**Part A: Ex No: 1- Accept a character and check the case of**

**Character**

**Aim:**

Accept a character from console and check the case of the character.

**Pre-Learning Skills**

* Knowing the templates installed and version of visual studio used
* Understanding the different types of application in visual studio
* How to save the application in their own directories
* Identifying the commands to build and run the solutions

**Procedure:**

1. Open a new Project.
2. Select the console application from the templates and locate the path to store it.
3. Print title or heading for the program
4. Accept a character using ReadLine () function.
5. Check for the character case by its ASCII value.
6. Save and execute the code

**Source Code: LP\_1 –** To Change Case using built in methods

1: using System;

2:

3: namespace Lab\_Practice\_1\_1

4: {

5: internal class LP1\_1

6: {

7: static void Main(string[] args)

8: {

9: string input\_str;

10: char input\_char;

11:

12: Console.WriteLine("Enter a single charecter:");

13: input\_str = Console.ReadLine(); // Read String from user

14:

15: if (input\_str.Length != 1) // Check if exactly 1 char entered

16: {

17: Console.WriteLine("ERROR: Enter exactly 1 charecter only.");

18: }

19: else

20: {

21: input\_char = char.Parse(input\_str); // Convert string to char

22:

23: Console.WriteLine("Given character is: " + input\_char);

24:

25: if (Char.IsLower(input\_char))

26: Console.WriteLine("The entered character is Lowercase.");

27:

28: else if (Char.IsUpper(input\_char))

29: Console.WriteLine("The entered character is Uppercase.");

30:

31: else

32: Console.WriteLine("The entered character is not an alphabet.");

33: }

34:

35: Console.WriteLine("Press any key to continue . . .");

36: Console.ReadKey();

37: }

38: }

39: }

**Source Code: LP1\_2 –** To Change Case using User Defined Function

1: using System;

2:

3: namespace LP1\_2

4: {

5: class Program

6: {

7: static void Main(string[] args)

8: {

9: Console.Write("\n\nProgram to Check whether a character is lower case or

upper case :\n");

10: Console.Write("--------------------------------------------------------\n");

11: Console.Write("Input a Single character : ");

12: char ch = (char)Console.Read();

13: // Call of Function

14: checkCase(ch);

15: Console.ReadKey();

16: }

17: //Function Definition

18: static void checkCase(char ch)

19: {

20: if (ch >= 'A' && ch <= 'Z')

21: Console.WriteLine("\n" + ch +

" is an UpperCase character");

22: else if (ch >= 'a' && ch <= 'z')

23: Console.WriteLine("\n" + ch +

" is an LowerCase character");

24: }

25: }

26: }

27:

**Source Code: Ex\_1**

1: using System;

2:

3: namespace Ex\_1

4: {

5: class Program

6: {

7: static void Main(string[] args)

8: {

9: Console.Write("Program to check whether a character is

lower or upper case.\n");

10: Console.Write("----------------------------------------

--------\n");

11: Console.Write("Input a character: ");

12: char ch = (char)Console.Read();

13:

14: if (Char.IsUpper(ch))

15: {

16: Console.WriteLine("\nThe character is

uppercase.\n");

17: }

18: else if(Char.IsLower(ch))

19: {

20: Console.WriteLine("\nThe character is

lowercase.\n");

21: }

22: else

23: {

24: // User has entered a symbol or number etc. Example: 1,2, #, ?...

25: Console.WriteLine("\nThe character is not an

alphabet.\n");

26: }

27: Console.ReadKey();

28: }

29: }

30: }

**Explanation:**

1. ReadLine ( ) function works similar to scanf ( ) function. Waits for the user until an input is given from the keyboard

2. Write ( ) and WriteLine ( ) functions work similar to printf ( ) function

3. Readkey ( ) accepts one character similar to getch( ) function

**Output:** (Write down the screen outputhere)

**Result:**

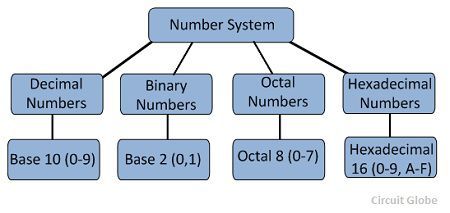
The above program \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ was successfully executed and then the output was verified under \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**Viva Questions**

1. List out the decision making system

* Simple if statement
* if … else
* else…if ladder
* switch…case

1. Number System



1. What is ASCII value?

The ASCII decimal (Dec) number is created from binary, which is the language of all computers. For an example, the lower case "h" character (char) has a decimal value of 104, which is "01101000" in binary.

1. **How will you read numeric data using Readline( )?**

ReadLine ( ) function returns a ‘string’, so in case we want to work with an integer number we have to convert the string to integer by using Convert.ToInt16 (string).

**Example:**

Console.Write("Please Enter your age: ");

Age = Convert.ToInt16(Console.ReadLine());

Console.WriteLine("User Age: " + Age);

**Skill Practice;**

* Write down the American Standard Code for Information Interchange(ASCII) value for the string HELLO

|  |  |  |
| --- | --- | --- |
| Letter | Decimal Value(ASCII) | Binary Value for the letter |
| H |  |  |
| E |  |  |
| L |  |  |
| L |  |  |
| O |  |  |

**Part A: Ex No: 2 - Accept any character and Display Vowel or Not**

**Aim:**

Develop a Vb.Net program to accept any one character from keyboard and display whether it is vowel or not.

**Pre-Learning Skills**

* Understanding the vowels and consonants
* Must know the Relational and Logical Operators available

**Procedure:**

1. Open a new Project.
2. Select the console application from the templates and locate the path to store it.
3. Accept a character using ReadLine () function.
4. Check for the character if it belongs to,’ A’, ‘E’,’I’,’O’,’U’.
5. Save and execute the code.

**Source Code: LP\_2\_1**

1: using System;

2: namespace LP2\_1

3: {

4: class Program

5: {

6: static void Main(string[] args)

7: {

8: string input\_str;

9: char input\_char;

10: Console.WriteLine("Enter a Single Character");

11: // Reads an input from user

12: input\_str = (Console.ReadLine());

13: // Checks if user entered input is 1 character

14: if (input\_str.Length != 1)

15: Console.WriteLine("Given Input is " + input\_str);

16: else

17: {

18: // Converts string to a Single Character

19: input\_char = char.Parse(input\_str);

20: Console.WriteLine("Given Character is " +

input\_char);

21: // Condition for vowel checking

22: if (input\_char == 'a' || input\_char == 'e' ||

input\_char == 'i' || input\_char == 'o' ||

input\_char == 'u' || input\_char == 'A' || input\_char == 'E' || input\_char == 'I' || input\_char == 'O' || input\_char == 'U')

23: Console.WriteLine("Entered Character '" +

input\_char + "' is Vowel.");

24: else

25: Console.WriteLine("Entered Character '" +

input\_char + "' is Not a Vowel.");

26: }

27: Console.ReadKey();

28: }

29: }

30: }

31:

**Source Code: LP\_2\_2**

1: using System;

2:

3: namespace LP2\_2

4: {

5: class Program

6: {

7: static void Main(string[] args)

8: {

9: Console.WriteLine("Enter an Alphabet");

10: char ch = Convert.ToChar(Console.ReadLine().ToLower());

11: switch (ch)

12: {

13: case 'a':

14: Console.WriteLine("It is vowel");

15: break;

16: case 'i':

17: Console.WriteLine("It is vowel");

18: break;

19: case 'o':

20: Console.WriteLine("It is vowel");

21: break;

22: case 'u':

23: Console.WriteLine("It is vowel");

24: break;

25: case 'e':

26: Console.WriteLine("It is vowel");

27: break;

28: default:

29: Console.WriteLine("It Is Not Vowel");

30: break;

31: }

32: Console.ReadKey();

33: }

34: }

35: }

**Source Code: LP2\_3**

1: using System;

2:

3: namespace LP2\_3

4: {

5: class Program

6: {

7: static void Main(string[] args)

8: {

9: string vowels = "aeiouAEIOU";

10: // get char input from user

11: Console.WriteLine("Enter One Character");

12: // Call of Function

13: string char\_to\_check = get\_one\_char();

14: //Print given caharacter

15: Console.WriteLine(char\_to\_check);

16: // Checks if character is in string

17: if (vowels.Contains(char\_to\_check))

18: Console.WriteLine("Given Character is Vowel");

19: else

20: Console.WriteLine("Given Character is Not Vowel");

21:

22: Console.ReadKey();

23: }

24: static string get\_one\_char() // Function definition

25: {

26: string inp\_str = Console.ReadLine();

27: char c = inp\_str[0];

28: return (inp\_str);

29: }

30: }

31: }

32:

**Source Code: Ex\_2**

1: using System;

2:

3: namespace Ex\_2

4: {

5: class Program

6: {

7: static void Main(string[] args)

8: {

9: string vowels = "aeiouAIEOU";

10: string input\_str;

11:

12: Console.WriteLine("Program to check if charecter is Vowel.");

13: Console.WriteLine("Enter a Single Character:");

14: // Reads input string

15: input\_str = Console.ReadLine();

16:

17: // Check if string has single charecter only

18: if (input\_str.Length == 1)

19: {

20: Console.WriteLine("Given Character is " + input\_str);

21: // check if given charecter is present in "aeiouAEIOU" string

22: if (vowels.Contains(input\_str))

23: Console.WriteLine(" Entered Character is Vowel.");

24: else

25: Console.WriteLine(" Entered Charecter is not vowel.");

26: }

27: else

28: Console.WriteLine("ERROR:Enter 1 charecter only.");

29:

30: Console.WriteLine("Press any key...");

31: Console.ReadKey();

32: }

33: }

34: }

35:

**Output:** (Write down the screen outputhere)

**Result:**

The above program \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ was successfully executed and then the output was verified under \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

**Viva Questions**

**List out some of the various string operations and functions used in c#**

|  |  |
| --- | --- |
| **String Functions** | **Definitions** |
| CompareTo() | Compare two strings and returns integer value as output.  It returns 0 for true and 1 for false. |
| Contains() | The C# Contains method checks whether specified character or string is exists or not in the string value. |
| EndsWith() | This EndsWith Method checks whether specified character is the last character of string or not. |
| Equals() | The Equals Method in C# compares two string and returns Boolean value as output. |
| ToLower() | Converts String into lower case based on rules of the current culture. |
| ToUpper() | Converts String into Upper case based on rules of the current culture. |
| Length | It is a string property that returns length of string. |
| Substring() | This method returns substring. |
| Trim() | It removes extra whitespaces from beginning and ending of string. |

**Part A: Ex No: 3 - To implement a Calculator with memory**

**and recall operations.**

**Aim:**

To develop a Vb.Net program to implement a calculator with Memory and Recall operations.

**Pre-Learning Skills**

* Understanding the Integrated Development Environment(IDE)
* Understanding the different input controls from the tool box
* Learning the basic mathematical formulas and arithmetic operations
* Usage of control flow statements (switch..case)

**Procedure:**

1. Open a New project.

2. Choose the windows application from the Templates.

3. Design the Form as a calculator by drag and drop array of button control for numbers & operator symbol and textbox control for inputs

4. Change the Form Caption to the Title of the program

5. Drag and Drop required Controls from the tool box and change the name and text properties.

6. Place separate buttons for arithmetic and memory operations.

7. Write the following code in the click () event of the button.

8. Save and Execute.

**Source Code: LP\_3\_1-** Simple Arithmetic Operations

1: using System;

2:

3: namespace LP3\_1

4: {

5: class Program

6: {

7: static void Main(string[] args)

8: {

9: Console.WriteLine("Enter the action to be performed");

10: Console.WriteLine("Press 1 for Addition");

11: Console.WriteLine("Press 2 for Subtraction");

12: Console.WriteLine("Press 3 for Multiplication");

13: Console.WriteLine("Press 4 for Division \n");

14: int action = Convert.ToInt32(Console.ReadLine());

15:

16: Console.WriteLine("Enter 1st input");

17: int input\_1 = Convert.ToInt32(Console.ReadLine());

18:

19: Console.WriteLine("Enter 2nd input");

20: int input\_2 = Convert.ToInt32(Console.ReadLine());

21:

22: int result = 0;

23: switch (action)

24: {

25: case 1:

26: {

27: result = Addition(input\_1, input\_2);

28: break;

29: }

30: case 2:

31: {

32: result = Subtraction(input\_1, input\_2);

33: break;

34: }

35: case 3:

36: {

37: result = Multiplication(input\_1, input\_2);

38: break;

39: }

40: case 4:

41: {

42: result = Division(input\_1, input\_2);

43: break;

44: }

45: default:

46: Console.WriteLine("Wrong action!! try again");

47: break;

48: }

49: Console.WriteLine("The result is {0}", result);

50: Console.ReadKey();

51: }

52: // Addition

53: public static int Addition(int input\_1, int input\_2)

54: {

55: int result = input\_1 + input\_2;

56: return result;

57: }

58: // Substraction

59: public static int Subtraction(int input\_1, int input\_2)

60: {

62: int result = input\_1 + input\_2;

63: return result;

64: }

65: // Multiplication

66: public static int Multiplication(int input\_1, int input\_2)

67: {

68: int result = input\_1 + input\_2;

69: return result;

70: }

71: // Division

72: public static int Division(int input\_1, int input\_2)

73: {

74: int result = input\_1 + input\_2;

75: return result;

76: }

77: }

78:}

79:

**Source Code: Ex\_3**

1: using System;

2: using System.Windows.Forms;

3:

4: namespace Ex\_3

5: {

6: public partial class Form1 : Form

7: {

8: public Form1()

9: {

10: InitializeComponent();

11: }

12:

13: double operand1,operand2;

14: string operation = string.Empty;

15: double memory, result = 0.0;

16:

17: private void NumEvent(object sender, EventArgs e)

18: {

19: Button btn = (Button)sender;

20: txt\_result.Text += btn.Text;

21: }

22:

23: private void OperatorsEvent(object sender, EventArgs e)

24: {

25: Button btn = (Button)sender;

26: operation = btn.Text;

27:

28: if (txt\_result.Text != "")

29: operand1 = double.Parse(txt\_result.Text);

30: else

31: operand1 = 0;

32:

33: txt\_result.Text = "";

34: }

35:

36: private void btn\_clear\_Click(object sender, EventArgs e)

37: {

38: txt\_result.Clear();

39: }

40:

41: private void btn\_memory\_Click(object sender, EventArgs e)

42: {

43: if(txt\_result.Text!="")

44: memory += double.Parse(txt\_result.Text);

45: }

46:

47: private void btn\_memory\_clear\_Click(object sender, EventArgs e)

48: {

49: memory = 0;

50: }

51:

52: private void btn\_recall\_Click(object sender, EventArgs e)

53: {

54: txt\_result.Text = memory.ToString();

55: }

56:

57: private void btn\_result\_Click(object sender, EventArgs e)

58: {

59: if (txt\_result.Text != "")

60: operand2 = double.Parse(txt\_result.Text);

61: else

62: operand2 = 0;

63:

64: switch (operation)

65: {

66: case "+":

67: result = (operand1) + (operand2);

68: break;

69: case "-":

70: result = (operand1) - (operand2);

71: break;

72: case "\*":

73: result = (operand1) \* (operand2);

74: break;

75: case "/":

76: if (operand2 != 0)

77: result = (operand1) / (operand2);

78: else

79: {

80: MessageBox.Show("Cannot divide by zero.","Error");

81: result = 0;

82: }

83: break;

84: default:

85: break;

86: }

87:

88: txt\_result.Text = result.ToString();

89: }

90: }

91:}

92:

**Output:** (Print your screen for output)

**Result:**(Write down the result)

**Viva Questions**

1. **What is the difference between “continue” and “break” statements in C#?**

Using break statement, you can 'jump out of a loop' whereas by using continue statement, you can 'jump over one iteration' and then resume your loop execution.

1. **Mathematical Function in c#**

|  |  |
| --- | --- |
| **Math Functions** | **Description** |
| Abs | Absolute value. Abs (a) corresponds to a = a if a> = 0 and a = -a if a <0 |
| Sqrt | To find the square root |
| Pow | Pow (x,y) corresponds to xY |
| Exp | Exp (x) corresponds to eX. where e is the base of the natural is logarithms |
| Log | natural logarithm. Logarithm to base e |
| Log10 | Logarithm to base 10 |
| Sin | Sine value |
| Asin | Arc sine |
| Sinh | Hyperbolic sine |
| Cos | Cosine |
| Acos | Arc cosine |
| Cosh | Cosine hyperbolic |
| Tan | Tangent |
| Atan | Arctangent |
| Tanh | Hyperbolic tangent |
| Round | Round the floating point numbers |
| Ceiling | Round Up |
| Floor | Round off |
| Max | Max (a,b) is the larger of the values ​​a and b |
| Min | Min (a ,b) is the smaller of the values ​​a and b |

**Skill Practice;**

* List out the windows from Visual studio Integrated Development Environment(IDE)

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

* List out the basic property name from the properties window

|  |  |  |  |
| --- | --- | --- | --- |
| **Property Name** | **Purpose or Usage** | **Property Name** | **Purpose or Usage** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Part A: Ex No: 4 - To implement a Calendar control to pick**

**a date and display the day**

**Aim:**

Develop a form in VB.Net to pick a date from calendar control and display the Day, Month, Year details in separate Textboxes.

**Pre-Learning Skills**

* Understanding the date and time functions
* Identifying the various format of date available

**Procedure:**

1. Open a New project.
2. Choose the **windows** **application** from the Templates.
3. Design the Form as follows

* Change the Form Caption to the Title of the program
* Drag and Drop required Controls from the tool box and change the name and text properties.
* Add a Calendar control to select the date.

1. Write the following code in the click () event of all the buttons.
2. Save and Execute (press F5).

**Source Code: LP\_4\_1**

1: using System;

2: using System.Windows.Forms;

3: namespace LP4\_1

4: {

5: public partial class Form1 : Form

6: {

7: public Form1()

8: {

9: InitializeComponent();

10: }

11: private void button1\_Click(object sender, EventArgs e)

12: {

13: // Get the current date in a datetime format i.e,

//12/04/2021

14: string currentDate1 =

DateTime.Now.ToString("dd/MM/yyyy");

15: // Displaying the current date

16: MessageBox.Show(currentDate1);

17: }

18: }

19:}

**Source Code: LP\_4\_2**

1: using System;

2: using System.Windows.Forms;

3:

4: namespace LP4\_2

5: {

6: public partial class Form1 : Form

7: {

8: public Form1()

9: {

10: InitializeComponent();

11: }

12: private void button1\_Click(object sender, EventArgs e)

13: {

14: listBox1.Items.Add(textBox1.Text);

15: }

16: private void listBox1\_SelectedValueChanged()

17: {

18: DateTime currentDate1 = DateTime.Now;

19: label1.Text = currentDate1.ToString(listBox1.Text);

20: }

21: }

22:}

**Source Code: LP\_4\_3**

1: using System;

2: using System.Windows.Forms;

3:

4: namespace LP4\_3

5: {

6: public partial class Form1 : Form

7: {

8: public Form1()

9: {

10: InitializeComponent();

11: }

12: private void btn\_click\_to\_add\_Click()

13: {

14: DateTime currentDate1 = DateTime.Now;

15: DateTime addedDate =

currentDate1.AddDays(int.Parse(txt\_input.Text));

16: MessageBox.Show(addedDate.ToString("F"));

17: }

18: private void Form1\_Load(object sender, EventArgs e)

19: {

20: txt\_current\_date.Text = DateTime.Now.ToString();

21: }

22: private void btn\_add\_month\_Click()

23: {

24: DateTime currentDate1 = DateTime.Now;

25: DateTime addedmonth =

currentDate1.AddMonths(int.Parse(txt\_input.Text));

26: MessageBox.Show(addedmonth.ToString("F"));

27: }

28: private void btn\_add\_year\_Click(object sender, EventArgs e)

29: {

30: DateTime currentDate1 = DateTime.Now;

31: DateTime addedyear =

currentDate1.AddYears(int.Parse(txt\_input.Text));

32: MessageBox.Show(addedyear.ToString("F"));

33: }

34: }

35:}

36:

**Source Code: Ex\_4**

1: using System;

2: using System.Windows.Forms;

3:

4: namespace Ex\_4

5: {

6: public partial class Form1 : Form

7: {

8: public Form1()

9: {

10: InitializeComponent();

11: }

12: private void dateTimePicker1\_ValueChanged()

13: {

14: DateTime dt = dateTimePicker1.Value;

15: //Displays the day number

16: txt\_day.Text = dt.Day.ToString();

17: // Displays the month number

18: txt\_month.Text = dt.Month.ToString();

19: // Displays the year

20: txt\_year.Text = dt.Year.ToString();

21: //Displays the day

22: txt\_dayname.Text = dt.ToString("dddd");

23: //Displays the month name

24: txt\_monthname.Text = dt.ToString("MMMM");

25: }

26: }

27: }

28:

**Output:** (Print your screen for output)

**Result :**( Write down the result)

**Viva Questions**

1. **How to increase the Date corresponding with month,date,year?**

DateSerial(year(Now),Month(Now)+1,1)  
 Hour, min, sec, month, year, DateSerial, dateadd, datediff, weekday, datevalue, timeserial,timevalue.

1. **Name some date function?**

Dateadd (), Datediff (), Datepart (), Cdate ()

1. **What is Mask Edit and why it is used?**

Masked intelligent user control enhances the function of the TextBox control, which can mask the Date, IP Address, Phone numbers, digits, decimal and checks the validation, and automatically set the delimiter location.

**Skill Practice;**

* Design a window form using calendar control and to find your age as on today

**Part A: Ex No: 5 - To Implement Common Dialog Control with**

**File and Directory Controls**

**Aim:**

To develop a VB.Net application using the File, Directory and Directory Controls to implement a Common dialog box.

**Pre-Learning Skills**

* Understanding the list of common dialog controls from the tool box
* Properties of each control can be analyzed
* Learn about events and its types

**Procedure:**

1. Open a New project.
2. Choose the **windows** **application** from the Templates.
3. Add the following .net components from library (right click toolbox and click choose items)

* Drivelistbox
* Dirlistbox
* Filelistbox

1. Design the Form as follows

* Drag and Drop required Controls from the tool box and change the name and text properties.

1. Write the following code in the click () event of all the buttons.
2. Save and Execute (press F5).

**Source Code: LP\_5\_1-** To display selected color using color dialog

1: using System;

2: using System.Windows.Forms;

3: namespace Lab\_Practice\_5\_1

4: {

5: public partial class LP\_5\_1 : Form

6: {

8: private void show\_color\_Click(object sender, EventArgs e)

9: {

10: ColorDialog dlg = new ColorDialog();

11: dlg.ShowDialog();

12: if (dlg.ShowDialog() == DialogResult.OK)

13: {

14: string str = null;

15: str = dlg.Color.Name;

16: MessageBox.Show (“Selcted Color is “ + str);

17:

18: }

19: }

20: }

21: }

22:

**Source Code: LP\_5\_2 -** To Change the text font using font dialog

1: using System;

2: using System.Windows.Forms;

3: namespace Lab\_Practice\_5\_2

4: {

5: public partial class LP\_5\_2 : Form

6: {

7:

8: private void show\_color\_Click(object sender, EventArgs e)

9: {

10: FontDialog fontdlg = new FontDialog();

11: fontdlg.ShowDialog();

12: if (fontDlg.ShowDialog() != DialogResult.Cancel)

13: {

14: textBox1.Font = fontDlg.Font;

15: label1.Font = "Selected Font “ + fontDlg.Font;

16: }

17: }

18: }

19: }

20:

**Source Code: Ex\_5**

1: using System;

2: using System.Windows.Forms;

3:

4: namespace Ex\_5

5: {

6: public partial class Form1 : Form

7: {

8: public Form1()

9: {

10: InitializeComponent();

11: }

12: private void fileListBox1\_SelectedIndexChanged()

13: {

14: txt\_foldername.Text = fileListBox1.Path;

15: txt\_foldername.Text = dirListBox1.Path;

16: txt\_filename.Text = (fileListBox1.FileName);

17: // Loads the image in picture box

18: pictureBox1.ImageLocation = dirListBox1.Path + "\\" +

fileListBox1.FileName;

19: }

20: private void driveListBox1\_SelectedIndexChanged()

21: {

22: dirListBox1.Path = driveListBox1.Drive;

23: }

24: private void dirListBox1\_SelectedIndexChanged()

25: {

26: fileListBox1.Path = dirListBox1.Path;

27: }

28: private void btn\_select\_folder\_Click()

29: {

30: folderBrowserDialog1.ShowDialog();

31: txt\_foldername.Text =

folderBrowserDialog1.SelectedPath;

32: }

33: private void btn\_select\_file\_Click()

34: {

35: openFileDialog1.ShowDialog();

36: txt\_filename.Text = openFileDialog1.FileName;

37: }

38: }

39:}

40:

**Output:** (Print your screen for output)

**Result:**(Write down the result)

**Viva Questions**

1. **How to assign FONT to label during runtime?**

Label.font.name=”Times New Roman”;

Label.font.size=20

1. **What are common dialog controls?**

The common dialog boxes control, which is a custom control, allows your project to use the dialog boxes that are provided as a part of the Windows environment.

1. **Name some showprinter method in Printer object?**

cdlPrinter.ShowPrinter  
Printer.Copies = cdlPrinter.Copies  
Printer.Orientation = cdlPrinter.Orientation  
Printer.EndDoc

**Skill Practice;**

* Observe the windows form and fill out the format

|  |  |  |  |
| --- | --- | --- | --- |
| **Control Name** | **Control Property Name** | **Control**  **Property Value** | **Event Name** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Part A: Ex No: 6 - Store Details of students using ADO.Net**

**Aim:**

Develop a database application to store the details of students using   
ADO.NET

**Pre-Learning Skills**

* Understanding the SQLServer database creation
* Known about Table creation, Column creation with data type and constraints
* User Interaction forms

**Procedure:**

1. Open a New project.
2. Choose the **windows** **application** from the Templates.
3. Open Server explorer from the menu
4. Right click the data connection and choose Add New connection.
5. Select the database file and path and click create new one
6. Expand the Database and select tables
7. Right Click and select add new table
8. Design the table as per the students data
9. Click Update to save the table
10. Design the Form as follows

* Change the Form Caption to the Title of the program
* Drag and Drop required Controls from the tool box and change the name and text properties.
* Place a Save button and Close button.

1. Write the following code in the click () event of all the buttons.

1. Open a connection using SQL Provider.

2. Execute Query using Command

3. Close the connection

1. Save and Execute (press F5).

**Source Code:**

1: using System;

2: using System.Data;

3: using System.Data.SqlClient;

4: using System.Windows.Forms;

5: namespace Ex\_6

6: {

7: public partial class Form1 : Form

8: {

9: public Form1()

10: {

11: InitializeComponent();

12: }

13: // ADO.Net Connection Object

14: SqlConnection con = new SqlConnection();

15: //ADO.Net Command Object

16: SqlCommand cmd = new SqlCommand();

17: string sql;

18: private void btn\_save\_Click(object sender, EventArgs e)

19: {

20: //Connection Property

21: con.ConnectionString = @"Data

Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=" +

Application.StartupPath + @"\student\_db.mdf;Integrated

Security=True";

22: System.Diagnostics.Debug.WriteLine(con.ConnectionString);

23: // Open Connection

24: con.Open();

25: //Parameterized SQL Command for INSERT Records in a TABLE

267: sql = "insert into

student\_master(rollno,student\_name,dept\_name,address,

mobileno)values(@rno,@stuname,@dept,@address,@mobile)”;

27: cmd = new SqlCommand(sql, con);

28: // Add Parameter Value

29: cmd.Parameters.Add("@rno",SqlDbType.Int).Value =

txt\_rno.Text;

30: cmd.Parameters.Add("@stuname",SqlDbType.VarChar).Value=

txt\_stuname.Text;

31: cmd.Parameters.Add("@dept", SqlDbType.VarChar).Value =

cmb\_deptname.SelectedItem.ToString();

32: cmd.Parameters.Add("@address",SqlDbType.VarChar).Value=

txt\_address.Text;

34: cmd.Parameters.Add("@mobileno",SqlDbType.VarChar).Value=

txt\_mobile.Text;

35: //Execute method return integer value

36: int result = cmd.ExecuteNonQuery();

37: if (result == 1)

38: MessageBox.Show("Student's Data Inserted.");

39: else

40: MessageBox.Show("Student's Data Not Inserted.");

41: // Close Connection

42: con.Close();

43: }

44:}` }

**Output:** (Print your screen for output)

**Result:**(Write down the result)

**Viva Questions**

1. **What is ADO.NET**?

ADO.NET is the primary relational data access model for Microsoft .NET-based applications.It may be used to access data sources.

ADO.NET provides mainly the following two types of architectures:

* Connected Architecture.
* Disconnected Architecture.

1. Define different execute methods of ADO.NET command object?

* **ExecuteScalar: -** This method returns a single value from the first row and first column of the result get from the execution of SQL query.
* **ExecuteNonQuery**: - This method executes the DML SQL query just like insert, delete or update and then returns the number of rows affected by the action.

1. **What are the different namespaces available in ADO.NET?**

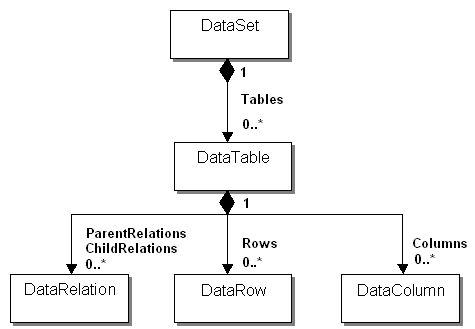
**System.Data:** It contains the definition for rows, columns, relations, views, tables, constraints, and databases.

**System.Data.SqlClient:** It is a collection of classes that are helpful in connecting to a Microsoft SQL Server 2008 database such as SqlConnection, SqlCommand, SqlDataAdapter, etc.

**System.Data.Odbc:** It consists of classes that are required for connecting with most Odbc Drivers. These classes include OdbcConnection, OdbcCommand.

**System.Data.OracleClient:** It has classes required for connection with an Oracle database, OracleConnection, OracleCommand

1. **How can you define the DataSet structure?**



1. **What are the different namespaces available in ADO.NET?**

**System.Data:** It contains the definition for rows, columns, relations, views, tables, constraints, and databases.

**System.Data.SqlClient:** It is a collection of classes that are helpful in connecting to a Microsoft SQL Server 2008 database such as SqlConnection, SqlCommand, SqlDataAdapter, etc.

**System.Data.Odbc:** It consists of classes that are required for connecting with most Odbc Drivers. These classes include OdbcConnection, OdbcCommand.

**System.Data.OracleClient:** It has classes required for connection with an Oracle database, OracleConnection, OracleCommand

**Skill Practice;**

* Observe the windows form and fill out the format

|  |  |  |
| --- | --- | --- |
| **Control Name** | **Control Property Name** | **Control**  **Property Value** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* Observe the ADO.Net Code and fill out the format

|  |  |  |
| --- | --- | --- |
| **Command Object** | **Properties used** | **Method used if any** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* Design a user interactive form for employee management

**Part A: Ex No: 7 - Create an ASP.Net webpage for login entry**

**Aim:**

Create a simple ASP.NET page to Output Text with a form, two HTML text boxes, an HTML button, and an HTML <span> element. Create an event procedure for the button.

**Procedure:**

1. Open Visual Studio –> Create New Empty Website
2. Add New Web forms
3. Design Login page with Two TextBox along with Button control as below

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="loginpage.aspx.cs" Inherits="loginpage" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head runat="server">

    <title>Login Page</title>

</head>

<body>

    <form id="form1" runat="server">

    <div align="center">

    <fieldset style ="width:200px;">

    <legend>Login page </legend>

<span id=”user” style="color: red;">

<asp:label ID =”label1” runat=”server”>Enter a <span>User Name</span>

</span>

        <asp:TextBox ID="txtusername" placeholder="username" runat="server"

            Width="180px"></asp:TextBox>

        <br />

        <br />

<span id=”pass” style="color: red;">

<asp:label ID =”label1” runat=”server”>Enter a <span>Password</span>

</span>

        <asp:TextBox ID="txtpassword" placeholder="password" runat="server"

            Width="180px" TextMode="Password"></asp:TextBox>

        <br />

        <br />

        <asp:Button ID="btnsubmit" runat="server" Text="Submit"

           Width="81px" onclick="btnsubmit\_Click" />

            <br />

           <span id=”user” style="color: red;">

<asp:label ID =”lbl\_result” runat=”server”>Enter a <span>User Name</span>

</span>

    </fieldset>

    </div>

    </form>

</body>

</html>

1. Write C# code on Login Button click events.

**Source Code:**

1: using System;

2: using System.Web;

3: using System.Web.UI;

4: using System.Web.UI.WebControls;

5:

6: namespace Ex\_7

7: {

8: public partial class login : System.Web.UI.Page

9: {

10: protected void btn\_login\_Click(object sender, EventArgs e)

11: {

12: if (txtusername.Text == "")

13: lbl\_result.Text = "Enter UserName";

14: else if (txtpassword.Text == "")

15: lbl\_result.Text = "Enter Password";

16: else

17: {

18: // check whether given username and password is equal to

19: // “Computer” & “mspvl”

20: if (txtusername.Text == "Computer" &&

txtpassword.Text == "mspvl")

21: lbl\_result.Text = "Login Success";

22: else

23: lbl\_result.Text = "Invalid Name or Password";

24: }

25: }

26: }

27: }

28:

**Output:** (Print your screen for output)

**Result:**(Write down the result)

**Viva Questions;**

* 1. **What does <span> HTML Tag do?**

The <span> element is the inline equivalent to the block-level <div> element. It is used to select inline content for purely stylistic purposes.

* 1. **What is static and dynamic login form?**

Static Login Form

Username and password are fixed

Predefined user can login

Does not use database

Write username and password values on coding page

Dynamic Login Form

Username and Password are not fixed

Multiple use can login

Use Database

Fetch username and password values from database