**Module 01 – Event Venue**

Tim Mastarone

Rasmussen University

Microsoft C# Programming

Instructor: Jim Barringer

Module 01 - Lab 1

April 3, 2024

The image below is a screen shot of my EventVenue C# console app after valid user input:

A screenshot of a computer program

Description automatically generated

If an invalid ticket type is entered (not a digit 1-5), then an error message is displayed:

A screenshot of a computer

Description automatically generated

If a non-integer is entered for the quantity an error message is also displayed:

A screenshot of a computer program

Description automatically generated

The code below is my Program.cs file:

//Event venue ticket price calculator

//prices array correlates to ticketTypes array

int [] prices = {100, 75, 50, 40, 30};

string[] ticketTypes = { "Tier 1", "Tier 2", "Tier 3", "Orchestra", "Floor" };

string userSelection;

int ticketQty, totalPrice, ticketIndex;

//display a welcome message

Console.WriteLine("Welcome to the Ticket Price Calculator.");

Console.WriteLine("This program will ask for some information and provide a cost report.");

//display the user menu

Console.WriteLine("Enter your ticket type from the menu below:");

Console.WriteLine("1: Tier 1");

Console.WriteLine("2: Tier 2");

Console.WriteLine("3: Tier 3");

Console.WriteLine("4: Orchestra");

Console.WriteLine("5: Floor");

Console.WriteLine("");

Console.Write("Your Entry: ");

userSelection = Console.ReadLine();

if (userSelection != null && int.TryParse(userSelection, out ticketIndex))

{

Console.WriteLine("");

Console.WriteLine("Enter the quantity of tickets purchased:");

Console.Write("Your Entry: ");

userSelection = Console.ReadLine();

if (userSelection != null && int.TryParse(userSelection, out ticketQty))

{

if (ticketIndex >= 1 && ticketIndex <= 5) // Ensure ticketIndex is within range

{

ticketIndex -= 1; // decrement to align with displayed menu options 1-5 (arrays indexed starting at 0)

totalPrice = ticketQty \* prices[ticketIndex];

// Calculate the padding needed based on the maximum number of digits in ticketQty

int qtyPadding = Math.Max(1, 21 - ticketQty.ToString().Length); // Minimum width of 2

// Construct the padding string for the quantity column

string qtyPad = new string(' ', qtyPadding);

// Output the table with formatted columns

Console.WriteLine("");

Console.WriteLine("Type of Ticket Ticket Price # of Tickets Sold Cost");

Console.WriteLine($"{ticketTypes[ticketIndex],-18}${prices[ticketIndex],-14}{ticketQty,1}{qtyPad}${totalPrice}");

}

else

{

Console.WriteLine("Invalid ticket type.");

}

}

else

{

Console.WriteLine("Invalid quantity. Please enter a valid number.");

}

}

else

{

Console.WriteLine("Invalid input. Please enter a valid number for ticket type.");

}