Tutorial – while.. Loops

Q1

Write a while loop to print all the whole numbers between 1 and 20

Modify to sum all the numbers between 1 and 20

int counter = 1;

int total = 0;

while(counter < 21)

{

total += counter;

counter++;

}

Console.WriteLine($”The total is {total}”);

Q2

What value will sum contain after the following

int sum = 0;

int i = 1;

while (i <= 15)

{

sum += i; same as sum = sum + i;

i+=3; // same as i = i + 3;

}

I =1 4 7 10 13 16

Sum = 1 5 12 22 35

OUTPUT: 35

Q3

First correct any syntax errors. What value will the variable sum have after the execution of the code

int sum = 50;

j = 2;

while (j <12) // deleted ;

{

sum += j;

j += 3;

}

j = 2 5 8 11 14

Sum = 50 52 57 65 76

OUTPUT: 76

Q4

What value will the variable sum have after the execution of the code

int sum = 100;

int j = 10;

while (j > 7)

{

sum -= j; // same as sum = sum –j;

j-–;

}

Sum = 100 90 81 73

J = 10 9 8 7

OUTPUT: 73

Q5

Find the value of the variable total after the execution of the code

int i = 5;

int total = 100;

while (i < 20)

{

total -= i;

i += 4;

}

I= 5 9 13 17 21

total = 100 95 86 73 56

OUTPUT: 56

Q6

Write an if statement that check if a number is an odd number, then write a while statement to display the odd numbers from x through to y (both integers)

Console.WriteLine(“Enter a Number : “);

int num = int.Parse(Console.ReadLine());

if(!(num % 2 = 0))

{ Console.WriteLine($”{num{ is an odd number”); }

else

{

Console.WriteLine($”{num{ is an even number”);

}

int count = 0;

Console.WriteLine(“Enter a Start Number : “);

int x = int.Parse(Console.ReadLine());

Console.WriteLine(“Enter an End Number : “);

int y = int.Parse(Console.ReadLine());

while(count < = y)  
{

if(!(num % 2 = 0))

{

Console.WriteLine($”{num{ is an odd number”);

}

}

Q7

Write a C# program to find the number of months needed to pay off a loan, where each month interest is applied to the balance and a payment is deducted from the balance. Input the amount of the loan, the size of the monthly payment, and the monthly interest rate.

decimal amount = 0.0m, monthlyRepay = 0.0m, interestRate = 0.0m, balance = 0.0m;

int numMths = 0;

Console.WriteLine(“Enter amount of loan : “);

amount = decimal.Parse(Console.ReadLine());

balance = amount;

Console.WriteLine(“Enter amount of repayment per month : “);

monthlyRepay = decimal.Parse(Console.ReadLine());

Console.WriteLine(“Enter interest rate : “);

interstRate = decimal.Parse(Console.ReadLine());

interestRate = interestRate /100;

for(numMths =0; balance >= amount; numMths++)

{

balance -= monthlyRepay;

balance = balance + (amount \* (interestRate);

}

Console.WriteLine($“It will take {numMths} months to repay the loan “);

Q8

What do you understand by a *sentinel* value? Write a piece of code to demonstrate how one might be used to find the average of a series of exam scores.

A Sentinel Value is a value that the compiler ‘listens for’ and used as the condition to end a loop.

e.g. Press -999 to quit:

const string QUIT\_VALUE = “-999”; // set the sentinel to -999

/\* loop will run and prompt user to enter values until -999 (the sentinel value

\* assigned to QUIT\_VALUE constant) is entered \*/

int count = 0;

double total = 0;

while(x != QUIT\_VALUE)

{

Console.WriteLine(“Enter a Test Score (or – 999 to exit) : “);

int result = int.Parse(Console.ReadLine());

total += result; // accumulate the subtotal

count++; // increment the counter

}

// calculate the average

double avgScore = total / count;

// display avg to 2 decimals

Console.WriteLine($“\nThe average test score is {avgScore:f2}”);