Oop quiz answers:

Quiz week 1

* A class is like a template or blueprint for object creation
* Objects encapsulate both **functionality and data**
* **String** is used to store a telephone number
* A constructor has the same name as that of class and it does not have any return type is **FALSE**
* **Property** is added to a class in order to provide access to hidden data
* **Constructor** is used to initialize an object when created
* OOP offers means of managing complexity for larger software through breaking the problems down into small parts as objects
* **Abstraction** is the process of identifying the essential aspect of an real-world entity and ignoring unimportant properties
* **Encapsulation** prevents direct manipulation of the objects information
* **Attribute** refers to the characteristics of an object
* **Behavior** refers to the action that an object is capable of performing
* Add **private fields** to a class to store the things the object knows without enabling objects of other classes know about them
* If a class does not have any constructor explicitly defined, **A default construct is provided by .Net framework**
* **General** is not a valid specifier for a field/method

Quiz week 2

* What is the primary purpose of using UML in .NET 8 application development?
  + Ans: To visualise and document the design of the application
* When should you prefer using a Dictionary than a List in C#?
  + Ans: When you need lookups based on unique keys
* Unit tests are only useful after the application has been developed
  + Ans: False
* Which of following best describes .NET Base Class Library (BCL)?
  + Ans: It provides optimised code for common tasks like file handling, database collection, and drawing utilities
* What does Assert.AreEqual(expected,actual) do in unit testing?
  + Ans: Verify that the expected and actual values are equal
* Which one is the correct tagging annotation used to indicate that the class is a unit testing class?
  + Ans: [Test()]
* What is the primary purpose of the Assert class in .Net unit testing?
  + Ans: To format the output
* What is the Base Class Library (BCL) in the .NET Framework?
  + Ans: A set of reusable classes, interfaces, and value types to provide fundamental functionalities
* In a UML Class Diagram for a .NET 8 application, which of the following is the correct symbol for a private attribute?
  + Ans: A minus sign (-) before the attribute name
* Unit Test is used to verify that object functionality/functions work as expected.
  + Ans: True
* Which of following is correct way t add items into a List in C#?

List<int> numbers = new List<int>();

* Ans: numbers.Add(10)
* Which one is the correct unit test packages used in C#?
  + Ans: NUnit
* Fill in the missing words

The correct steps for the creation of unit test are:

* \_\_\_\_\_\_\_ the Test
* \_\_\_\_\_\_\_ the operation
* \_\_\_\_\_\_\_ the results
  + Ans: Setup, Perform, Check
* In UML, which diagram would you use to represent the detailed structure of a class with its methods and attributes in a .NET 8 application
  + Ans: Class Diagram
* What is the primary purpose of structure diagrams in UML for .NET 8 applications?
  + Ans: To describe the static structure and relationships of the system
* Only one single test can be written for each class.
  + Ans: False
* Why is early error detection important in software development?
  + Ans: It reduces the cost and efforts needed to fix bugs
* Which Assert method is best for verifying boolean conditions?
  + Ans: Assert.IsTrue()
* Given the following C# code snapshots. Is the test passed or fail?

[Test] public void TestDivision { bool isEven = (7%2==0); Assert.IsTrue(isEven, “The number is even”); }

* Ans: Fail

Quiz week 3:

* **In the context of a game, which of the following is an example of an association relationship?** **Ans:** A Game object has a reference to a Player object
* **What is the primary purpose of a UML Sequence Diagram?** **Ans:** To visualize the dynamic interactions between objects over time
* **When passing an object to a method in C#, what is actually passed?** **Ans:** A reference to the object's memory on the heap
* **In which following scenario would an aggregation relationship be most appropriate?** **Ans:** A Library object that contains multiple Book objects

**Consider the following C# code. Where is the Person object stored in memory?** csharp

CopyEdit

public class Person {

    private string Name;

}

Person p = new Person();

p.Name = "Harry Porter";

* **Ans:** p is a reference on the stack, and the Person object is on the heap
* **Which of the following best represents a composition relationship, which is a stronger form of aggregation?** **Ans:** A Book object contains Chapter objects that cannot exist without the book
* **In C#, which of the following data types is stored directly on the stack?** **Ans:** int
* **What is the main difference between reference types and value types in C#?** **Ans:** Reference types store references to data on the heap, while value types store the data directly
* **What happens when a method finishes executing in C#?** **Ans:** The stack frame for that method is removed, and control returns to the caller
* **Which of the following best describes the garbage collector in C#?** **Ans:** It automatically manages memory for objects on the heap
* **Which of the following best describes a dependency relationship?** **Ans:** A short-term interaction where one object temporarily uses another

**In the following code snapshot, what type of relationship exists between BlackjackGame and Deck?** csharp

CopyEdit

public class BlackjackGame {

    private Deck deck;

    public BlackjackGame() {

        deck = new Deck();

    }

}

* **Ans:** Association
* **In an OOP program, why is it important to identify object relationships early in the software design process?** **Ans:** It saves implementation efforts and reduces potential refactoring

Quiz week 4:

* Which of the following best describes polymorphism?
  + **Ans:** A printer that can print different documents, images, or PDFs
* Which of the following represents the concept of specialization in object-oriented programming?
  + **Ans:** Create a Rectangle class that inherits from Shape and adds Width and Height attributes
* \_\_\_\_\_\_ enables the use of same method name with different implementation for child classes
  + **Ans:** Polymorphism
* Which of the following statements about polymorphism is FALSE?
  + **Ans:** You can achieve polymorphism without inheritance in C#
* Which of the following best describes runtime polymorphism?
  + **Ans:** When the method to be executed is determined at runtime
* Which following statement is FALSE about inheritance in C#?
  + **Ans:** A class can inherit from multiple parent classes simultaneously
* Which of the following best describes method overriding?
  + **Ans:** Providing a new implementation of an inherited method in a child class
* Which of the following is the correct inheritance declaration by derived/child classes in C#?
  + **Ans:** public class Circle:Shape
* Which of the following is the correct definition of abstract method in C#?
  + **Ans:** public abstract void Print();
* In C#, what is the default access modifier for class members if none is specified?
  + **Ans:** private
* Which of the following is TRUE about polymorphism with C#?
  + **Ans:** You can refer to a child object using a parent reference
* Inheritance allows classes to inherit commonly attributes and behavior from parent classes.
  + **Ans:** True
* Which of the following statements about generalization and specialization is TRUE?
  + **Ans:** Generalization represents common characteristics shared among classes
* Given the C# code:

csharp

CopyEdit

public class Vehicle {

    public virtual void Start(){

        Console.WriteLine("Starting the vehicle!!");

    }

}

public class Car : Vehicle {

    public override void Start(){

        Console.WriteLine("Starting the car!!");

    }

}

public class Truck : Vehicle {

    public override void Start(){

        Console.WriteLine("Starting the truck!!");

    }

}

Vehicle[] vehicles = { new Car(), new Truck(), new Vehicle() };

foreach (Vehicle v in vehicles) {

    v.Start();

}

* **What will be the output?**
  + **Ans:** Starting the car  
      
     Starting the truck  
      
     Starting the vehicle
* Given the following C# classes:

csharp

CopyEdit

public class Vehicle {

    public string Brand = "Ford";

    public void Honk(){

        Console.WriteLine("Beep!!");

    }

}

public class Car : Vehicle {

    public string ModelName = "Mustang";

}

Car myCar = new Car();

Console.WriteLine(myCar.Brand);

myCar.Honk();

* **What is the correct output for the above code?**
  + **Ans:** Ford  
      
     Beep!!

Quiz week 5:

* What will the following code output?

csharp

CopyEdit

try {

    int x = 5 / 0;

} catch (DivideByZeroException){

    Console.WriteLine("Cannot divide by zero.");

} finally {

    Console.WriteLine("Execution complete.");

}

csharp

CopyEdit

- Ans: Cannot divide by zero. Execution complete.

* Which of the following is the correct way to declare an interface in C#?
  + Ans: public interface IShape {}
* Why are interfaces important in OOP?
  + Ans: They enable polymorphism and flexible interaction between unrelated classes
* Which following interface would be MOST suitable for ensuring objects can be sorted?
  + Ans: IComparable
* What is the output of the following code?

csharp

CopyEdit

public interface IAnimal {

    void Speak();

}

public class Dog : IAnimal {

    public void Speak(){

        Console.WriteLine("Woof!");

    }

}

IAnimal myDog = new Dog();

myDog.Speak();

diff

CopyEdit

- Ans: Woof!

* Which of the following is TRUE about the finally block in C#?  
    
  + Ans: It always executes, whether an exception occurs or not
* Exceptions are used to detect compiling errors.  
    
  + Ans: False
* When should exceptions NOT be used in C#?  
    
  + Ans: To control normal program flow
* How can you create a custom exception class in C#?  
    
  + Ans:

csharp

CopyEdit

class CustomException: Exception {

    public CustomException(string message) : base(message) {}

}

* Which of the following statements about interfaces and polymorphism is TRUE?  
    
  + Ans: Interfaces allow objects to be treated as their implemented interface type
* What happens if an exception is thrown inside a try block but there is no matching catch block?  
    
  + Ans: The program will crash
* What happens if a class does NOT implement all members of an interface?  
    
  + Ans: The class must be marked as abstract
* Exceptions are objects that contain an error message.  
    
  + Ans: True
* What is the output of the following code?

Quiz week 6:

* Which of the following describes ‘loose coupling’ in Responsibility-Driven Design (RDD)?  
    
  + **Ans:** Classes are independent, with minimal dependencies
* In the context of a chess game, which of the following is NOT a plausible role?  
    
  + **Ans:** For Loop
* What is the benefit of using CRC (Candidate-Responsibility-Collaboration) cards in RDD?  
    
  + **Ans:** Identifying and refining object roles and responsibilities iteratively
* Which type of UML diagram is commonly used to communicate the static structure of classes and their relationships?  
    
  + **Ans:** Class Diagram
* What is the focus of Step 3 in Responsibility-Driven Design?  
    
  + **Ans:** Collaborating with other objects to fulfill responsibilities
* Consider a Library Management System designed to handle book borrowing, returns, and user management. The User class has the responsibility to borrowBook(). Which principle of Responsibility-Driven Design (RDD) does this illustrate?  
    
  + **Ans:** Encapsulation of Responsibilities
* How does high cohesion benefit object-oriented design?  
    
  + **Ans:** Simplifies maintenance by grouping related functionality within a class
* Consider a Library Management System. How should the collaboration between “User” and “Library” be managed when borrowing a book?  
    
  + **Ans:** The Library handles the transaction, checking book availability and updating records, while the User requests the action
* Which of the following best describes a ‘role’ in Responsibility-Driven Design (RDD)?  
    
  + **Ans:** A general responsibility or purpose that an object may fulfill
* In a chess game example, which responsibility would likely belong to the ‘Piece’ class?  
    
  + **Ans:** Determining valid moves based on types (e.g., rook, bishop, and queen)
* Which UML relationship should be used to represent the permanent association between the “Library” and the “Book” classes?  
    
  + **Ans:** Aggregation (hollow diamond)
* Which of the following best represents the responsibilities of the “Book” class in the Library Management System?  
    
  + **Ans:** Track book information (title, author, availability) and update availability status