الوحدة 2: الأساسيات(Basics)

أ. أنواع البيانات في بايثون(Data Types in Python)

أنواع البيانات (Data Types) هي التصنيفات التي تحدد طبيعة القيمة المخزّنة في الذاكرة وكيفية التعامل معها في البرنامج.

في بايثون، يتم تحديد نوع البيانات تلقائيًا عند إسناد قيمة إلى متغير .(Variable)

ب. الأنواع الأساسية في بايثون

النوع(Type)	الوصف(Description)	مثال(Example)
int (عدد صحيح Integer)	يخزن أعدادًا صحيحة بدون فاصلة عشرية	age = 20
float (عدد عشري Float)	يخزن أعدادًا عشرية بفاصلة	price = 19.99
str (سلسلة نصية String)	يخزن نصوصًا أو حروفًا	name = "Ali"
bool (قيمة منطقية Boolean)	يخزن قيمتين فقط True :أوFalse	is_student = True

أمثلة بسيطة

```
age = 20         # int

price = 19.99    # float

name = "Ali"     # str

is_student = True  # bool

print(age, price, name, is_student)
```

مثال

برنامج بسيط يخزن بيانات طالب:

```
student_name = "Sara"       # str

student_age = 18            # int

average_grade = 15.7        # float

passed = True               # bool



print("Name:", student_name)

print("Age:", student_age)

print("Average:", average_grade)

print("Passed:", passed)
```

ج. المتغيرات وقواعد التسمية(Variables and Naming Rules)

المتغير (Variable) هو اسم يُستخدم للإشارة إلى موقع في الذاكرة يُخزّن فيه قيمة قابلة للتغيير أثناء تشغيل البرنامج.

د. قواعد تسمية المتغيرات(Naming Rules)

- يجب أن يبدأ الاسم بحرف أو شرطة سفلية _ولا يبدأ برقم.
- يمكن أن يحتوي على حروف وأرقام وشرطة سفلية فقط.
- الأسماء حسّاسة لحالة الأحرف.(Case-sensitive)
- لا يجوز استخدام الكلمات المحجوزة.(Reserved Keywords)

أمثلة صحيحة:

```
name = "Ali"
```

```
student_age = 20

_total = 100
```

أمثلة خاطئة:

```
2name = "Ali"      # يبدأ برقم

class = "Math"     # كلمة محجوزة

student-age = 20  # يحتوي على رمز غير مسموح
```

مثال

```
first_name = "Omar"

last_name = "Hassan"

full_name = first_name + " " + last_name



print("Full Name:", full_name)
```

هـ. العمليات الحسابية والمنطقية(Arithmetic & Logical Operators)

العمليات الحسابية(Arithmetic Operators)

العملية (Operation)	الرمز (Symbol)	المثال (Example)	الناتج (Result)	المثال (Example)
الجمع (Addition)	+	5 + 3	8	5 + 3
الطرح (Subtraction)	-	10 - 4	6	10 - 4
الضرب (Multiplication)	*	6 * 2	12	6 * 2

القسمة (Division)	/	10 / 2	5.0	10 / 2
باقي القسمة (Modulus)	%	10 % 3	1	10 % 3
الأس (Exponent)	**	2 ** 3	8	2 ** 3
القسمة الصحيحة (Floor Division)	//	10 // 3	3	10 // 3

العمليات المنطقية(Logical Operators)

العملية(Operation)	**الرمز(Symbol)**	**الوصف(Description)**
و(AND)	and	Trueفقط إذا كانت كل الشروطTrue
أو(OR)	or	Trueإذا تحقق أحد الشروط
ليس(NOT)	not	يعكس القيمة المنطقية

أمثلة بسيطة

```
a = 10

b = 5



print(a + b)      # 15

print(a > b and b > 0)   # True
```

مثال

```
score = 75

passed = score >= 50 and score <= 100

print("Passed:", passed)
```

و. دوال الإدخال والإخراج(Input & Output Functions)

- **دالة الإخراج(Output): print()**

تُستخدم لعرض النصوص أو النتائج على الشاشة.

print("Welcome to Python")

- **دالة الإدخال(Input): input()**

تُستخدم للحصول على مدخلات المستخدم من لوحة المفاتيح. القيمة المدخلة تكون دائمًا من نوع .str

```
name = input("Enter your name: ")

print("Hello,", name)
```

مثال

```
برنامج لحساب مجموع رقمين مدخلين من المستخدم:

a = input("Enter first number: ")

b = input("Enter second number: ")



result = int(a) + int(b)

print("The sum is:", result)
```

ز. تحويل الأنواع(Type Casting)

عند التعامل مع بيانات مدخلة أو عمليات حسابية، نحتاج أحيانًا إلى تحويل نوع البيانات يدويًا.

دوال التحويل الأساسية:

الدالة (Function)	الاستخدام(Usage)	مثال (Example)

int()	تحويل إلى عدد صحيح	int("10")
float()	تحويل إلى عدد عشري	float("3.14")
str()	تحويل إلى نص	str(100)

مثال بسيط

```
x = "50"

y = int(x) + 10

print(y)  # 60
```

مثال

برنامج لحساب معدل طالب بناءً على درجات مدخلة:

```
math = float(input("Enter Math grade: "))

physics = float(input("Enter Physics grade: "))

english = float(input("Enter English grade: "))



average = (math + physics + english) / 3

print("Average grade:", average)
```