

الوحدة 4: التكرار (Loops)

أ. مفهوم التكرار (Loops Concept)

التكرار (Loop) هو آلية في البرمجة تسمح بتنفيذ مجموعة من الأوامر عدة مرات دون الحاجة إلى كتابتها مرارًا وتكرارًا يدويًا.

نستخدم الحلقات عندما نريد معالجة بيانات متكررة أو تكرار مهمة حتى يتحقق شرط معين.

مثال على أهمية الحلقات:

إذا أردنا طباعة الأرقام من 1 إلى 100، لا نحتاج إلى كتابة (`print`) مئة مرة. نستخدم حلقة تقوم بالمهمة آليًا.

ب. الحلقة (for (For Loop))

تُستخدم الحلقة `for` عندما نعرف عدد التكرارات مسبقًا أو عندما نريد التكرار على عناصر سلسلة (مثل قائمة أو نص).

البنية العامة:

```
for variable in sequence:
```

```
    أوامر تنفذ في كل دورة #
```

- `variable`: متغير يمثل العنصر الحالي في كل دورة.

- `sequence`: مجموعة عناصر مثل `list`, `string`, `range()`.

أمثلة بسيطة

```
for i in range(5):
```

```
    print("Iteration:", i)
```

النتائج:

```
Iteration: 0
```

```
Iteration: 1
```

```
Iteration: 2
```

```
Iteration: 3
```

```
Iteration: 4
```

مثال 1: طباعة الأعداد من 1 إلى 10

```
for number in range(1, 11):
```

```
    print(number)
```

مثال 2: طباعة أسماء الطلبة من قائمة

```
students = ["Ali", "Sara", "Omar"]
```

```
for name in students:
```

```
    print("Student:", name)
```

ج. الحلقة (While Loop) while

تُستخدم الحلقة while عندما لا نعرف عدد التكرارات مسبقاً، وتستمر في التنفيذ ما دام الشرط True.

البنية العامة:

```
while condition:
```

```
    True أوامر تنفذ طالما الشرط #
```

- condition : شرط منطقي يتم تقييمه في كل دورة.

- يجب أن يتغير الشرط داخل الحلقة وإلا ستحدث حلقة لا نهائية. (Infinite Loop)

مثال بسيط

```
count = 0

while count < 5:

    print("Count:", count)

    count += 1
```

مثال 1: طلب كلمة مرور صحيحة

```
password = ""

while password != "Python123":

    password = input("Enter password: ")

print("Access granted!")
```

د. أوامر التحكم في الحلقات (Control Statements)

- **break**: إيقاف الحلقة مبكرًا

تُنتهي تنفيذ الحلقة فورًا حتى لو لم يكتمل التكرار.

```
for i in range(10):

    if i == 5:

        break

    print(i)
```

الناتج:

0
1
2
3
4

• **continue**: تخطي دورة واحدة

تتجاهل بقية أوامر الدورة الحالية وتنتقل للدورة التالية.

```
for i in range(5):  
  
    if i == 2:  
  
        continue  
  
    print(i)
```

الناتج:

0
1
3
4

هـ. تطبيقات عملية (Practical Applications)

1. حساب مجموع الأعداد من 1 إلى N

```
n = int(input("Enter a number: "))

total = 0

for i in range(1, n + 1):

    total += i

print("Sum =", total)
```

2. العد التنازلي (Countdown)

```
count = 10

while count > 0:

    print(count)

    count -= 1

print("Time's up!")
```

3. جدول الضرب (Multiplication Table)

```
number = int(input("Enter a number: "))
```

```
for i in range(1, 11):  
    print(number, "x", i, "=", number * i)
```