





الدرس: أوامر الطباعة والمتغيرات (#C)

شريحة 1: شريحة العنوان

■ أوامر الطباعة والمتغيرات

أساسيات علوم الحاسوب

شريحة 2: ماذا سنتعلم اليوم؟

-  الطباعة على الشاشة: كيف نجعل البرنامج يعرض معلومات.
-  المتغيرات: ما هي وكيف نستخدمها لتخزين المعلومات.
-  استقبال المدخلات: كيف نجعل البرنامج يتفاعل مع المستخدم.
-  ذاكرة الحاسوب: أين يتم تخزين هذه المعلومات مؤقتاً.

🎯 سؤال للنقاش:

"كيف تتعرف عليك الألعاب وتناديك باسمك؟"

شريحة 3: الهيكل الأساسي للبرنامج

كل برنامج C# يبدأ بهذا الهيكل. فكر فيه كـ "الحاوية" التي نضع بداخلها أوامرنا.

class Program

```
{  
  
    static void Main(string[] args)  
  
    {  
  
        !أوامرنا تُكتب هنا //  
  
    }  
  
}
```

- **Main:** هي نقطة البداية. الحاسوب يبدأ بقراءة وتنفيذ الأوامر من هنا.
-

شريحة 4: أوامر الطباعة

هناك طريقتان رئيسيتان لطباعة النصوص على الشاشة:

الأمر	الشرح	مثال الكود والنتائج
Console.Write()	يطبع النص ويبقى المؤشر في نفس السطر.	Console.Write("مرحباً"); Console.Write("يا عالم"); النتائج: مرحباً يا عالم!
Console.WriteLine()	يطبع النص ثم ينزل إلى سطر جديد.	Console.WriteLine("مرحباً"); Console.WriteLine("يا عالم"); مرحباً يا عالم!

شريحة 5: ما هو المتغير؟ 🤖

ببساطة، تخيل المتغير كصندوق تخزين 📦 .

- له اسم (ملصق): لتعرف ماذا بداخله (مثال: studentName).
 - له نوع: يحدد ما الذي يمكن وضعه بداخله (نصوص، أرقام، ... إلخ).
 - يحتوي على قيمة: المعلومة الفعلية التي نخزنها (مثال: "أحمد").
- نحن نستخدم المتغيرات لحفظ المعلومات التي قد تتغير أو التي نحتاجها لاحقاً في البرنامج.

شريحة 6: أنواع المتغيرات (أنواع الصناديق)

في البداية، سنركز على نوعين أساسيين:

- **string (نص):** لتخزين سلسلة من الحروف والرموز (الاسم، جملة، عنوان).
 - مثال: string message = "أهلاً بالجميع";
- **int (عدد صحيح):** لتخزين أعداد صحيحة (بدون كسور عشرية) مثل العمر أو عدد النقاط.
 - مثال: int age = 15;

شريحة 7: التفاعل مع المستخدم

لجعل البرنامج تفاعلياً، نجمع بين تعريف المتغير وأمر القراءة.

الخطوة 1: تعريف متغير (صندوق) لتخزين اسم المستخدم //

```
string userName;
```

الخطوة 2: طباعة سؤال للمستخدم //

```
Console.WriteLine("ما هو اسمك؟");
```

الخطوة 3: قراءة إجابة المستخدم وتخزينها في المتغير //

```
userName = Console.ReadLine();
```

الخطوة 4: استخدام القيمة المخزنة //

```
Console.WriteLine(userName + " أهلاً بك يا");
```

- **Console.ReadLine():** يجعل البرنامج يتوقف وينتظر المستخدم ليكتب شيئاً ويضغط **.Enter**.

شريحة 8: أين تُحفظ المتغيرات؟

عندما تشغل برنامجك، يقوم نظام التشغيل بتخصيص مساحة مؤقتة له في ذاكرة الوصول العشوائي (RAM).

- فكر فيها كـ "رف" خاص ببرنامجك.
- يتم وضع كل "صناديقك" (متغيراتك) على هذا الرف لتستخدمها.
- عندما يغلق البرنامج، يتم تنظيف الرف بالكامل وتختفي كل البيانات.

شريحة 9: ✨ تحدي التطبيق

المهمة: اكتب برنامجاً كاملاً يقوم بالخطوات التالية:

1. يسأل المستخدم عن اسمه ويخزنه في متغير.
2. يسأل المستخدم عن عمره ويخزنه في متغير.
3. يطبع رسالة ترحيبية مخصصة تجمع الاسمين، مثل: "مرحباً يا خالد! عمرك 16 عاماً. سعيد بلقائك!"

تلميح: Console.ReadLine() يرجع نصاً (string). ستحتاج إلى طريقة لتحويل العمر من نص إلى رقم (int)! ابحث عن Convert.ToInt32().

شريحة 10: ملخص ما تعلمناه اليوم

- **الطباعة:** استخدمنا Write للطباعة في نفس السطر و WriteLine للطباعة في سطر جديد.
 - **المتغيرات:** هي "صناديق" نستخدمها لتخزين المعلومات (string للنصوص و int للأرقام).
 - **التفاعل:** استخدمنا ReadLine لقراءة مدخلات المستخدم.
 - **الذاكرة:** تعلمنا أن المتغيرات تُخزن بشكل مؤقت في الذاكرة (RAM).
- أحسنتم! هذه هي لبنات البناء الأساسية لأي برنامج قوي.**

🏠 وظيفة بيتية – مبنى البرنامج، أوامر الطباعة والمتغيرات بلغة C#

✳️ القسم الأول: اختر الإجابة الصحيحة

1. ما الفرق بين Console.WriteLine() و Console.Write()؟

- أ WriteLine (تطبع النص في نفس السطر، و Write تنتزل لسطر جديد.
- ب كلاهما يقومان بنفس الوظيفة.
- ج WriteLine (تطبع وتنزل إلى سطر جديد، أما Write تطبع دون النزول لسطر جديد ☒.
- د WriteLine (تطبع الأرقام فقط و Write تطبع النصوص.

2. ما وظيفة المتغير في البرنامج؟

- أ لتنفيذ العمليات الحسابية فقط.
- ب لتخزين معلومات نحتاجها لاحقاً في البرنامج ☒.
- ج لطباعة النتائج على الشاشة.
- د لحذف البيانات من الذاكرة.

3. ما نوع المتغير المناسب لتخزين اسم الطالب؟


- أ int
- ب string ☒
- ج bool
- د double

4. ماذا يفعل الأمر Console.ReadLine()؟


- أ يطبع نصاً على الشاشة.
- ب يقرأ سطرًا من المدخلات من المستخدم ☒.


○ ج) ينفذ العمليات الحسابية.


○ د) يعرض قيمة متغير.

 القسم الثاني: أكمل الكود التالي

```
class Program
{
    static void Main(string[] args)
    {
        "اطبع "أهلاً وسهلاً" //
        "إنم اطبع في نفس السطر "يا طالب الصف العاشر" //
    }
}
```

 المطلوب: استخدم Write و WriteLine بشكل صحيح.


 القسم الثالث: برمجة وتطبيق

 اكتب برنامجاً يقوم بما يلي:

1. يسأل المستخدم عن اسمه.
2. يسأل المستخدم عن عمره.
3. يطبع رسالة على الشكل التالي:
4. أهلاً يا [الاسم]! عمرك [العمر] سنة. نتمنى لك يوماً سعيداً!

تلميح:

استخدم Console.ReadLine() لقراءة المدخلات، و int.Parse() لتحويل العمر إلى عدد.

 القسم الرابع: سؤال تفكير (اختياري)

عندما تنتهي من تشغيل البرنامج، تختفي كل القيم التي أدخلتها. لماذا؟ وأين كانت محفوظة خلال التشغيل؟