Starta programmet Bastun

Skriv ut”Mata in ett Fahrenheit tal”: // Spara i en int

Omvandla Fahrenheit till Celsius // med en Metod som genererar en int

DO

{

OM inmatat värde är < 73 C

Skriv ut ”skruva upp värmen”

ANNARS OM inmatat värde är >77C

Skriv ”Det är för varmt, nu får vi vänta lite”  
 Skriv”Öppna dörren.”

Grader --dateTime, wait

ANNARS

Skriv ”Nu kan vi basta”

}While (temp = temp + värde)

TRY CATCH

Int tal = 0;

bool loop(true);

Do

{

CW(“Skriv in temperature”);

}

try

{

tal = int.Parse(Console.Readline();

}

catch

{

CW(“Fel inmatning det där är inte ett heltal”);

}

while (loop==true); // kan oxå skrivas som while (loop), Det är precis samma sak. Dvs true.

CW(”Du har skrivit in ett korrekt tal”)

Console.ReadLine();

// Avancerad TRY CATCH Om användaren skriver in ngt annat än ett heltal

Int mynum = 0;

While (!int.Parse(Console.Readline(), out mynum)

{

Skriv “ Fel inmatning det där är inte ett heltal”

}

Console.WriteLine(mynum);

Extra för betyg C

Eftersom temperaturen i formeln kan generera ett decimaltal i

Celsius så behöver vi en avrundare till närmsta heltal

Amerikanen i bastun Denna uppgift går ut på att en amerikanare ska skriva in en temperatur för ett bastuaggregat tills man skriver in en temperatur som anses lagom. För att kontrollera detta ska vi ha en temperatur som anger den lägsta godtagbara temperaturen och en som anger den högsta godtagbara temperaturen. Tyvärr förstår vår kära bastubesökare enbart Fahrenheit medan bastuaggregatet enbart förstår Celsius.

Därför har du fått skriva ett program som konverterar det som skrivs in från Fahrenheit till Celsius. Den optimala temperaturen för bastun är 75 grader men den godtagbara temperaturen är från 73 till 77 grader. Detta ska styras hela i villkorssatsen i loopen. Om talet är mindre än den minsta godtagbara temperaturen ska programmet skriva att det är för kallt och man får skruva upp lite. Därefter får bastubesökaren ställa in ett nytt värde på bastuaggregatet och loopen upprepas.

I denna uppgift ska ni hantera följande:

1. Undantagshantering

2. Metoder

3. Villkor Då amerikanaren kan skriva in i Fahrenheit så innebär det ett värde på 167 grader (=75 grader) men det vet ju inte han eller hon då det är deras första bastubesök någonsin.

Betyg E

Be användaren skriva in en temperatur i Fahrenheit Detta värde ska skickas till en metod som gör om Fahrenheit till Celsius: Kodhjälp i koden till höger. Denna metod heter FahrToCels och returnerar ett heltal. Metoden tar emot ett värde som användaren skrivit in. Därefter ska du komplettera med koden som krävs för att användaren ska få skriva in en ny temperatur i Fahrenheit om det är för kallt eller för varmt. Det innebär att du ska lösa det med loopar och villkor på ett lämpligt sätt.

Här kommer en förklaring till pilarna:

1. Vi skapar en heltalsvariabel (celsius) som vi ska tilldela ett värde.

2. Vi anropar en metod FahrToCels där vi skickar med värdet fahrenheit (exempelvis 140) som metoden behöver för att fungera.

3. Detta värde (140) sparas som fahr i metoden och används för uträkningen av värdet cel som returneras och lagras i variabeln celcius.

Betyg C

Omräkningen från Fahrenheit till Celsius ovan är inte optimal då det uträknade värdet för Celsius troligtvis blir ett decimaltal och detta ”kapas” rakt av då detta värde ovan lagras som heltal (int). Det avrundas alltså inte. Lös uppgiften och att värdet Celsius sparas som ett flyttal med decimaler. För att en beräkning ska fungera med decimaltal måste alla tal i formeln vara av flyttalstyp. Denna formel kommer inte att fungera då fahr är ett heltal. double temp = (fahr - 32) \* 5 / 9; Här behöver man alltså omvandla fahr till samma double (eller float om ni jobbar med det). Ett sätt att göra detta på är att ”typomvandla” fahr i formeln ovan så det behandlas som en double i formeln.

För att lösa detta kan man jobba med så kallad explicit typomvandling som beskrivs på MSDNs hemsida med ett enkelt exempel på följande länk: https://msdn.microsoft.com/en-us/library/ms173105.aspx#Anchor\_1 1 2 3

Jag vill fortfarande att det tal användaren skriver in lagras i en int först eftersom man med stor sannolikhet bara skriver in heltal. Det innebär att metoden ska se ut så här: public static double FahrToCels(int fahr) { //Typomvandla fahr till flyttal //Gör uträkning med decimaler i svaret return temp; } Utöver detta fungerar programmet på ett liknande sätt förutom att det är ett decimaltal som testas mot en lämplig temperatur. Se om du också kan lösa det med att avrunda talet om du väljer att skriva ut det.

Programmet ska också kunna hantera fel som uppstår med try & catch. Ett vanligt fel man gör när man jobbar med try och catch är att man lägger för mycket kod i try-blocket. Egentligen ska bara själva typ-omvandlingen ligga här. Se exemplet på sidan 87 i läroboken eller se filmen om undantagshantering.

Betyg A För betyget A ska uppgiften lösas enligt kraven för betyget C på ett säkert och lämpligt sätt. Utöver detta ska amerikanen kunna välja att skriva in en nolla för att slumpa en temperatur. Om detta sker ska inget värde skickas till metoden – detta kan man lösa med metod-överlagring som beskrivs i studieguiden