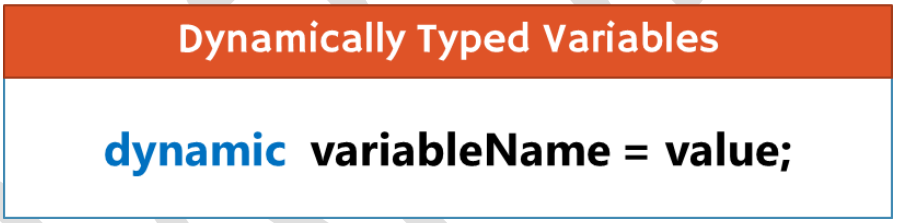
29

Dynamically Typed Variables

Understanding Dynamically Typed Variables

- Dynamically Typed Variables are the variables that are declared with 'dynamic' keyword.
- Declared without specifying the type explicitly.
- There is no fixed type for the variable.
- You can assign any type of value to these variables.
- C# compiler skips "type-checking" at compilation time; instead, it resolves the data types of its values, at run time.



- The "dynamic" type variables are converted as "object" type in most cases.
  - dynamic dynamicVariable = 100; →à object dynamicaVariable = 100;
- The Dynamically Typed Variable can change its data type, any no. of times, at run time.
- Methods and other members of 'dynamically typed variables' will not be checked by the compiler at compilation time; will be checked by CLR at run time.
  - If the method or other member not available, it would not cause compile-time error; it raises run-time error, when the execution flow encountered that particular statement.
  - dynamicVariable.NonExistingMethod( ); //run-time error (exception)
- The Dynamically Typed Variables need not be initialized, while declaration.

- The Dynamically Typed Variable doesn't have "Intellisense" in Visual Studio.
- The "dynamic" keyword is allowed for local variables, method parameters, fields, properties, return types etc.