|  |  |  |
| --- | --- | --- |
| ****C# Basics and Fundamentals**** | | |
| 1. **No.** | **Topics** | **Page** |
| 1 | What is C#? | 4 |
| 2 | Explain the main features of C# | 4 |
| 3 | What is the difference between value types and reference types in C#? | 4 |
| 4 | What is a nullable type in C#? | 4 |
| 5 | Explain the concept of boxing and unboxing. | 5 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**What is C#?**

* C# is a modern, high level programming langugue developed by Microsoft.
* It’s a part of .Net ecosystem, which is free, open source, cross platform framework that supports wide range of application development

**Explain the main features of C#**

* Object Oriented Programming - (Encapsulation, Inheritance, Polymorphism and Abstraction)
* Type Safety - enforces type constrains, helps us to perform operation on the compatible type reducing the runtime error
* Garbage Collection - automatic garbage collection, automatically remove the unused object from the memory. Reduces the developer’s effort of memory management
* Rich Library Support
* Cross Platform Development

**What is the difference between value types and reference types in C#?**

* Value Types are stored in the stack memory whereas ref type are stored in head memory only it’s address is stored in stack memory
* Stack is used in pace of small and immutable data and where performance is priority
* Heap is used in place of handling complex object, to share data across multiple places and places where null need to be handled

|  |  |
| --- | --- |
| **Stack** | **Heap** |
| Generally Faster | Heap allocation takes more time |
| Cannot be null unless made nullable | Can be null |
| Copies the Value | Copies the reference |

**What is nullable type in C#?**

* In C# the compiler won’t allow you to assign null value to a variable.
* C# 2.0 provide a special feature to allow us null value to a variable that is nullable type but only for the value type
* C# 8.0 provided us the feature to assign null values for the reference types also

**Explain the concept of boxing and unboxing.**

* **Boxing -** converting a value type to reference type
* **Un-boxing -** Converting a reference type to value type
* Boxing and unboxing are expensive operation because boxing involves heap allocation and the unboxing involves casting and type checking
* Avoid boxing and unboxing in performance critical application