1. Write a JavaScript program to check whether a string entered in a textbox contains valid

PANCARD number or not

Format: 5 Character,4 Numbers, 1 Character

Ex:- AGSFE1245A

// html file

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title></title>

<script src="/js/JavaScript.js"></script>

</head>

<body>

<section>

<label id="error">PANCARD NO:</label>

<input type="text" id="pan" name="pancard" />

<input type="button" id="button" value="submit" name="submit" onclick="validatePAN()" />

</section>

</body>

</html>

//javascript file

function validatePAN()

{

var P = document.getElementById("pan");

if (P.value != "") {

objVal = obj.value;

var reg = /^([a-zA-Z]{5})(\d{4})([a-zA-Z]{1})$/;

if (objVal.search(reg) == -1) {

alert("NOT VALID PANCARD");

P.focus();

return false;

}

else

{

alert("VALID PANCARD");

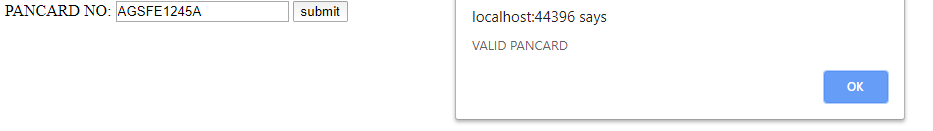
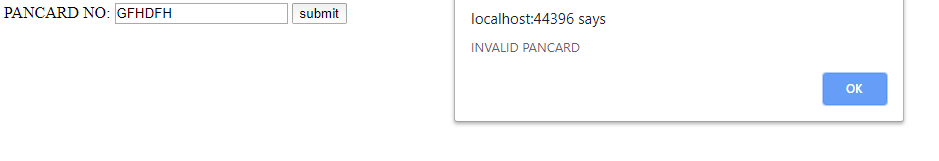
}

}

}

OUTPUT –

SCREENSHOT OF OUTPUT FOR VALID AND NOT VALID PANCARD RESULT.



2. Create the following table

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title></title>

<style>

table, th, td, tr {

border: 1px solid black;

}

</style>

</head>

<body>

<table cellspacing="0">

<tr>

<th rowspan="2">DeptName</th>

<th colspan="3"> Deptno </th>

</tr>

<tr>

<td>Male</td>

<td>Female</td>

<td>Total</td>

</tr>

<tr>

<td>Sales</td>

<td>19</td>

<td>74</td>

<td>93</td>

</tr>

<tr>

<td>Marketing</td>

<td>20</td>

<td>88</td>

<td>93</td>

</tr>

<tr>

<td>Finance</td>

<td>45</td>

<td>54</td>

<td>99</td>

</tr>

<tr>

<td>Accounts</td>

<td>52</td>

<td>61</td>

<td>113</td>

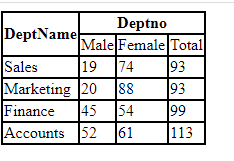
</tr>

</table>

</body>

</html>

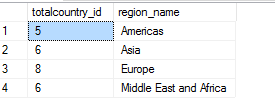
OUTPUT –



# SQL SERVER

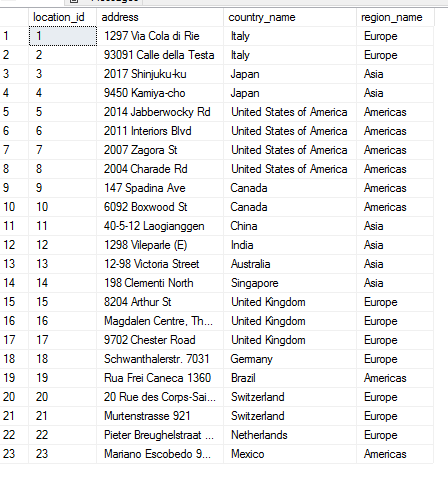
--1.--display the count of countries region\_name

select count(country\_id) as totalcountry\_id, region\_name from countries c inner join regions r on c.region\_id= r.region\_id group by region\_name ;



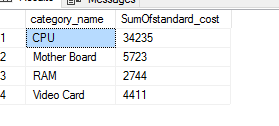
--2. display location\_id, address,country\_name , region\_name

select location\_id, address,country\_name, region\_name from countries c inner join regions r on c.region\_id= r.region\_id inner join locations l on l.country\_id= c.country\_id



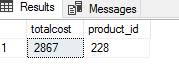
--3 . display category\_name, sum of standardCost

select category\_name, SUM(standard\_cost) as SumOfstandard\_cost FROM product\_categories pc inner join productinfo pi on pc.category\_id= pi.category\_id group by category\_name



--4. find the sum of standard\_Cost as TotalCost, product\_Id wise whose sum(standard\_cost)>2500

select sum(standard\_cost)as totalcost, product\_id from productinfo group by product\_id having sum(standard\_cost)>2500;



**C#**

1.

using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp7

{

class Student

{

private int StudentNo;

private string StudentName;

private double Maths;

private double Physics;

private double Chemistry;

private double Total;

private double Average;

ArrayList StudentDetails = new ArrayList();

public void Accept(int n)

{

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

{

Student s = new Student();

s.StudentNo = i + 1;

Console.WriteLine("Enter the name of student");

s.StudentName = Console.ReadLine();

Console.WriteLine("Enter the Math marks of student");

s.Maths = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter the Physics marks of student");

s.Physics = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter the Chemistry marks of student");

s.Chemistry = Convert.ToDouble(Console.ReadLine());

s.Total = s.Maths + s.Physics + s.Chemistry;

s.Average = s.Total / 3;

StudentDetails.Add(s);

}

}

public void Displaystudents()

{

foreach (Student student in StudentDetails)

{

Console.WriteLine("Student {0} Details are as follows", student.StudentNo);

Console.WriteLine("Student Number: " + student.StudentNo);

Console.WriteLine("Student Name: " + student.StudentName);

Console.WriteLine("Maths Marks: " + student.Maths);

Console.WriteLine("Physics Marks: " + student.Physics);

Console.WriteLine("Chemistry Marks: " + student.Chemistry);

Console.WriteLine("Total Marks: " + student.Total);

Console.WriteLine("Average Marks: " + student.Average);

}

}

public void HighestScore()

{

foreach (Student student1 in StudentDetails)

{

List<double> Score = new List<double>();

Score.Add(student1.Average);

}

}

}

}

………… program.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp7

{

class Program

{

static void Main(string[] args)

{

Student stu = new Student();

int No\_Of\_Students;

Console.WriteLine("Enter the no. of Students");

No\_Of\_Students = Convert.ToInt32(Console.ReadLine());

stu.Accept(No\_Of\_Students);

stu.Displaystudents();

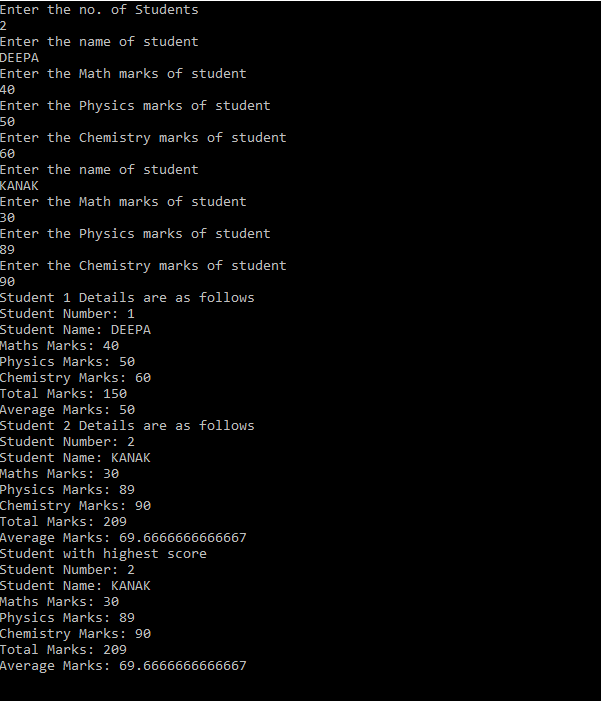
stu.HighestScore();

Console.ReadKey();

}

}

}



2.

Create a program to accept Category\_id from the console application

Display the Product\_Id,Product\_Name,Description,Standard\_Cost,List\_price though list

PROGRAM .CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp8

{

class Program

{

static void Main(string[] args)

{

Product p = new Product();

int No\_Of\_Products;

Console.WriteLine("Enter the no. of Products");

No\_Of\_Products = Convert.ToInt32(Console.ReadLine());

for (int i = 0; i < No\_Of\_Products; i++)

{

ProductDetails pd = new ProductDetails();

pd.Category\_id = i + 1;

Console.WriteLine("Enter the id of product");

pd.Product\_id = Convert.ToInt32(Console.ReadLine());

Console.WriteLine("Enter the name of product");

pd.Product\_Name = Console.ReadLine();

Console.WriteLine("Enter the decription of product");

pd.Description = Console.ReadLine();

Console.WriteLine("Enter the standard cost of product");

pd.Standard\_Cost = Convert.ToDouble(Console.ReadLine());

Console.WriteLine("Enter the list price of product");

pd.List\_price = Convert.ToDouble(Console.ReadLine());

p.AddProduct(pd);

}

Console.WriteLine("Enter the category id for which you want to see the details"); // category id increment automatically depends on no of products

// eg. for product 1 category id is 1

// for product 2 category id is 2 etc.

int cat\_id = Convert.ToInt32(Console.ReadLine());

p.Displayproduct(cat\_id);

Console.ReadKey();

}

}

}

ProductDetails.CS

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp8

{

class ProductDetails

{

public int Category\_id { get; set; }

public int Product\_id { get; set; }

public string Product\_Name { get; set; }

public string Description { get; set; }

public double Standard\_Cost { get; set; }

public double List\_price { get; set; }

}

}

Product.CS

using System;

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp8

{

class Product

{

ArrayList productdetails = new ArrayList();

public void AddProduct(ProductDetails pdetails)

{

productdetails.Add(pdetails);

}

public void Displayproduct(int id)

{

foreach (ProductDetails pd1 in productdetails)

{

if (pd1.Category\_id == id)

{

Console.WriteLine("Product Details are as follows");

Console.WriteLine("Product Id: " + pd1.Product\_id);

Console.WriteLine("Product Name: " + pd1.Product\_Name);

Console.WriteLine("Description: " + pd1.Description);

Console.WriteLine("Standard Cost: " + pd1.Standard\_Cost);

Console.WriteLine("List Price: " + pd1.List\_price);

}

}

}

}

}

