In PowerShell, the `||` operator (logical OR) commonly used in Bash is **not directly supported**. However, you can achieve similar functionality using PowerShell's logical operators or conditional constructs. Here's how you can replicate the behavior:

1. Using `-or` Logical Operator

The `-or` operator is used to evaluate two conditions. If either condition is true, the result is true.

```
if ($condition1 -or $condition2) {
    Write-Output "At least one condition is true."
}
```

2. Using `if` Statements for Command Execution

To mimic `||` for command execution (run the second command only if the first fails), you can use an `if` statement with `$?`, which checks the success of the previous command.

```
Get-Command NonExistentCommand
if (-not $?) {
    Write-Output "The first command failed, so this runs."
}
```

3. Using `try` and `catch` for Error Handling

For more robust error handling, you can use `try` and `catch` blocks to execute a fallback command if the first one fails.

```
try {
    Get-Command NonExistentCommand
} catch {
    Write-Output "The first command failed, running fallback."
}
```

While `||` isn't directly available, these approaches provide equivalent functionality in PowerShell.