1) **DivideByZeroException**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace User

{

class Program

{

static void Main(string[] args)

{

int n, res, Cost;

Console.WriteLine("Enter the Item Cost for n days");

Cost = Convert.ToInt32(Console.ReadLine());

do {

Console.WriteLine("Enter total(n) days");

n = Convert.ToInt32(Console.ReadLine());

try

{

res = Cost / n;

Console.WriteLine("Item Cost Per Day is " + res);

}

catch (DivideByZeroException ex)

{

Console.WriteLine("DivideByZeroException");

}

}

while (n == 0);

{

}

Console.ReadLine();

}

}

}

2) **Invalid Date Exception**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Globalization;

namespace InvalidDateExceptionEbox2

{

class Program

{

static void Main(string[] args)

{

Console.WriteLine("Enter the start date(dd/MM/yyyy hh:mm:ss tt):");

string start = Console.ReadLine();

Console.WriteLine("Enter the end date(dd/MM/yyyy hh:mm:ss tt):");

string end = Console.ReadLine();

string format = "dd/MM/yyyy hh:mm:ss tt";

try

{

DateTime d1 = DateTime.ParseExact(start, format, CultureInfo.InvariantCulture);

DateTime d2 = DateTime.ParseExact(end, format, CultureInfo.InvariantCulture);

Console.WriteLine("Starting Date: " + d1.ToString("dd/MM/yyyy hh:mm:ss tt"));

Console.WriteLine("Ending Date: " + d2.ToString("dd/MM/yyyy hh:mm:ss tt"));

}

catch (FormatException ex)

{

Console.WriteLine("Invalid Date Format...");

}

}

}

}

3) **Seat Booking**

using System;

class SeatNotAvailableException :Exception

{

public string Message;

public SeatNotAvailableException (string Message) :base (Message)

{

this.Message= Message;

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

class Program

{

static void Main(string[] args)

{

try

{

Console.WriteLine("Enter the booking details");

string bookingId = Console.ReadLine();

Console.WriteLine("Enter the seat number to book");

int bookingSeatNo = Convert.ToInt32(Console.ReadLine());

if (bookingSeatNo < bookingId.Length)

{

if (bookingId[bookingSeatNo - 1] == '0')

{

Console.WriteLine("Booked successfully");

}

else

{

throw new SeatNotAvailableException("Seat booked already");

}

}

else

{

throw new SeatNotAvailableException("Array index is out of range.");

}

}

catch(SeatNotAvailableException e)

{

Console.WriteLine(e.Message);

}

}

}

4) **Seat Not Available Exception**

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

class SeatNotAvailableException :Exception

{

public string Message;

public SeatNotAvailableException (string Message) :base (Message)

{

this.Message= Message;

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

class Program

{

static void Main(string[] args)

{

int count = 0, i;

Console.WriteLine("Enter the total number of seats");

int size = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Enter the number of seats to be booked:");

int n = Convert.ToInt32(Console.ReadLine());

int[] arr = new int[size];

for (int j = 0; j < size; j++)

arr[j] = 0;

for (i = 0; i < n; i++)

{

Console.WriteLine("Enter the seat number " + (i + 1));

int seat = Convert.ToInt32(Console.ReadLine());

try

{

if (seat > size)

throw new SeatNotAvailableException("Array index is out of range.");

else

{

if (arr[seat - 1] == 0)

{

arr[seat - 1] = 1;

Console.WriteLine("Seat booked");

}

else

{

throw new SeatNotAvailableException("SeatNotAvailableException: Already Booked");

}

}

}

catch (SeatNotAvailableException e)

{

Console.WriteLine(e.Message);

}

}

if (!arr.Contains(1))

Console.WriteLine("No seats booked");

else

{

Console.WriteLine("The seats booked are:");

for (i = 0; i < size; i++)

{

if (arr[i] == 1)

{

Console.WriteLine(i + 1);

}

}

}

Console.ReadLine();

}

}

5) **Mobile Number Validation**

using System;

public class ContactDetail

{

private long \_mobile;

private long \_alternateMobile;

private long \_landLine;

private string \_email;

private string \_address;

public ContactDetail()

{

}

public ContactDetail(long mobile,long alternateMobile,long landLine,string email,string address)

{

this.\_mobile = mobile;

this.\_alternateMobile = alternateMobile;

this.\_landLine = landLine;

this.\_email = email;

this.\_address = address;

}

public long Mobile

{

get

{

return this.\_mobile;

}

set

{

this.\_mobile = value;

}

}

public long AlternateMobile

{

get

{

return this.\_alternateMobile;

}

set

{

this.\_alternateMobile = value;

}

}

public long LandLine

{

get

{

return this.\_landLine;

}

set

{

this.\_landLine = value;

}

}

public string Email

{

get

{

return this.\_email;

}

set

{

this.\_email = value;

}

}

public string Address

{

get

{

return this.\_address;

}

set

{

this.\_address = value;

}

}

public override String ToString()

{

Console.WriteLine("Contact Details:");

return "Mobile: " + this.Mobile +"\n"+ "Alternate Mobile: " + AlternateMobile +"\n"+ "LandLine: " + LandLine +"\n"+ "Email Id: " + Email +"\n"+ "Address: " + Address+"\n";

}

}

using System;

public class DuplicateNumberException:Exception

{

public string message;

public DuplicateNumberException(string message) : base(message)

{

this.message = message;

}

}

using System;

public class ContactDetailBO

{

public void Validate(ContactDetail cd)

{

if (cd.Mobile != cd.AlternateMobile)

{

Console.WriteLine(cd.ToString());

}

else

{

throw new DuplicateNumberException("Exception: Same Mobile no and Alternate Mobile no.");

}

}

}

using System;

public class Program

{

public static void Main()

{

ContactDetail cd = new ContactDetail();

Console.WriteLine("Enter the mobile number:");

cd.Mobile = Convert.ToInt64(Console.ReadLine());

Console.WriteLine("Enter the alternate mobile number:");

cd.AlternateMobile = Convert.ToInt64(Console.ReadLine());

Console.WriteLine("Enter the landline number:");

cd.LandLine = Convert.ToInt64(Console.ReadLine());

Console.WriteLine("Enter the email id:");

cd.Email = Console.ReadLine();

Console.WriteLine("Enter the address:");

cd.Address = Console.ReadLine();

ContactDetailBO cd1 = new ContactDetailBO();

try

{

cd1.Validate(cd);

}

catch(DuplicateNumberException e)

{

Console.WriteLine(e.message);

}

Console.ReadLine();

}

}