BACHELOR OF COMPUTER APPLICATION

R.D COLLEGE SHEIKHPURA

(A CONSTITUENT UNIT OF MUNGER UNIVERSITY)

MUNGER UNIVERSITY, MUNGER

GROUP-B

SUBJECT : C#

SUBJECT CODE : 405

PROJECT NAME : CBT(EXAM)

SEMESTER : 4TH

SESSION : 2021-2024

➢ SUBMITTED BY

MEMBERS OF THE GROUP

NAME	ROLL NO
CHHOTU KUMAR	**521090005**
VIVEK KUMAR	**521090031**
SONI KUMARI	**521090027**
SWATI KUMARI	**521090029**
MONI KUMARI	**521090018**

TEAM LEADER ⟶ CHHOTU KUMAR

UNDER THE GUIDENCE OF ANJANI SIR

BACHELOR OF COMPUTER APPLICATION

R.D COLLEGE SHEIKHPURA

(A CONSTITUENT UNIT OF MUNGER UNIVERSITY)

MUNGER UNIVERSITY, MUNGER

GROUP-B

SUBJECT : C#

SUBJECT CODE : 405

PROJECT NAME : CBT(EXAM)

SEMESTER : 4TH

SESSION : 2021-2024

.....

SUPERIVSION .....

EXTERNAL EXAMINER

CONTENTS

ACKNOWLEDGEMENTS

A part from the efforts of my team, the success~ of any project depends largely on the encouragement and guidelines of many others we take this opportunity to express our gratitude to the people who have been. instrumental in the successful completion. of this Project:

we express deep Sense of gratitude to almighty god for giving us Strength, for the successful completion of the proved.

we express our heartfelt gratitude to our parents for constant encouragement while carrying out this project.

We gratefully acknowledge the contribution of the individual who contributes in bringing this project up to this level, who continous to look after our despite our flows.

who express our deep sense of gratitude to the luminary the principal, Dr Diwakar sir who has been continuously motivating and exlending their helping hand to us.

we express our sincere thanks to the our class teacher TP sir , Anjani sir, for Constant encouragement and the guidance Provided during this project. The guidance and support received from all the member who contributed and who are contributing to this project, was vital for the success of the project we are greatful for their constant support and help.

PREFACE

In the preparation of this project of CBT, we have precisely demarcated all the important point. we have made our best possible efforts to remove all the errors.

It is a greate pleasure for us to thanks all these valuable suggestions that have given to us by ANJANI SIR. We must thank and guidance as well as our parents, teachers who directed us to complete this project.

INTRODUCTION OF C#

C# (pronounced "See Sharp") is a modern, object-oriented, and type- safe programming language. C# enables developers to build many types of secure and robust applications that run in .NET. C# has its roots in the C family of languages and will be immediately familiar to C, C++, Java, and JavaScript programmers. This tour provides an overview of the major components of the language in C# 8 and earlier. If you want to explore the language through interactive examples, try the introduction to C# tutorials.

.NET architecture: -

C# programs run on .NET, a virtual execution system called the common language runtime (CLR) and a set of class libraries. The CLR is the implementation by Microsoft of the common language infrastructure (CLI), an international standard. The CLI is the basis for creating execution and development environments in which languages and libraries work together seamlessly.

INTRODUCTION OF CBT

A computer based test(CBT) is a way of giving test using computer tests are conducted whit the help of the internet and a computer-assisted facility. the CBT exam streamlines intstructor's evaluation tasks by incorporating features such as MCQS, comprehensive responses analytic based surveys and so on.

What is CBT

In a computer based test , candidates use computer screen and a mouse to answer questions. questions appear on the screen one by one and candidates solve the problems in a rough paper, and chose the correct answer on the screen . most of the time 3 - 4 .question sets are prepared for computer based test and question sets are randomly allocated to the candidates.

Advantages of computer - Based Tests

• Location independent: computer based tests can be taken at any time from any Place, given that there is a stable internet connection. The students can appear for the exam at a Convenient location. they don't need to go to a test center to take exams. It enables the student from remote areas to take up exam at their own time as well.

Reduces Time:- Paper based exams takes a lot of time to set up. But in computer based exam, the Question papers are distributed virtually. The candidates gets a set time, the Candidatel answers automatically get submitted. It saver the time and effort of collecting answer the time and effort of collecting answer Sheets from the Candidates one by one.

• Increases security:- in a computer based examination, multiple sets of questions paper are created within similar Questions and difficulty. And these questions Paper are given to the candidates randomly, It lowers the risk of cheating in examination.

DISADVANTAGE of computer - Based Tests

1. The initial cost of acquiring the CBT tecnology is expensive. that is what deter many government and private organizations from integrating it into the education system .
2. CBT exam have it's limitations .it can be affected by external factor like a technical glitch, power problem, internet problem or other system failures.
3. In India, not a lot of test center are equipped with infrastructure that can conduct computer based examinations at scale.

OBJECTIVE

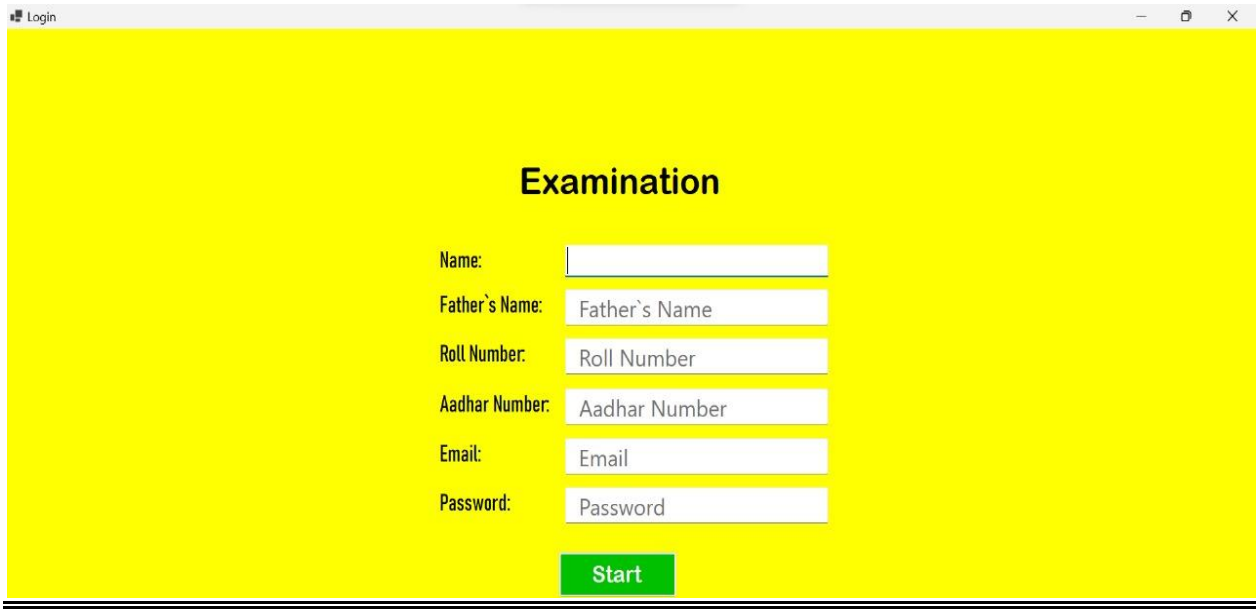
It not only saves time and resources but also helps in tracking students progress through the data that is generated. The student-centric analytics provides the learners with in depth insignts into their activities ultimately helping them learn about their strength and weaknesses.

Conclusion

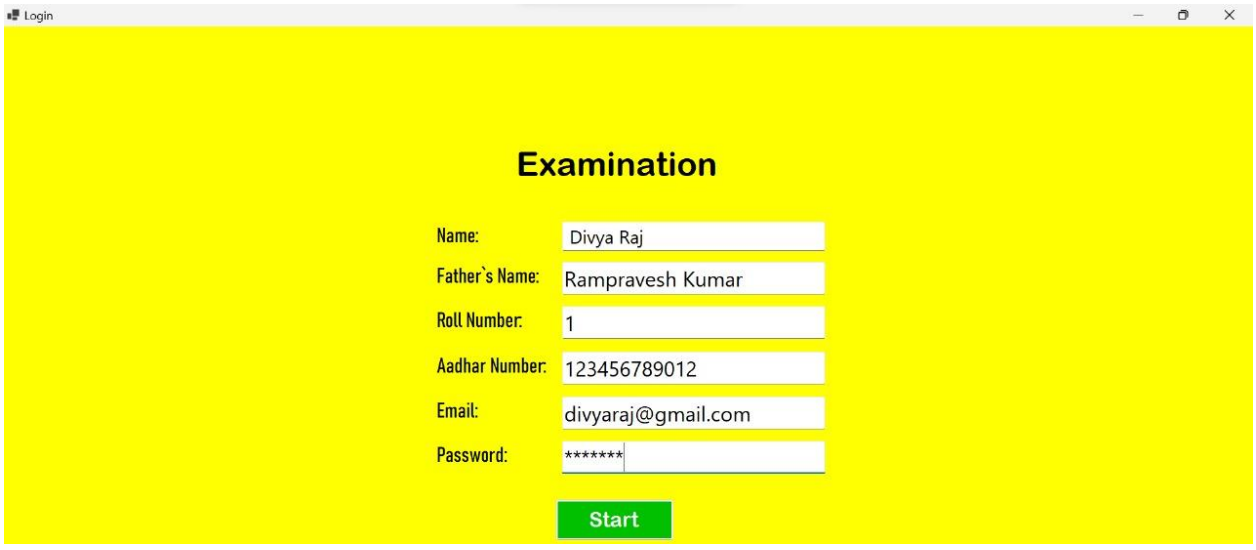
computer based assessment is a tool to measure student's Progress. if not used appropriately, It will not produce the desired outcomes. Technology offers a multitude of benefits to assessment. But successful assessment depends on the appropriate use of technology so they can fulfill their purpose.

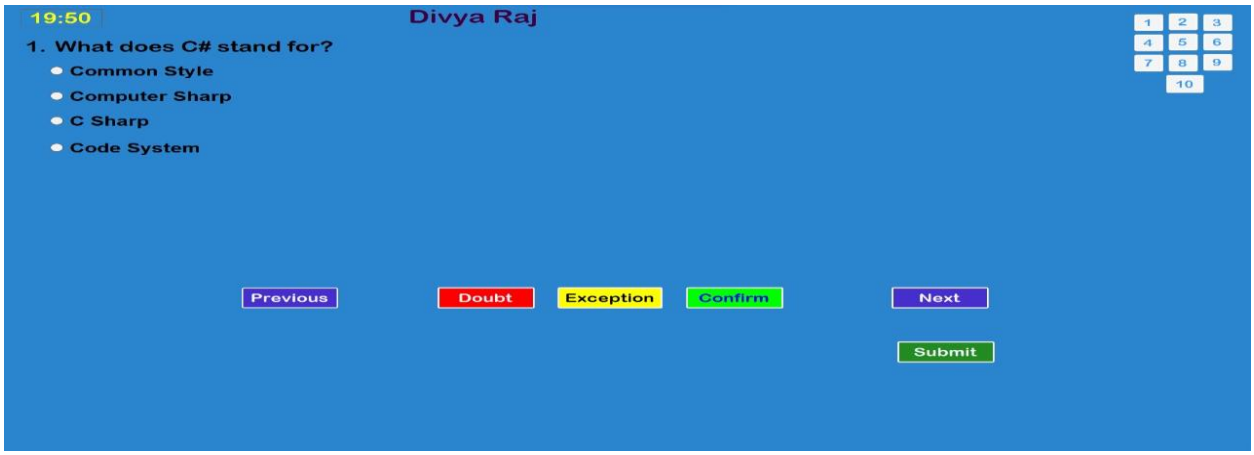
In the source code window, the object displayed is form1 and the associated procedure is load.

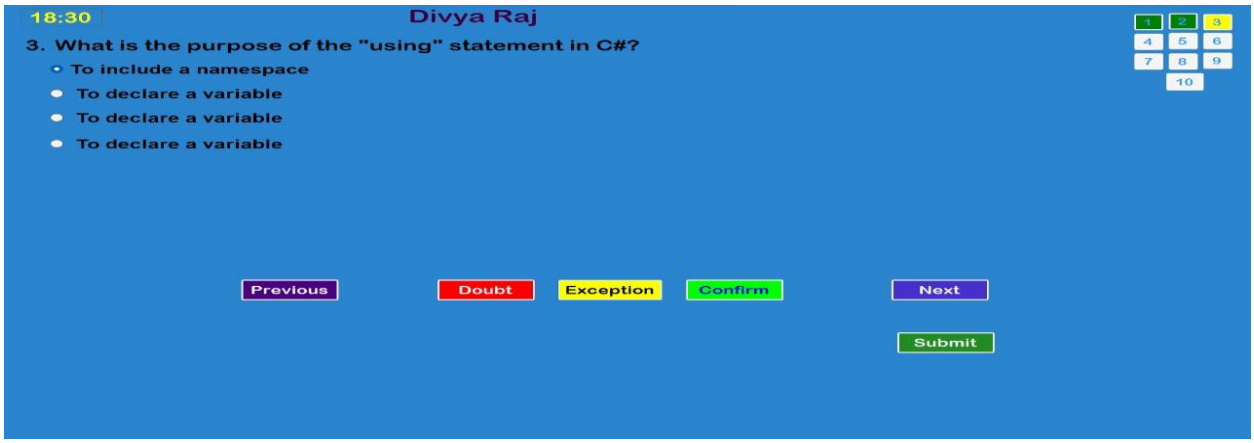
DESIGN MAIN SCREEN

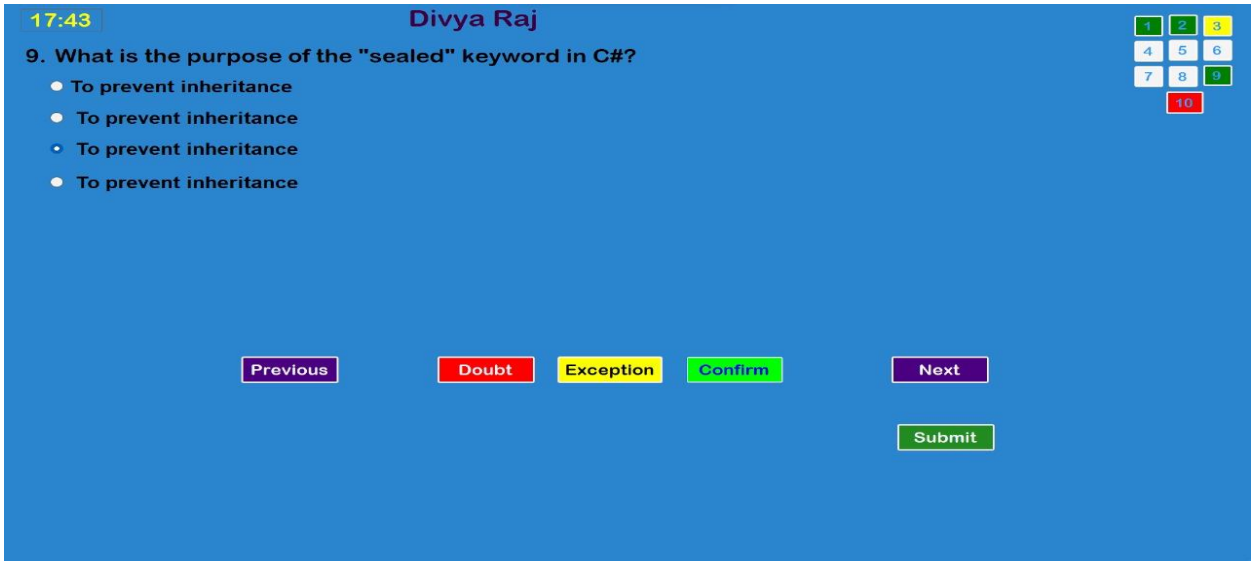


(FIG:-HOME PAGE)




19:50
Divya Raj
1. What does C# stand for?
Common Style
Computer Sharp
C Sharp
Code System
1 2 3
4 5 6
7 8 9
10
Previous
Doubt
Exception
Confirm
Next
Submit


18:30
Divya Raj
3. What is the purpose of the "using" statement in C#?
To include a namespace
To declare a variable
To declare a variable
To declare a variable
1 2 3
4 5 6
7 8 9
10
Previous
Doubt
Exception
Confirm
Next
Submit


17:43
Divya Raj
9. What is the purpose of the "sealed" keyword in C#?
To prevent inheritance
To prevent inheritance
To prevent inheritance
To prevent inheritance
1 2 3
4 5 6
7 8 9
10
Previous
Doubt
Exception
Confirm
Next
Submit

SOURCE CODE

(login form code)

```
using System;
using System.Text.RegularExpressions;
namespace CBT
{
public partial class Form1 : Form
{
userDetails userDetails = new userDetails();
public Form1()
{
InitializeComponent();
}
private void StartExamination_btn_Click(object sender, EventArgs e)
{
if (ValidateUserData())
{
ExaminationForm examinationForm = new ExaminationForm(userDetails);
this.Hide();
examinationForm.ShowDialog();
}
}
private bool ValidateUserData()
{
// Check if the required fields are not empty and meet specific criteria
if (string.IsNullOrEmpty(Form1_Name_TextBox.Text) ||
string.IsNullOrEmpty(Form1_FatherName_TextBox.Text) ||
string.IsNullOrEmpty(Form1_RollNumber_TextBox.Text) ||
string.IsNullOrEmpty(Form1_Aadhar_TextBox.Text) ||
string.IsNullOrEmpty(Form1_Email_TextBox.Text) ||
string.IsNullOrEmpty(Form1_Password_TextBox.Text))
{
MessageBox.Show("Please fill in all the required fields with valid data.");
return false;
}
// Validate Roll Number (numeric input)
if (!int.TryParse(Form1_RollNumber_TextBox.Text, out int rollNumber))
{MessageBox.Show("Please enter a valid numeric value for Roll Number.");
```

```
return false;
}
// Validate Aadhar Number (12-digit numeric input)
if (Form1_Aadhar_TextBox.Text.Length != 12 || !long.TryParse(Form1_Aadhar_TextBox.Text,
out long aadharNumber))
{
MessageBox.Show("Please enter a valid 12-digit numeric value for Aadhar Number.");
return false;
}
// Validate Email format
if (!IsValidEmail(Form1_Email_TextBox.Text))
{
MessageBox.Show("Please enter a valid email address.");
return false;
}
// Additional validation logic can be added for other fields
// If all validations pass, update the userDetails object
userDetails.Name = Form1_Name_TextBox.Text;
userDetails.FatherName = Form1_FatherName_TextBox.Text;
userDetails.RollNumber = rollNumber;
userDetails.AadharNumber = aadharNumber.ToString(); // Store Aadhar as string to preserve
leading zeros
userDetails.Email = Form1_Email_TextBox.Text;
userDetails.Password = Form1_Password_TextBox.Text;

return true;
}
private bool IsValidEmail(string email)
{
try
{
var addr = new System.Net.Mail.MailAddress(email);
if (!addr.Host.Contains('.'))
{
return false;
}
return addr.Address == email;
}
catch
{
return false;
}
}
}
```

(exam form code)

```
using System.Timers;
namespace CBT
{
public partial class ExaminationForm : Form
{
int Count = 0;
int PreviousQuestionN = 0;
List<Question> questions = new List<Question>();
private userDetails userDetails = new userDetails();
public ExaminationForm()
{
}
public ExaminationForm(userDetails userDetails)
{
this.ControlBox = false;
this.WindowState = FormWindowState.Maximized;
InitializeComponent();
SetAllQuestions();
this.userDetails = userDetails;
currentUserName.Text = userDetails.Name.ToString();
this.Text = null;
}

//----------> For Timer <----------
System.Timers.Timer timer1 = new System.Timers.Timer();
int m = 19, s = 59;

//==============>> Form Load <<=============
private void ExaminationForm_Load(object sender, EventArgs e)
{
timer1 = new System.Timers.Timer();
timer1.Interval = 1000;
timer1.Elapsed += OnTimeEvent;
timer1.Start();
GetCurrentQuestions();
}

//====================>>ALl Helper Methods Are Below<<==================

//------------> GetCurrentQuestions <-------------
private void GetCurrentQuestions()
{
```

```
QuestionN_lbl.Text = $"{Count + 1}.";
Question_lbl.Text = questions[Count].Text;
Option_A.Text = questions[Count].Options[0].ToString();
Option_B.Text = questions[Count].Options[1].ToString();
Option_C.Text = questions[Count].Options[2].ToString();
Option_D.Text = questions[Count].Options[3].ToString();
SetUserInputOption();
}

//------------>>unSelectAllOption<<------------
private void unSelectAllOption()
{
Option_A.Checked = false;
Option_B.Checked = false;
Option_C.Checked = false;
Option_D.Checked = false;
}

//--------------->>SetUserInputOption<<------------
private void SetUserInputOption()
{
//  MessageBox.Show($"User input value of Q {Count+1} = {questions[Count].userInput}");

switch (questions[Count].userInput)
{
case 'A':
Option_A.Checked = true;
break;
case 'B':
Option_B.Checked = true;
break;
case 'C':
Option_C.Checked = true;
break;
case 'D':
Option_D.Checked = true;
break;
case ' ':
Option_A.Checked = false;
Option_B.Checked = false;
Option_C.Checked = false;
Option_D.Checked = false;
break;
}
```

```
}

//--------------->>LockConfirmedQuestion<<-----------
private int LockConfimedQuestions()
{
if (getCurrentColorOfQuestionBtn() == Color.Green)
{
return 0;
}
else
{
return 1;
}
}

//====================>>ALl Helper Methods Are Above<<==================

//------------->>For Option A<<-------------
private void Option_A_CheckedChanged(object sender, EventArgs e)
{
questions[Count].userInput = 'A';
}

//------------->>For Option B<<-------------
private void Option_B_CheckedChanged(object sender, EventArgs e)
{
questions[Count].userInput = 'B';
}
//------------->>For Option C<<-------------
private void Option_C_CheckedChanged(object sender, EventArgs e)
{
questions[Count].userInput = 'C';
}
//------------->>For Option D<<-------------
private void Option_D_CheckedChanged(object sender, EventArgs e)
{
questions[Count].userInput = 'D';
}
//------------->Next,Previous,Submit<<-----------
private void Next_btn_Click(object sender, EventArgs e)
{
if (questions[Count].userInput != ' ' && getCurrentColorOfQuestionBtn() == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green);
```

```
}
unSelectAllOption();
Previous_btn.BackColor = Color.Indigo;
if (Count < 9)
{
Count++;
GetCurrentQuestions();
}
else
{
Next_btn.BackColor = Color.Gray;
}
SetUserInputOption();
}
private void Previous_btn_Click(object sender, EventArgs e)
{
unSelectAllOption();
Next_btn.BackColor = Color.Indigo;
if (Count > 0)
{
Count--;
GetCurrentQuestions();
}
else
{
Previous_btn.BackColor = Color.Gray;
}
SetUserInputOption();
}
private void Submit_btn_Click_1(object sender, EventArgs e)
{
DialogResult result = MessageBox.Show("Are you sure to submit?", "Confirmation",
MessageBoxButtons.YesNo);
if (result == DialogResult.Yes)
{
int temp = 0;
foreach (var element in questions)
{
if (element.CorrectAnswer == element.userInput)
{
temp++;
}
}
userDetails.TotalCorrect = temp;
```

```
ResultForm ResultForm = new ResultForm(userDetails);
this.Hide();
ResultForm.ShowDialog();
}
}

//=============>>Set All Questions<<=============
private void SetAllQuestions()
{
//QuestionNo 1
Question q1 = new Question(
"What does C# stand for?",
new List<string> { "Common Style", "Computer Sharp", "C Sharp", "Code System" },
'C',
' '
);
questions.Add(q1);

//QuestionNo 2
Question q2 = new Question(
"In C#, which keyword is used to declare a class?",
new List<string> { "type", "class", "struct", "object" },
'B',
' '
);
questions.Add(q2);

//QuestionNo 3
Question q3 = new Question(
"What is the purpose of the \"using\" statement in C#?",
new List<string> { "To include a namespace", " To declare a variable", " To declare a variable", "
To declare a variable" },
'A',
' '
);
questions.Add(q3);

//QuestionNo 4
Question q4 = new Question(
"In C#, what is the default access modifier for class members if no access modifier is
specified?",
new List<string> { "private",
"private",
"protected",
```

```
"protected" },
'B',
' '
);
questions.Add(q4);

//QuestionNo 5
Question q5 = new Question(
"How do you declare a constant in C#?",
new List<string> { "const int x = 5;",
"const int x = 5;",
"const int x = 5;",
"const int x = 5;" },
'A',
' '
);
questions.Add(q5);

//QuestionNo 6
Question q6 = new Question(
"Which of the following statements is used to handle exceptions in C#?",
new List<string> { "try-catch",
"if-else",
"if-else",
"for loop" },
'A',
' '
);
questions.Add(q6);

//QuestionNo 7
Question q7 = new Question(
"What does the \"this\" keyword refer to in C#?",
new List<string> { "Current instance of the class",
"Current instance of the class",
"Derived class",
"Static class" },
'A',
' '
);
questions.Add(q7);

//QuestionNo 8
Question q8 = new Question(
```

```
"In C#, what is the purpose of the \"as\" operator?",
new List<string> { "Arithmetic operation",
"Arithmetic operation",
"Logical AND",
"Bitwise shift" },
'B',
' '
);
questions.Add(q8);
//QuestionNo 9
Question q9 = new Question(
"What is the purpose of the \"sealed\" keyword in C#?",
new List<string> { "To prevent inheritance",
" To prevent inheritance", " To prevent inheritance", " To prevent inheritance" },
'A',
' '
);
questions.Add(q9);
//QuestionNo 10
Question q10 = new Question(
"Which of the following is NOT a valid C# data type?",
new List<string> { "double", "char", "decimal", "byte" },
'D',
' '
);
questions.Add(q10);
}
//------------>>Timer<<--------------
private void OnTimeEvent(object? sender, ElapsedEventArgs e)
{
if (Timer_lbl.IsHandleCreated)
{
Timer_lbl.BeginInvoke(new Action(() =>
{
s -= 1;
if (s == 0 && m > 0)
{
s = 59;
m -= 1;
}
if (m == 0 && s == 0)
{
MessageBox.Show(this, "Examination Time Out");
int temp = 0;
```

```
foreach (var element in questions)
{
if (element.CorrectAnswer == element.userInput)
{
temp++;
}
}
userDetails.TotalCorrect = temp;
ResultForm ResultForm = new ResultForm(userDetails);
this.Hide();
ResultForm.ShowDialog();
}
Timer_lbl.Text = $"{m:00}:{s:00}";
}));
}
}
private void Q1_btn_Click(object sender, EventArgs e)
{
Count = 0;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 0;
}
private void Q2_btn_Click(object sender, EventArgs e)
{
Count = 1;
//GetCurrentQuestions();
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 1;
}
private void Q3_btn_Click(object sender, EventArgs e)
{
Count = 2;
//GetCurrentQuestions();
```

```
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();

PreviousQuestionN = 2;
}
private void Q4_btn_Click(object sender, EventArgs e)
{
Count = 3;
// GetCurrentQuestions();
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();

PreviousQuestionN = 3;
}
private void Q5_btn_Click(object sender, EventArgs e)
{
Count = 4;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 4;
}
private void Q6_btn_Click(object sender, EventArgs e)
{
Count = 5;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 5;
}
```

```
private void Q7_btn_Click(object sender, EventArgs e)
{
Count = 6;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 6;
}
private void Q8_btn_Click(object sender, EventArgs e)
{
Count = 7;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 7;
}
private void Q9_btn_Click(object sender, EventArgs e)
{
Count = 8;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 8;
}
private void Q10_btn_Click(object sender, EventArgs e)
{
Count = 9;
if (questions[PreviousQuestionN].userInput != ' ' &&
getCurrentColorOfQuestionBtn(PreviousQuestionN) == Color.WhiteSmoke)
{
BtnColoringBg(Color.Green, PreviousQuestionN);
}
GetCurrentQuestions();
PreviousQuestionN = 9;
}
```

```
private void QuestionInDoubtBtn_Click(object sender, EventArgs e)
{
BtnColoringBg(Color.Red);
}
private void QuestionInExceptionBtn_Click(object sender, EventArgs e)
{
BtnColoringBg(Color.Yellow);
}
private void QuestionInConfirmBtn_Click(object sender, EventArgs e)
{
BtnColoringBg(Color.Green);
}
private void BtnColoringBg(Color bg)
{
switch (Count)
{
case 0:
Q1_btn.BackColor = bg;
break;
case 1:
Q2_btn.BackColor = bg;
break;
case 2:
Q3_btn.BackColor = bg;
break;
case 3:
Q4_btn.BackColor = bg;
break;
case 4:
Q5_btn.BackColor = bg;
break;
case 5:
Q6_btn.BackColor = bg;
break;
case 6:
Q7_btn.BackColor = bg;
break;
case 7:
Q8_btn.BackColor = bg;
break;
case 8:
Q9_btn.BackColor = bg;
break;
case 9:
```

```
Q10_btn.BackColor = bg;
break;
}
}
private void BtnColoringBg(Color bg, int QN)
{
switch (QN)
{
case 0:
Q1_btn.BackColor = bg;
break;
case 1:
Q2_btn.BackColor = bg;
break;
case 2:
Q3_btn.BackColor = bg;
break;
case 3:
Q4_btn.BackColor = bg;
break;
case 4:
Q5_btn.BackColor = bg;
break;
case 5:
Q6_btn.BackColor = bg;
break;
case 6:
Q7_btn.BackColor = bg;
break;
case 7:
Q8_btn.BackColor = bg;
break;
case 8:
Q9_btn.BackColor = bg;
break;
case 9:
Q10_btn.BackColor = bg;
break;
}
}
private Color getCurrentColorOfQuestionBtn()
{
Button[] questionButtons = { Q1_btn, Q2_btn, Q3_btn, Q4_btn, Q5_btn, Q6_btn, Q7_btn,
Q8_btn, Q9_btn, Q10_btn };
```

```
if (Count >= 0 && Count < questionButtons.Length)
{
return questionButtons[Count].BackColor;
}
else
{
return Color.WhiteSmoke;
}
}
private Color getCurrentColorOfQuestionBtn(int PreviousQN)
{
Button[] questionButtons = { Q1_btn, Q2_btn, Q3_btn, Q4_btn, Q5_btn, Q6_btn, Q7_btn,
Q8_btn, Q9_btn, Q10_btn };

if (PreviousQN >= 0 && PreviousQN < questionButtons.Length)
{
return questionButtons[PreviousQN].BackColor;
}
else
{
return Color.WhiteSmoke;
}
}
}
}
```

(result form code)

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace CBT
{
public partial class ResultForm : Form
{
private userDetails userDetails;
public ResultForm(userDetails userDetails)
```

```
{
InitializeComponent();
this.userDetails = userDetails;
}
private void ResultForm_Load(object sender, EventArgs e)
{
Name_lbl.Text = userDetails.Name;
FatherName_lbl.Text = userDetails.FatherName;
AadharNumber_lbl.Text = userDetails.AadharNumber;
RollNumber_lbl.Text = $"{userDetails.RollNumber}";
Email_lbl.Text = userDetails.Email;
CorrectAns_lbl.Text = $"{userDetails.TotalCorrect}";
Marks_lbl.Text = $"{userDetails.TotalCorrect * 2}";
//  CorrectAns_lbl.Text = userDetails.TotalCorrect;
// Marks_lbl.Text = int.Parse(userDetails.TotalCorrect) * 2;
}
private void CorrectAns_lbl_Click(object sender, EventArgs e)
{
}
private void StartExamination_btn_Click(object sender, EventArgs e)
{
Form1 form1 = new Form1();
ExaminationForm exForm = new ExaminationForm();
this.Close();
form1.Close();
exForm.Close();
}
}
}
```

(Question Store)

```
using System;
using System.Collections.Generic;
public class Question
{
public string Text { get; set; }
public List<string> Options { get; set; }
public char CorrectAnswer { get; set; }
public char userInput { get; set; }

public Question(string text, List<string> options, char correctAnswer,char userAnswere)
{
Text = text;
Options = options;
CorrectAnswer = correctAnswer;
```

```
userInput = userAnswere;
}
}
```

(user details store)

```
using System;
using System.Collections.Generic;
public class userDetails
{
public string Name { get; set; }
public int RollNumber { get; set; }
public string FatherName { get; set; }
public string AadharNumber { get; set; }
public string Email { get; set; }
public string Password { get; set; }


//**************
public string MarkPerQuestion { get; set; }
public string TotalNumberOfQuestons { get; set; }
public int TotalCorrect { get; set; }

}
```

OUTPUT

(FIG:-OUTPUT WINDOWS)

FUTURE SCOPE OF THE PROJECT

The future scope of computer-based tests is promising, with on going advancements in technology. expect continued security growth in adaptive testing, enhanced security measures, and the integration of artificial intelligence to personalize assessments. Additionally, remoteproctoring and virtual environments may become more prevalent offering flexibility and accessibility in the evaluation process.

BIBLIOGRAPHY

For successfully completing our project on CBT(exam) we have take help from the following websites like.

www.wikipedia.com

code guld.com

slide.share

youtube


EXAM