

C# Questions Solutions

C# Questions Solutions - C# حول أسئلة

Beginner's Guide - دليل المبتدئين

Q1: Define Object Oriented Programming (تعريف البرمجة كائنية التوجه)

الإجابة / Answer:

التي (Objects) "هي أسلوب برمجي يعتمد على مفهوم "الكائنات (OOP) البرمجة كائنية التوجه تحتوي على بيانات وسلوكيات معاً.

Object-Oriented Programming (OOP) is a programming paradigm based on the concept of "objects" which contain data and behavior together.

Four Main Principles: / المبادئ الأساسية الأربعة

1. **التغليف (Encapsulation):** إخفاء تفاصيل البيانات وإظهار الوظائف الضرورية فقط.
 - Hiding data details and showing only necessary functions
2. **الوراثة (Inheritance):** قدرة كلاس (فئة) على وراثة خصائص وسلوكيات من كلاس آخر.
 - Ability of a class to inherit properties and behaviors from another class
3. **تعدد الأشكال (Polymorphism):** قدرة الكائنات على اتخاذ أشكال متعددة.
 - Ability of objects to take multiple forms
4. **التجريد (Abstraction):** إخفاء التفاصيل المعقدة وإظهار الوظائف الأساسية.
 - Hiding complex details and showing essential features

Simple Example: / مثال بسيط

```
// تعريف كلاس سيارة
public class Car
{
    // الخصائص (Properties)
    public string Color;
    public string Model;

    // الطرق (Methods)
    public void Start()
    {
        Console.WriteLine("السيارة تعمل - Car is starting");
    }
}
```

Q2: Write a complete C# program (كتابة كامل C# برنامج)

أعد البرنامج؛ برتاليل الصوي خعلمي ثم ميفو للأعداد
الزوجية والفردية

Required: Program with array of 10 integers, user input, count odd and even numbers

Complete Solution: / الحل الكامل

```
using System;

class Program
{
```

```

static void Main()
{
    // إنشاء مصفوفة لتخزين 10 أرقام
    // Create an array to store 10 numbers
    int[] numbers = new int[10];

    // متغيرات العد
    // Variables for counting
    int evenCount = 0; // عد الأرقام الزوجية
    int oddCount = 0;  // عد الأرقام الفردية

    // سؤال المستخدم عن الأرقام
    // Ask user to input numbers
    Console.WriteLine("الرجاء إدخال 10 أرقام صحيحة");
    Console.WriteLine("Please enter 10 integer numbers:");

    for (int i = 0; i < 10; i++)
    {
        Console.Write($"أدخل الرقم {i + 1}: ");
        numbers[i] = int.Parse(Console.ReadLine());
    }

    // استخدام foreach لتكرار كل عنصر
    // Using foreach to loop through each element
    foreach (int num in numbers)
    {
        // التحقق من زوجية أو فردية العدد
        // Check if number is even or odd
        if (num % 2 == 0)
        {
            evenCount++; // زوجي
        }
        else
        {
            oddCount++;  // فردي
        }
    }

    // عرض النتائج
    // Display results
    Console.WriteLine("\n== النتائج / Results ==");
    Console.WriteLine($"عدد الأرقام الزوجية (Even numbers): {evenCount}");
    Console.WriteLine($"عدد الأرقام الفردية (Odd numbers): {oddCount}");

    // انتظار مفتاح الخروج
    Console.WriteLine("\n...اضغط أي مفتاح للخروج");
    Console.ReadKey();
}
}

```

Step-by-Step Explanation: / شرح الكود خطوة بخطوة

- إنشاء المصفوفة:** `int[] numbers = new int[10];`
 - أرقام تخزينية
- متغيرات العد:** `evenCount` و `oddCount`
 - لحساب عدد الأرقام الزوجية والفردية
- لحلقة for للإدخال:**
 - تطلب من المستخدم إدخال 10 أرقام
- للعنصر foreach حلقة:**
 - تمر على كل رقم في المصفوفة
 - (باقي القسمة) **تلتحق** الزوجي/الفردية
 - فالرقم زوجي `num % 2 == 0` إذا كان
 - ولا فهو فردي
- طبع النتائج:** `Console.WriteLine` لطباعة النتائج الزوجي والفردية

Example Run: / مثال على التشغيل

```

أدخل الرقم 1: 5
أدخل الرقم 2: 8
أدخل الرقم 3: 3
...
عدد الأرقام الزوجية: 4

```

Q3: Answer only four from the following (أجب على أربعة فقط من التالي)

1. The correct definition of array is: (التعريف الصحيح للمصفوفة)

الإجابة الصحيحة / Correct Answer: d. float [] x = new float [3];

Explanation: - float [] يعني مصفوفة من نوع float - new هو اسم المصفوفة x - float يعني مصفوفة float [3] - float [3] عناصر - هذا هو الشكل الصحيح: نوع البيانات [] اسم المصفوفة float [3] ينشئ مصفوفة بـ 3 عناصر - نوع البيانات [الحجم] new

Wrong Options: - a. float [3] x; - خطأ في موضع الأقواس - b. float [3] x = new float []; - خطأ في الصيغة - c. float [3] x = new float [3]; - خطأ في موضع الأقواس

2. The correct definition of variables in a method: (التعريف الصحيح للمتغيرات في الدالة)

الإجابة الصحيحة / Correct Answer: c. public static void Sum(params int[] k1, params int[] k2) { }

Explanation: - params يمكن استخدام - params اسم المعامل - params مع مصفوفات - الصيغة الصحيحة

Important Note: - params لا يسمح باستخدام أكثر من C#، في الحقيقة - params واحد c بشكل عام، فالخيار params في نفس الدالة. الإجابة الأدق هي: - إذا كان السؤال عن صيغة الأخير في قائمة المعاملات params يوضح الصيغة الصحيحة - لكن عملياً، يجب أن يكون

3. The output on screen of the following codes: (ناتج الكود التالي)

```
for (int i = 0; i < 10; i++)
{
    if (i == 2 || i == 6) { continue; }
    if (i == 4 && i == 8) { break; }
    Console.Write(i);
}
```

الإجابة الصحيحة / Correct Answer: c. 01345789

الشرح خطوة بخطوة / Step-by-Step Explanation:

i	الشرط الأول First Condition	الشرط الثاني Second Condition	الإجراء Action	الناتج Output
0	لا (i ≠ 2 و i ≠ 6)	لا (0 == 4 كاذب)	طباعة	0
1	لا	لا	طباعة	1
2	نعم (i == 2)	-	continue (تخطي)	-
3	لا	لا	طباعة	3
4	لا	لا (4 == 4) لكن 8 == 4 خطأ	طباعة	4
5	لا	لا	طباعة	5
6	نعم (i == 6)	-	continue (تخطي)	-
7	لا	لا	طباعة	7
8	لا	لا (8 == 4 كاذب)	طباعة	8
9	لا	لا	طباعة	9

النتائج النهائية: 01345789

الإجابة الصحيحة / Correct Answer: a. A() { }

لها فواءيد أخرى مثل `public` 3. يمكن أن تكون الكلاسيكيات `public` أو `private` (دالة البناء Constructor) عادة تكون `public` لكن عادة تكون `private`.

```
public class A
{
    // 00000000 00000000 00000000
    public A()
    {
        Console.WriteLine("A تم إنشاء كائن من كلاس");
    }
}
```

حرف صغير (يجب أن يطابق اسم - { } a.) - b. Wrong Options / الخيارات الخاطئة
دالة - { } void A - d. دالة البناء لا يكون لها نوع إرجاع - c. int A { } (الكلاس بالصيغ
البناء لا يكون لها نوع إرجاع

الإجابة الصحيحة / Correct Answer: c. protected

مستويات وصول رئيسية في

المعدل	من خارج الكلاص		
public	نعم ✓	نعم ✓	نعم ✓
private	نعم ✓	لا X	لا X
protected	نعم ✓	نعم ✓	لا X
internal	نعم ✓	نعم (في نفس المشروع)	نعم (في نفس المشروع)

```
public class Parent
{
    public int publicVar = 1;        // 公開変数 publicVar
    private int privateVar = 2;      // 非公開変数 privateVar
    protected int protectedVar = 3; // 半公開変数 protectedVar
    internal int internalVar = 4;    // 内部変数 internalVar
}

public class Child : Parent
{
    public void TestAccess()
    {
        Console.WriteLine(publicVar);    // ✓ 公開変数
        // Console.WriteLine(privateVar); // ✗ 非公開変数
        Console.WriteLine(protectedVar); // ✓ 半公開変数
        Console.WriteLine(internalVar);  // ✓ 内部変数
    }
}
```

ملخص الإجابات / Summary of Answers

السؤال	الإجابة
Q1	OOP is a programming paradigm based on objects (explained above)
Q2	Complete C# program with array and foreach loop (code above)
Q3.1	d. float [] x = new float [3];
Q3.2	c. public static void Sum(params int[] k1, params int[] k2)
Q3.3	c. 01345789
Q3.4	a. A() { }
Q3.5	c. protected

نصائح للمبتدئين / Tips for Beginners

1. اكتب الكود بنفسك ولا تكتفي بالكود :
 - Practice coding: Write code yourself, don't just read
2. بيئة تطوير ممتازة للمبتدئين: **Visual Studio** استخدم :
 - Use Visual Studio: Excellent IDE for beginners
3. غير القيم وشاهد النتائج :
 - Try examples: Change values and see results
4. تساءل عن أخطاءك :
 - Read error messages: They help you understand problems
5. المتغيرات، الحلقات، الشروط، ثم انتقل إلى المواضيع المتقدمة :
 - Start with basics: Variables, loops, conditions, then move to advanced topics

تم إعداد هذا الملف بواسطة / **Prepared by:** GitHub Copilot التاريخ / **Date:** December 6, 2025

Good luck with your studies! / حظاً موفقاً في دراستك