|  |  |
| --- | --- |
|  | **Agenda** |
|  | What is C#.NET? |
|  | Why do we need? |
|  | How to write a C# program? |
|  | Programs/Demo |

|  |  |
| --- | --- |
|  | **What is C#.NET?** |
|  | C# is an object oriented programming language from Microsoft.  C# is pronounced as C-Sharp.  It is used to develop software applications for business (high level) targets to .NET technologies.  C# is intended to be a simple, modern, general-purpose, object-oriented programming language.  Its development team is led by **Anders Hejlsberg**. The most recent version is C# 5.0, which was released on August 15, 2012.  It is a PURE OOPS language (fully secured), C# is the best compotator for java   |  |  |  | | --- | --- | --- | |  | **Rules to write C# program (C# follows C and C++ syntax)** | | | **SNo** | **Rule** | **Description** | | 1 | Case Sensitive | Lower case keywords are different from upper case keywords | | 2 | Statement termination | Each and every statement must terminate using ; | | 3 | Main() method | Program startup should be in Main() method |   **Points to note:**  It is a high level language, used to develop application programs for business purpose.  C# uses csc compiler as a translator.  **C# Compiler:** csc is a compiler, it converts C# code to IL Code(.NET under stable code)C# and C#.NET are different  C# is pure oops language  C#.NET is a language, which targets to .NET technology.  C#.NET is a language understand .net framework.  C#.NET is built-in with Visual Studio.  **Points to Note:**  More than 60% of clients are looking .net programmers to use C#.NET |

|  |  |
| --- | --- |
|  | **What is a project?** |
|  | A project is a complete software application.  A project is with full user requirements with process.  A project can be a any kind of application like console application, windows application, mobile application, library, service, web application, website, web service, etc.,   |  |  |  | | --- | --- | --- | |  | **Programming language** | **Project extension** | | 1 | VC#.NET | .csproj | | 2 | VB.net | .vbproj | | 3 | VC++.net | .vcxproj | |

|  |  |
| --- | --- |
|  | **What is a Solution?** |
|  | In .NET, a solution is a collection of projects.  A solution extension will be .sln  **Points to Note:**  When a project created using visual studio, it exist inside a solution.  Solution is optional ( it is strongly recommended to have project inside solution) |

|  |  |
| --- | --- |
|  | **Working with Console Application** |
|  | |  |  | | --- | --- | |  | **What is a Console Application?** | |  | Console Application is instruction based application.  Used to implement application logic. |  |  |  | | --- | --- | |  | **Why do we need Console Application?** | |  | Console Application are powerful.  Console Application doesn’t required to install, doesn’t need to load. This application can be invoked by calling its name. |  |  |  | | --- | --- | |  | **Who will use this kind of application?** | |  | Administrators will use this kind of application. This application can interact with requirement directly. |  |  |  | | --- | --- | |  | **Points to note:** | |  | Each and every instruction related to C#, can be learn using console application. It is strongly recommended to learn console application. | |

|  |  |
| --- | --- |
|  | **Software to down visual studio** |
|  | Link for Visual Studio Express 2013 for Windows Desktop  <http://www.visualstudio.com/en-us/products/visual-studio-express-vs.aspx>  **Using above software we can develop following application**   1. Console Application 2. Class libraries 3. Windows Applications 4. Windows Services   The above link used to download visual studio express, allowed to use only for 30days.  You can even search for Ultimate edition, allowed to used 90 days without key. |

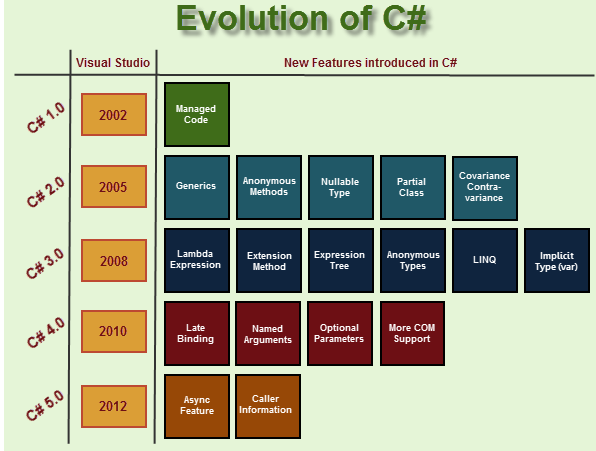
|  |  |
| --- | --- |
|  | **How to create console Application in Visual Studio?** |
| Step1 | Open visual studio |
| Step2 |  |
| Step3 |  |

|  |  |  |
| --- | --- | --- |
|  | **Structure of Console Application Program** |  |
|  | **Snapshot** | **Structure of C#.NET program** |
|  |  | using System;  namespace YourProjectName  {  class Program  {  static void Main()  {  Statements;  }  }  } |

|  |  |  |
| --- | --- | --- |
|  | **Keywords** | **Functionality** |
| 1 | using | This keyword is used import a library into current program.  Example:  using System;  using System.Data;  using System.IO;  using System.Data.SqlClient; |
| 2 | namespace | namespace keyword is used to group the classes (this is optional while creation of Console Application)  namespace behave like boundary for existing project.  Note: By default a namespace will be created with the name given for the project (it is optional)  When the project is created, a namespace gets added automatically with project name. |
| 3 | class | This keyword is used to create a class  (A class is a user defined data type)  Note: a program can have any number of classes.  There should be one class with name Program and it should contain Main() method |
| 4 | Main | This is the startup of .net program  **Note**: Main is case sensitive |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Language** | **C (or) c++** | **C#** | **Java** |
|  | To import a library | #include | using | import |
|  | Start up | main() | Main() | main(String[]args) |
|  | Mandatory libraries | stdio.h (or) iostream.h | System | Java.lang.\* |
|  | Library Terms | Header file/library file | Libraries/namespaces | Packages |
|  | File extension | .c, .cpp | .cs | .java |
|  | Object code | .obj | .il | .class |

|  |  |  |
| --- | --- | --- |
|  | **Operators used to regular programs** | |
| **SNo** | **Operators** | **Purpose** |
| 1 | + | Concatenation of strings and addition of numbers |
| 2 | () | Parenthesis, used to pass arguments |
| 3 | ; | Statement termination |
| 4 | " | String substitution |
| 5 | ' | Character substitution |
| 6 | = | Assignment operator |
| 7 | { } | To specify block |
| 8 | [] | Array notation |
| 9 | . | Separator for method and object |
| 10 | : | Inheritance operator  Identifier used case |
| 11 | , | Continuation operator |
| 12 | => | Lambda expression operator |
| 13 | ?: | Ternary operator |
| 14 | \* | Pointer |
| 15 | & | Reference in pointer |
| 16 | ~ | Destructor beginning |
| 17 | < > | Angular braces (refer Generics) |
| 18 | @ | Verbatim literal |
| 19 | -> | Accessible instance variable using pointer object |
| 20 | \ | Escape sequence |
| 21 | // | Single line comment |
| 22 | /\* \*/ | Multi line comments |
| 23 | /// | XML Comments |
| 24 | # | Pre processor statement |



|  |
| --- |
| **Structure/Syntax of C# program** |
| using Library;  class Program  {  static void Main()  {  statements;  }  } |

|  |  |  |
| --- | --- | --- |
| **Sl.no** | **Comment** |  |
| 1 | Single line comment | // To declare a variable |
| 2 | Multi line comment | /\*  This code belongs to  read data from user  \*/ |
| 3 | Xml comment | /// xml comments |

|  |  |
| --- | --- |
|  | **System namespace?** |
|  | It is a base namespace of all libraries in the .net  It contains base requirements like Console Class, Convert Class, String class, data types, etc.,  using System; |

|  |  |
| --- | --- |
|  | **Working with Console class** |
|  | **This class contains methods to handle console input and console output operations.**  **Console class a readymade(ready to use) class exists in System namespace;**   |  |  |  | | --- | --- | --- | | **Sno** | **Method** | **Purpose** | | **1** | **WriteLine()** | This method is used to display text in the Console output and move the cursor to next line.  Console.WriteLine("welcome to C#"); | | **2** | **Write()** | Used to display the text and cursor wait in the same line. | | **3** | **ReadLine()** | Used to read string in console (group of letters) | | **4** | **ReadKey()** | Used to read choice from the user.  Similar getch() in c-programming  Used to pause the output  Console.ReadKey(); |   **Points to note:**  **Console is a static class ( it will not allow to create object )** |

|  |  |
| --- | --- |
|  | **Example1: Display welcome message** |
|  |  |

Short cut key to run/execute program: F5

By pressing Ctrl+F5 we need not to use Console.ReadKey()

|  |  |
| --- | --- |
|  | **Example2: Program to display welcome message and move cursor to next line** |
|  | using System;  class Program  {  static void Main()  {  Console.WriteLine("Welcome to C#.NET");  Console.ReadKey();  }  } |

|  |  |
| --- | --- |
|  | **Working with String class** |
|  | String class (reference type existing in CTS) exist in system namespace; Used to create an object, used to store and manipulate string (Group of letters).  String class contains methods used to handle string.   |  |  | | --- | --- | |  | **To declare string object** | | **Syntax:** | String objectname; | | **Example:** | String name; |  |  |  | | --- | --- | |  | **To assign value into string object:** | | Declare variable  and Assign value | String name;  name="Kiran"; | | **To initialize string:** | String name="Praneet"; | | **To display string** | Console.WriteLine(name); |   **To read a string:**  string name = Console.ReadLine();  **Points to Note:**  String is not a data type, it is a class.  This class available in System namespace  String is immutable.  string is an alias String class |

|  |  |
| --- | --- |
|  | **Example to display string exist in a variable** |
|  |  |

|  |  |
| --- | --- |
|  | **Example2: To display string** |
|  | using System;  class Program  {  static void Main()  {  string name = "Praneet";  Console.WriteLine("Welcome " + name);  Console.ReadKey();  }  } |

|  |  |
| --- | --- |
|  | **Example3: Difference between Write and WriteLine methods** |
|  | using System;  class Program  {  static void Main()  {  Console.Write("Cursor in the same line ");  Console.ReadKey();  Console.WriteLine("Cursor in the next line ");  Console.ReadKey();  }  } |

|  |  |
| --- | --- |
| **Property** | **Purpose** |
| **String.Empty** | This property is used to assign empty value to a string.  It is strongly recommend to assign empty value when you initialization of string  String.Empty is the static property. |
| Length | This property returns the length of the characters in a string. |

|  |  |
| --- | --- |
|  | **Example assign empty value to a string** |
|  |  |

|  |  |
| --- | --- |
|  | **Example4: Read your name into string variable and display the given name** |
|  | using System;  class Program  {  static void Main()  {  string name = string.Empty;    Console.Write("Enter your name : ");  name = Console.ReadLine();    Console.WriteLine("Given Name is : "+ name);  Console.ReadKey();  }  } |

|  |  |
| --- | --- |
|  | **String class methods:** |
|  | String class contains both static and dynamic methods.  These are ready made definition in string class, used to manipulate the value inside string object.  To perform regular manipulations on strings, the following methods are used.   |  |  |  | | --- | --- | --- | | **SNo** | **Method** | **Purpose** | | 1 | **ToLower()** | Used to retrieve the given string into small letters  String name="Kiran Kumar";  Console.WriteLine(name.ToLower()); | | 2 | **ToUpper()** | Used to retrieve the given string into capital letters  String name="Kiran Kumar";  Console.WriteLine(name.ToUpper()); | | 3 | **CompareTo()** | This method is used to compare two strings and return difference value.  Return 0 when both are same.  string name1 = "kiran";  string name2 = "kiran";  if ( name1.CompareTo(name2)==0)  Console.WriteLine("Both are same");  else  Console.WriteLine("Not same"); | | 4 | **IndexOf()** | This is the method used to find the position of the string inside a string.  Syntax:  Str.IndexOf("substring");  Example:  String name="Fernandes";  Int x= name.IndexOf("d");  Note: This method return the position of first occurance. | | 5 | **Substring()** | This method is used to fetch string inside a string.  **Syntax:**  str.Substring(positionFrom,[positionTo]);  **Points to note:**  The second parameter is optional. | |  |  |  | | 6 | **Trim()** | This function trim the empty space of preceding and forwarding place. | | 7 | **Contains()** | This function return true when the substring exist inside a string. | | 8 | **Split()** | To split string into array | | 9 | **Remove()** | Used to remove specified number of letters | | 10 | **LastIndexOf()** | Used to return last index of specified letter | |

|  |  |
| --- | --- |
|  | **String Manipulations:** |
|  | using System;  namespace ConApp1  {  class Program  {  static void Main()  {  string name = "Kiran Kumar";  Console.WriteLine("Length is : "+ name.Length);  Console.WriteLine("Upper case : " + name.ToUpper());  Console.WriteLine("Lower case : " + name.ToLower());  Console.WriteLine("Data type is :" + name.GetType());  Console.ReadKey();  }  }  } |

|  |  |
| --- | --- |
|  | **Compare two strings:** |
|  | static void Main()  {  string name1 = "kiran";  string name2 = "kiran";  int x=name1.CompareTo(name2);  if ( x==0 )  Console.WriteLine("Both are same " +x);  else  Console.WriteLine("Not same");  } |

|  |  |
| --- | --- |
|  | Example: To compare 2 string in 2 variables |
|  |  |

|  |  |
| --- | --- |
|  | **Program to check the uname and password are correct or not** |
|  |  |

|  |  |
| --- | --- |
|  | **Locating position of string in a string** |
|  | using System;  namespace ConApp1  {  class Program  {  static void Main()  {  string name1 = "kiran";  Console.WriteLine(name1.IndexOf("i")); // index start from 0  name1 = "kiran kumar";  Console.WriteLine(name1.IndexOf("kumar"));  Console.ReadKey();  }  }  } |

|  |  |
| --- | --- |
|  | **Program to display first space and last space of letter in a given string** |
|  |  |

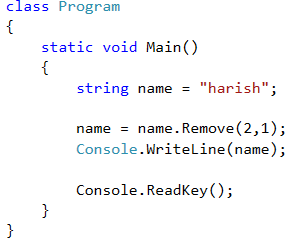
|  |  |
| --- | --- |
|  | **To display the position of space:** |
|  | string name = "Kiran Kumar";  Console.WriteLine(name.IndexOf(" "));  int sp = name.IndexOf(" ");  Console.WriteLine(sp); |

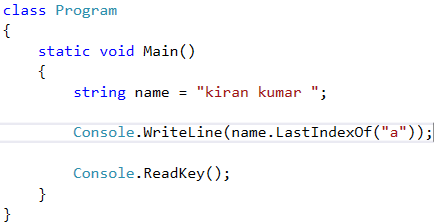
|  |  |
| --- | --- |
|  | **Workign with Substring() method** |
|  | This method is used fetch string existing inside a string.  First parameter refers to starting position of a string  Second parameter refers to number of letters to be displayed(optional)   |  |  | | --- | --- | |  | **Display first five letters** | |  | static void Main()  {  string name = "Kiran Kumar";  Console.WriteLine(name.Substring(0, 5));  } |  |  |  | | --- | --- | |  | **Display seven letters from 4th letter** | |  | static void Main()  {  string name1 = "Swarna Bonam";  Console.WriteLine(name1.Substring(3, 7));  } | |

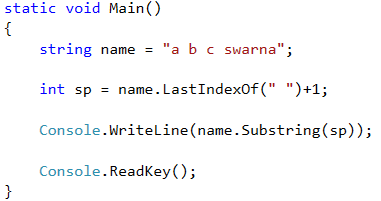
|  |  |
| --- | --- |
|  | **Display all letter from 5th letters** |
|  | static void Main()  {  string name1 = "Saeed Answar";  Console.WriteLine(name1.Substring(5));  } |
|  | **To display the first name in the fullname** |
|  | string name1 = "amitabh bachan";  int sp = name1.IndexOf(" ");  Console.WriteLine(name1.Substring(0,sp )); |

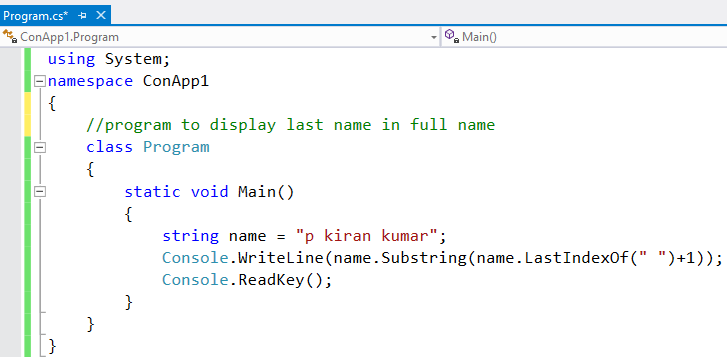
|  |  |
| --- | --- |
|  | **Display first name in the fullname** |
|  | string fullName = "Rajendra Prasad"; Console.WriteLine(fullName.Substring(0, fullName.IndexOf(" "))); |

|  |  |
| --- | --- |
|  | **Program for backspace code** |
|  | using System;  namespace ConApp1  {  class Program  {  static void Main()  {  // to remove last letter in the string  string msg = "kiran kumar";  msg = msg.Substring(0, msg.Length - 1);  Console.WriteLine(msg);  Console.ReadKey();  }  }  } |









|  |  |
| --- | --- |
|  | **Remove the last comma in the string list** |
|  | static void Main(string[] args)  {  string[] names = { "kiran", "Anitha", "nikil", "chocky" };  string st=string.Empty;  foreach (string name in names)  st += name + ", ";    int l = st.LastIndexOf(",");  st = st.Remove(l);  Console.WriteLine(st);  Console.ReadKey();  } |

|  |  |
| --- | --- |
|  | Display count of word occurances: |
|  | class Program  {  static void Main(string[] args)  {  string st = "Eat .net, drink .net, sleep .net, cause developing app in .net or mvc rocks, kiran";  int wc = 0;  foreach (Match m in Regex.Matches(st,".net"))  wc++;  Console.WriteLine("Word repeated for {0} times",wc);  }  } |

|  |  |
| --- | --- |
|  | Display count of word occurances: |
|  | using System;  using System.Text.RegularExpressions;  class Program  {  static void Main()  {  string st = "Eat .NET, drink .net, sleep .net, cause developing app in .net or mvc rocks, kiran";  int wc = 0;  foreach (Match m in Regex.Matches(st,".net",RegexOptions.IgnoreCase))  wc++;    Console.WriteLine("Word repeated for {0} times",wc);  }  } |

|  |  |
| --- | --- |
|  | **Interview questions** |
| 1 | What is the difference between static and dynamic methods? |
| 2 | Is string a datatype or class? |
| 3 | What is the difference between class and static class? |
| 4 | What is the difference between string.Compare and CompareTo() methods? |