المحاضرة السادسة

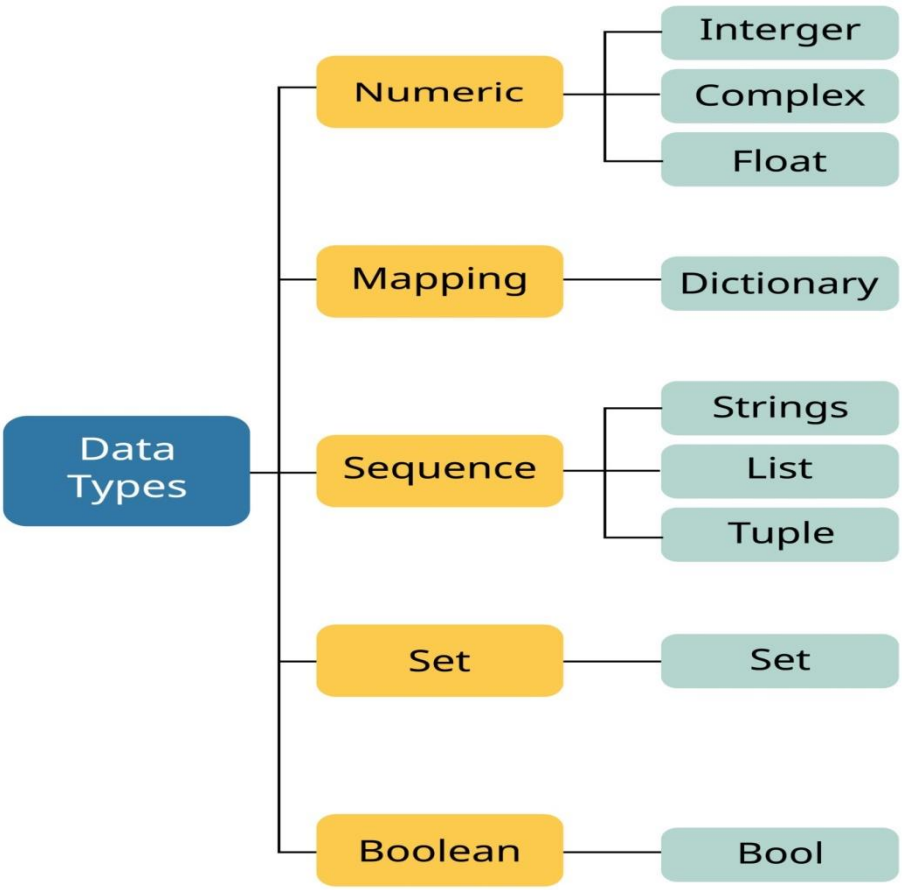
تعتمد البرمجة على ثلاثة أركان أساسية تحدد كيفية تخزين البيانات وتفسيرها ومعالجتها بواسطة الحاسوب: وهي المتغيرات، و أنواع البيانات، وهياكل التحكم.(Lynch, 2020)

1. المتغيرات(Variables)

يُعرّف المتغير (Variable) بأنه موقع مُسمّى في ذاكرة الحاسوب يُستخدم لتخزين قيمة أو معلومة مؤقتاً أثناء تنفيذ البرنامج .(Dovgan, 2021) تكمن أهمية المتغيرات في أنها تسمح للمبرمج بالإشارة إلى البيانات المتغيرة أو المعقدة من خلال اسم بسيط وواضح. تُسمى عملية منح قيمة ابتدائية للمتغير الإسناد (Assignment)، ويُرمز لها عادةً بعلامة التساوي (=) (Dovgan, 2021) .

2. أنواع البيانات(Data Types)

تُحدد أنواع البيانات طبيعة المعلومات التي يمكن للمتغير أن يخزنها، وهي ضرورية لتحديد العمليات المسموح بإجرائها على هذه القيمة ولضمان التخصيص الفعال للذاكرة ومنع الأخطاء المنطقية



تشمل الأنواع الأساسية الأولية ما يلي:(Lynch, 2020)

- العدد الصحيح :(Integer) للأعداد التي لا تحتوي على جزء كسري.
- العدد العشري :(Float/Double) للأعداد الحقيقية ذات الفواصل.
- السلسلة الحرفية :(String) لتخزين النصوص.
- القيمة المنطقية :(Boolean) تحتمل إحدى قيمتين فقط) صحيح أو خطأ(، وهي ضرورية لتقييم الشروط.

3. هياكل التحكم(Control Structures)

تُعد هياكل التحكم كتل برمجية تحدد الترتيب التسلسلي أو الشرطي الذي يتم به تنفيذ تعليمات البرنامج .(Dovgan, 2021)وهي تمنح البرنامج قدرته على اتخاذ القرارات والأتمتة، مُمكِّنةً إياه من الانحراف عن مسار التنفيذ الخطي البسيط.

- الهياكل الشرطية :(Conditional Structures) تسمح للبرنامج باتخاذ قرارات عبر تقييم تعبير منطقي. إن عبارات IF / ELSE هي الأشهر في هذا المجال؛ حيث يتم تنفيذ كتلة من التعليمات إذا كان الشرط صحيحاً، وكتلة أخرى (اختيارياً) إذا كان الشرط خاطئاً.
- هياكل التكرار :(Iteration / Loops) تُستخدم لتكرار كتلة من التعليمات عدة مرات. تُستخدم حلقة FOR عادة للتكرار على تسلسل محدد من العناصر، بينما تُستخدم حلقة WHILE لتنفيذ التعليمات طالما ظل شرط معين صحيحاً، وهي مفضلة عندما لا يكون عدد التكرارات معروفاً مسبقاً.(Lynch, 2020)

المتغيرات(Variables)

المتغير هو حاوية تخزن البيانات في الذاكرة، يُعطى اسم فريد ويحمل قيمة يمكن تغييرها أثناء تشغيل البرنامج

مثال:(Python)

python

```
name = "نص"  # أحمد
age = 25  # رقم صحيح
```

رقم عشري # height = 1.75

القواعد:

اسم فريد، يبدأ بحرف أوunderscore

حساس لحالة الأحرف(case-sensitive)

يمكن تغيير القيمةage = 26 :

أنواع البيانات(Data Types)

تصنف البيانات حسب طبيعتها لتحديد العمليات المسموحة عليها.

النوع	الوصف	مثال(Python)	العمليات المسموحة
رقم صحيح(int)	أرقام كاملة موجبة/سالبة	x = 42	+ - * / %
رقم عشري(float)	أرقام بها كسور	pi = 3.14	حسابات دقيقة
نص(string)	حروف، أرقام، رموز	text = "مرحباً"	الدمج، القطع
منطقي(boolean)	صح/خطأ	is_adult = True	and, or, not
قائمة(list)	مجموعة عناصر	numbers = [1, 2, 3]	إضافة، حذف
قاموس(dict)	أزواج مفتاح-قيمة	student = {"name": "فاطمة"}	البحث السريع

فحص النوعtype(age) ⟶ lass 'int'> :

بنى التحكم(Control Structures)

تتحكم في تدفق البرنامج وتتيح اتخاذ القرارات والتكرار.

أ. الشروط if/else - (Conditionals)

فحص النوع type(age) ⟶ lass 'int'> :

```python
age = 20
if age >= 18:
    print("بالغ")      # يطبع هذا
elif age >= 13:
    print("مراهق")
else:
    print("طفل")
```

الاستخدام :اختبار الشروط وتنفيذ كود مختلف.

ب. الحلقات(Loops)

1. حلقة for (تكرار عدد معروف):

```python
for i in range(5):      # 0,1,2,3,4
    print(f"الرقم: {i}")
```

2.حلقة : while (تكرار حتى شرط)

```python
count = 0
while count < 3:
    print(count)
    count += 1      # 0,1,2
```

ج. أوامر التحكم

- **break**:إيقاف الحلقة
- **continue**:تخطي التكرار الحالي
- **pass**:لا شيء(placeholder)

مثال متكامل:

```python
# حساب درجات الطلاب
grades = [85, 92, 78, 95]
total = 0

for grade in grades:
    if grade >= 90:
        print(f"ممتاز: {grade}")
    total += grade

average = total / len(grades)
print(f"المتوسط: {average}")  # 87.5
```

يجمع متغيرات، أنواع، شروط، وحلقات.

الفرق الرئيسي:

المفهوم	الغرض
متغيرات	تخزين البيانات
أنواع	تحديد طبيعة البيانات
بنى تحكم	توجيه تدفق البرنامج

هذه الأساسيات تشكل 80% من البرمجة اليومية، فهمها يتيح كتابة برامج معقدة