INFORMATION PROCESSING TECHNIQUES

Object v/s Var v/s Dynamic

WEEK 7

MURTAZA MUNAWAR FAZAL

Object v/s Var v/s Dynamic

Object	Var	Dynamic
Object was introduced with C# 1.0	Var was introduced with C# 3.0	Dynamic was introduced with C# 4.0
It can store any kind of value, because object is the base class of all type in .NET framework.	It can store any type of value, but It is mandatory to initialize var types at the time of declaration.	It can store any type of the variable, like old VB language variable.
Compiler has little information about the type.	It is type safe i.e. Compiler has all information about the stored value, so that it doesn't cause any issue at run-time.	It is not type safe i.e. Compiler doesn't have any information about the type of variable.
Object type can be passed as method argument and method also can return object type.	Var type cannot be passed as method argument and method cannot return object type. Var type work in the scope where it defined.	Dynamic type can be passed as method argument and method also can return dynamic type.

Object v/s Var v/s Dynamic

Object	Var	Dynamic
Need to cast object variable to original type to use it and performing desired operations.	No need to cast because compiler has all information to perform operations.	Casting is not required but you need to know the properties and methods related to stored type.
Cause the problem at run time if the stored value is not getting converted to underlying data type.	Doesn't cause problem because compiler has all information about stored value.	Cause problem if the wrong properties or methods are accessed because all the information about stored value is get resolve only at run time.
Useful when we don’t have more information about the data type.	Useful when we don’t know actual type i.e. type is anonymous.	Useful when we need to code using reflection or dynamic languages or with the COM objects, because you need to write less code.