

Bahan Ajar

Chapter 8



Materi Pembelajaran

Matakuliah :

# INTERFACING PERIPHERAL

Kode Matakuliah : SKO 20416

Prodi : SISTEM KOMPUTER

Dosen Pengampu Matakuliah:

Bayu Nugroho, S.Kom., M.Eng

# Tables of Content

## Python Loops

- The while Loop
- The break Statement
- The continue Statement
- The else Statement

## Python For Loops

- Looping Through a String
- The break Statement
- The continue Statement

## Tugas Mandiri

# The while Loop

With the while loop we can execute a set of statements as long as a condition is true

## Example

Print i as long as i is less than 6:

```
i = 1
while i < 6:
    print(i)
    i += 1
```

# The break Statement

With the break statement we can stop the loop even if the while condition is true:

## Example

Exit the loop when i is 3:

```
i = 1
while i < 6:
    print(i)
    if i == 3:
        break
    i += 1
```

# The continue Statement

With the continue statement we can stop the current iteration, and continue with the next:

## Example

Continue to the next iteration if i is 3:

```
i = 0
while i < 6:
    i += 1
    if i == 3:
        continue
    print(i)
```

# The else Statement

With the else statement we can run a block of code once when the condition no longer is true:

## Example

Print a message once the condition is false:

```
i = 1
while i < 6:
    print(i)
    i += 1
else:
    print("i is no longer less than 6")
```

# Python For Loops

A for loop is used for iterating over a sequence (that is either a list, a tuple, a dictionary, a set, or a string). This is less like the for keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages.

With the for loop we can execute a set of statements, once for each item in a list, tuple, set etc.

## Example

Print each fruit in a fruit list:

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
```

# Looping Through a String

Even strings are iterable objects, they contain a sequence of characters:

## Example

Loop through the letters in the word "banana":

```
for x in "banana":  
    print(x)
```

# The break Statement

With the break statement we can stop the loop before it has looped through all the items:

## Example

Exit the loop when `x` is "banana":

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    print(x)
    if x == "banana":
        break
```

# The continue Statement

With the continue statement we can stop the current iteration of the loop, and continue with the next:

## Example

Do not print banana:

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
    if x == "banana":
        continue
    print(x)
```

## **Tugas Mandiri (teori):**

1. A for loop is a count-controlled loop, and a while loop is a condition-controlled loop. True or false? Explained.
2. I am running my Python script, and it is stuck in an infinite loop! What do I do?

## **Tugas Mandiri (prakt):**

Tuliskan script program python untuk menghidupkan 6 buah LED secara bergantian di Raspberry Pi menggunakan fungsi **Loops** dengan beberapa kali pengulangan.

end

