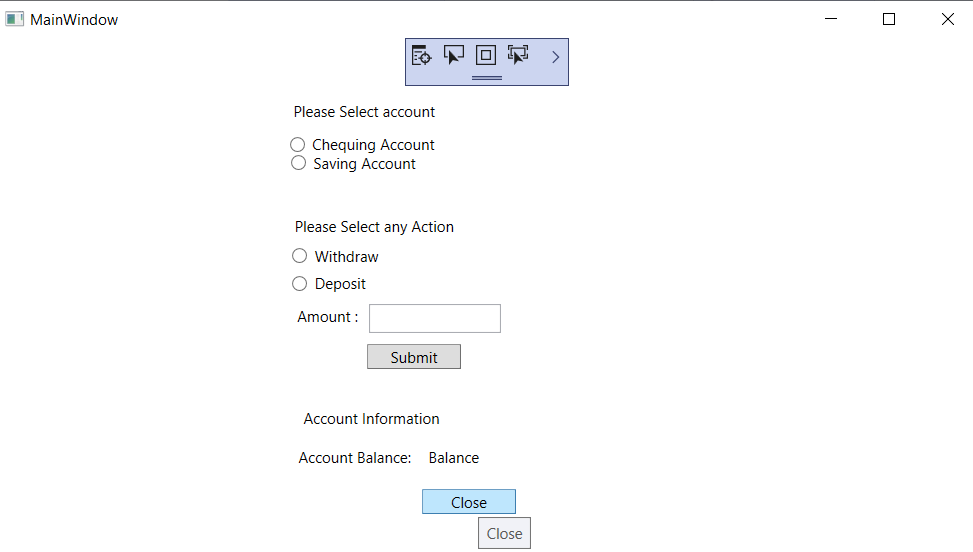
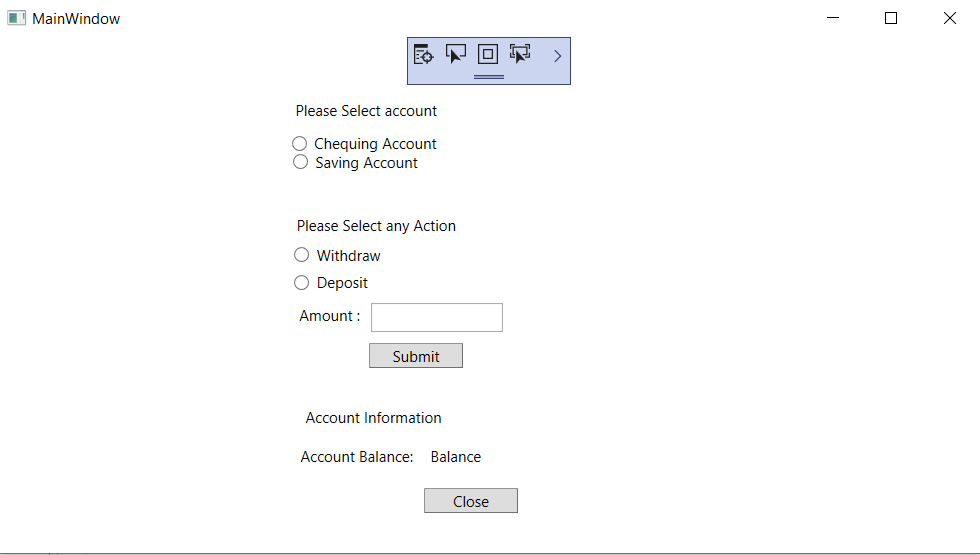
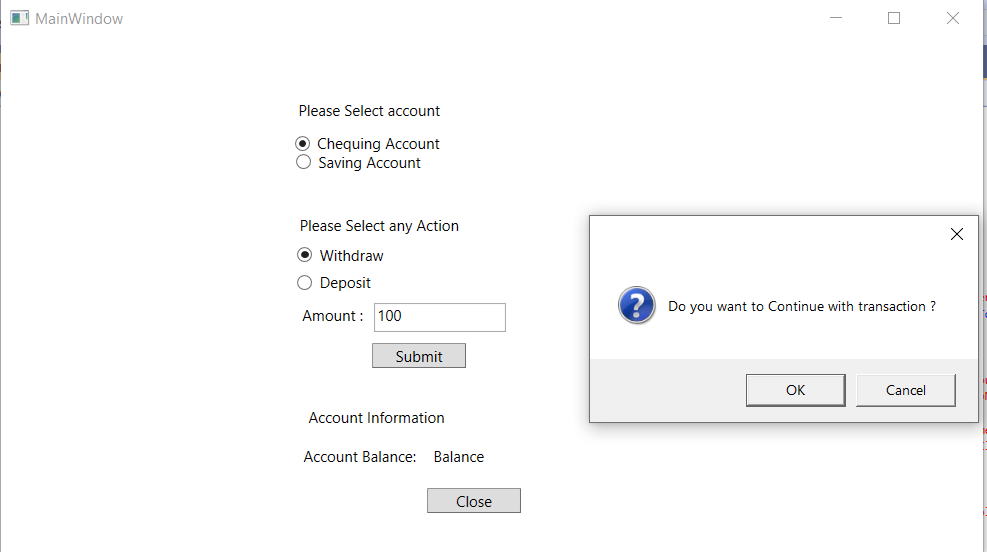
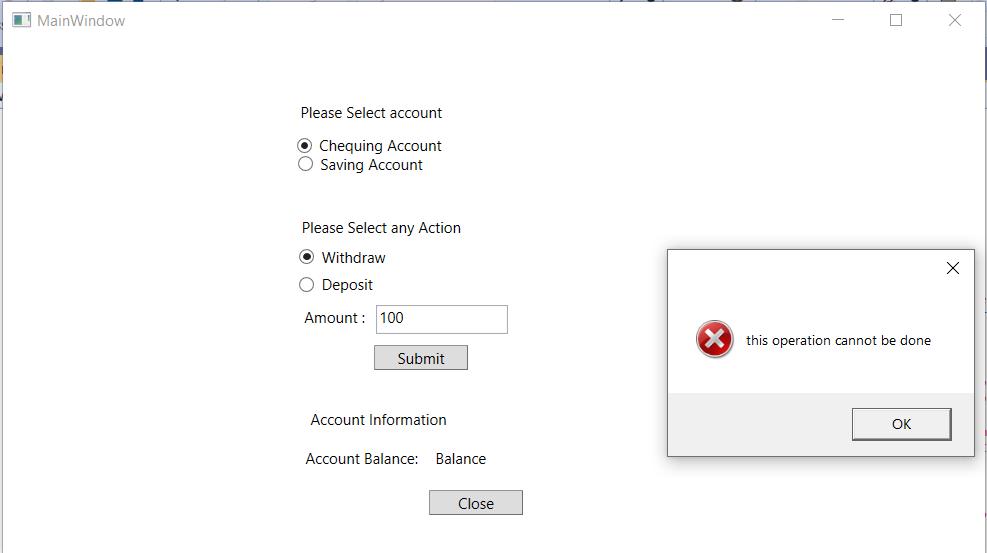
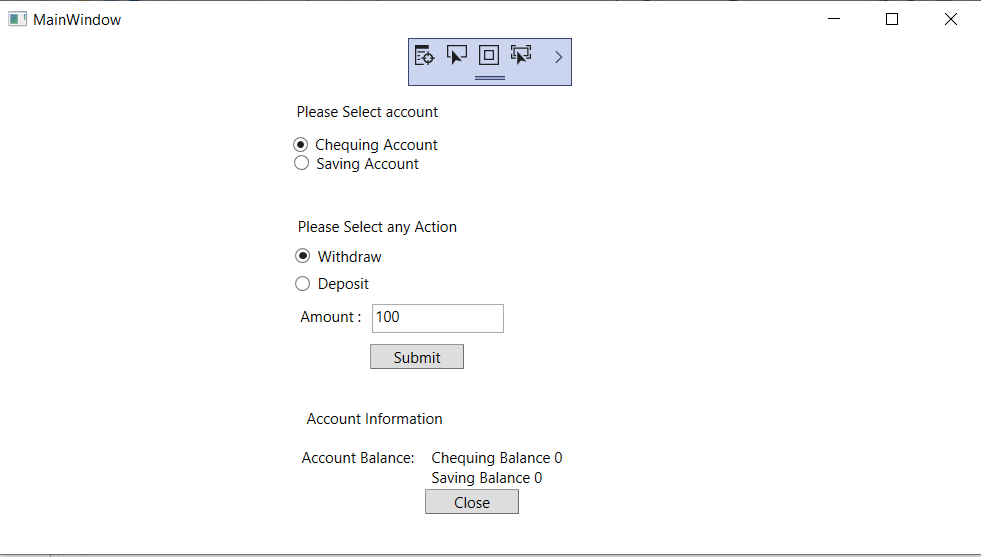
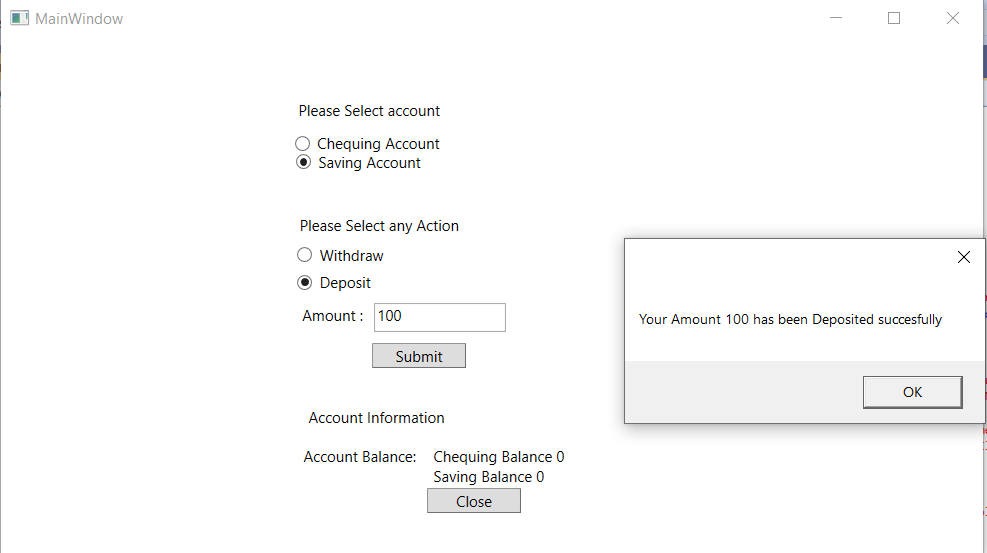
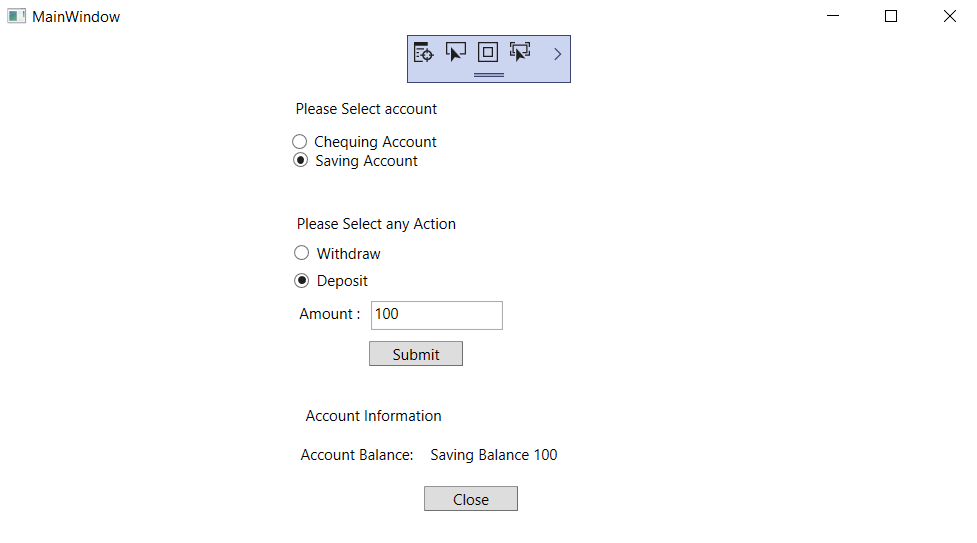
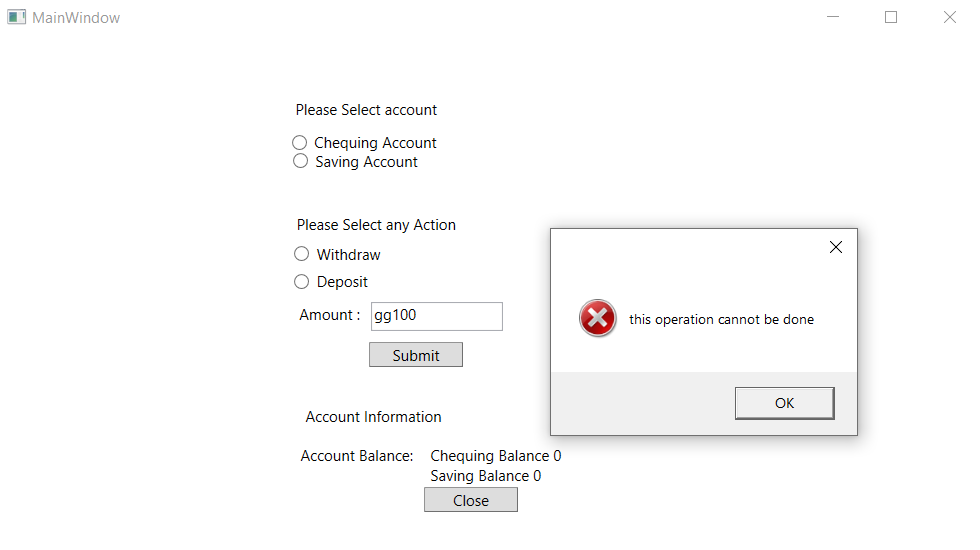
AksheshKumar Chaudhari(301087120)

Assignment-2

**Output:**



**Code:**

**C# Code:**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Data;

using System.Windows.Documents;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

using System.Windows.Navigation;

using System.Windows.Shapes;

namespace Assignment\_3

{

/// <summary>

/// Interaction logic for MainWindow.xaml

/// </summary>

public partial class MainWindow : Window

{

public MainWindow()

{

InitializeComponent();

}

double savingAmount=0;

double chequingAmount=0;

public void balanceExceed()

{

MessageBox.Show("Not Enough Balance");

}

private void Button\_Close(object sender, RoutedEventArgs e)

{

this.Close();

}

private void Button\_Click(object sender, RoutedEventArgs e)

{

try

{

MessageBoxResult messageBoxResult = System.Windows.MessageBox.Show("Do you want to Continue with transaction ?","" , MessageBoxButton.OKCancel,MessageBoxImage.Question);

if (messageBoxResult == MessageBoxResult.OK)

{

if (saving.IsChecked == true && withdraw.IsChecked == true && savingAmount>= Convert.ToDouble(amountInput.Text))

{

savingAmount = savingAmount - Convert.ToDouble(amountInput.Text);

MessageBox.Show($"Your Amount {amountInput.Text} has been withdrawn succesfully");

balance.Content = "Saving Balance "+savingAmount.ToString();

}

else if (saving.IsChecked == true && Deposit.IsChecked == true )

{

savingAmount = savingAmount + Convert.ToDouble(amountInput.Text);

MessageBox.Show($"Your Amount {amountInput.Text} has been Deposited succesfully");

balance.Content = "Saving Balance " + savingAmount.ToString();

}

else if (chequing.IsChecked == true && withdraw.IsChecked == true && chequingAmount >= Convert.ToDouble(amountInput.Text))

{

chequingAmount = chequingAmount - Convert.ToDouble(amountInput.Text);

MessageBox.Show($"Your Amount {amountInput.Text} has been withdrawn succesfully");

balance.Content = "Chequing Balance " + chequingAmount.ToString();

}

else if (chequing.IsChecked == true && Deposit.IsChecked == true)

{

chequingAmount = chequingAmount + Convert.ToDouble(amountInput.Text);

MessageBox.Show($"Your Amount {amountInput.Text} has been Deposited succesfully");

balance.Content = "Chequing Balance " + chequingAmount.ToString();

}

else

{

MessageBox.Show("this operation cannot be done"," ", MessageBoxButton.OK,MessageBoxImage.Error);

balance.Content = "Chequing Balance " + chequingAmount.ToString()+'\n'+ "Saving Balance " + savingAmount.ToString();

}

}

else if(messageBoxResult==MessageBoxResult.Cancel)

{

MessageBox.Show("this operation has been canceled succesfully");

balance.Content = chequingAmount.ToString();

}

}

catch

{

MessageBox.Show("please Enter valid value");

balance.Content = chequingAmount.ToString();

}

}

}

}

**XAML Code:**

<Window x:Class="Assignment\_3.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:Assignment\_3"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid>

<StackPanel HorizontalAlignment="Left" Height="54" Margin="234,78,0,0" VerticalAlignment="Top" Width="151">

<RadioButton x:Name="chequing" Content="Chequing Account" HorizontalAlignment="Left" Margin="0,0,-16.8,0" VerticalAlignment="Top" GroupName="Account" />

<RadioButton x:Name="saving" Content="Saving Account" HorizontalAlignment="Left" Margin="1,0,-1.8,0" VerticalAlignment="Top" GroupName="Account" />

</StackPanel>

<Label Content="Please Select account" HorizontalAlignment="Left" Margin="233,45,0,0" VerticalAlignment="Top"/>

<Label Content="Please Select any Action" HorizontalAlignment="Left" Margin="234,137,0,0" VerticalAlignment="Top"/>

<RadioButton x:Name="withdraw" Content="Withdraw" HorizontalAlignment="Left" Margin="236,167,0,0" VerticalAlignment="Top" GroupName="Action" />

<RadioButton x:Name="Deposit" Content="Deposit" HorizontalAlignment="Left" Margin="236,189,0,0" VerticalAlignment="Top" GroupName="Action" />

<Label Content="Amount :" HorizontalAlignment="Left" Margin="236,209,0,0" VerticalAlignment="Top" />

<TextBox x:Name="amountInput" HorizontalAlignment="Left" Height="23" Margin="298,213,0,0" TextWrapping="Wrap" VerticalAlignment="Top" Width="106" AutomationProperties.HelpText="Please Enter Value"/>

<Button x:Name="Submit" Content="Submit" HorizontalAlignment="Left" Margin="297,245,0,0" VerticalAlignment="Top" Width="75" Click="Button\_Click"/>

<Label Content="Account Information" HorizontalAlignment="Left" Margin="241,291,0,0" VerticalAlignment="Top"/>

<Label Content="Account Balance:" HorizontalAlignment="Left" Margin="237,322,0,0" VerticalAlignment="Top" />

<Label x:Name="balance" Content="Balance" HorizontalAlignment="Left" Margin="341,322,0,0" VerticalAlignment="Top" />

<Button x:Name="Close" Content="Close" HorizontalAlignment="Left" Margin="341,361,0,0" VerticalAlignment="Top" Width="75" ToolTip="Close" Click="Button\_Close"/>

</Grid>

</Window>