValidateRange(min, max)]

Wymusza, by wartość była w określonym zakresie liczbowym.

```
[ValidateRange(1, 100)]
[int]$Wiek
```

[ValidateLength(min, max)]

Stosowane dla ciągów znaków – ogranicza długość tekstu.

```
[ValidateLength(3, 10)]
[string]$Login
```

[ValidateSet(...)])

Ogranicza możliwe wartości parametru do wskazanych opcji.

```
[ValidateSet("Start", "Stop", "Restart")]
[string]$Akcja
```

[ValidatePattern("regex")]

Sprawdza zgodność wartości z wyrażeniem regularnym (np. adres e-mail, numer telefonu).

```
[ValidatePattern("^[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}$")]
[string]$Email
```

[ValidateScript({scriptblock})]

Pozwala na wykonanie własnej logiki walidacyjnej w bloku skryptowym. Można tu umieścić dowolny kod, który zwraca $true/$false.

```
[ValidateScript({ Test-Path $_ })]
[string]$Plik
```

Typ parametru jako walidacja

Typ danych przypisany do parametru (np. [int], [string]) również działa jak walidacja. Jeśli przekazana wartość nie daje się skonwertować, pojawi się błąd.

```
[int]$Liczba
```

Połączenia atrybutów walidacyjnych

Można łączyć wiele atrybutów walidacyjnych dla jednego parametru.

```
[ValidateNotNullOrEmpty()]
[ValidatePattern("^[A-Z]{3}\d{3}$")]
[string]$KodProduktu
```