

# Looping



# Three Doors Game - The Monty Hall Problem

[Video](#) [Video2](#)



Is there any advantage or disadvantage to switching?

A Simulation

# Find the Disney Star in this video?

- Green ticket to the first with the right answer...
- Make sure you use BQ in the subject
- Video
- Movie clip from '21'



# Looping Statements: Doing something more than once

- While loop
- Do-While loop
- For loop



# While loop syntax


```
while (<Boolean expression>)  
{  
    <statements>  
}
```



# While Loop Example

```
int count = 0;
const int max = 5;
int sum = 0;

while (count < max)
{
    sum = sum + count;
    count = count + 1;
}
labelAnswer.Text = Convert.ToString(sum);
```



## Do-While loop syntax:

```
do
{
    <statements>
} while (<Boolean expression>);
```



# Do-While Loop Example

```
int count = 0;
const int max = 5;
int sum = 0;

do
{
    sum = sum + count;
    count = count + 1;
} while (count < max);
labelAnswer.Text = Convert.ToString(sum);
```





## For loop syntax:

```
for (<initialization statement>;  
    <Boolean expression>;  
    <iterator statement>)  
{  
    <statements>  
}
```



For loop steps: 1-2-3, 4-2-3, 4-2-3, ...

```
1 → for (<initialization statement>;  
2 →   <Boolean expression>;  
   <iterator statement>) ← 4  
  {  
3 →   <statements>  
  }
```



# For Loop Example

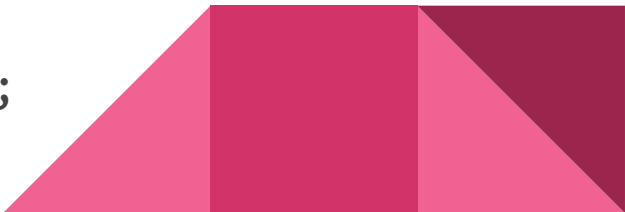
```
const int max = 5;  
int sum = 0;  
  
for (int count = 0; count < max; count++)  
{  
    sum = sum + count;  
}  
labelAnswer.Text = Convert.ToString(sum);
```



# More Examples


```
int count = 1;
const int max = 100;
int sum = 0;

while (sum < max)
{
    sum = sum + count;
    count = count + 1;
}
labelAnswer.Text = Convert.ToString(count);
```



# More Examples

```
Random random = new Random();
int count = 0;
int diceValue1;
int diceValue2;
do
{
    diceValue1 = random.Next(1, 7);
    diceValue2 = random.Next(1, 7);
    count = count + 1;
} while (diceValue1 != diceValue2);
labelAnswer.Text = Convert.ToString(count);
```



# But Wait! There's More!!!

- break;
- continue;



# Break Statement: How to get out early

- Break syntax:  
`break;`



# Continue Statement: How to skip this one

- Continue syntax:  
`continue;`





# Looping Assignment

The screenshot shows a Windows application window titled "Looping Assignment" with three distinct panels. The first panel, "X to the Nth Power", contains input fields for "X" (value 3) and "N" (value 2), a "Calculate" button, and the output "Answer: 9". The second panel, "Loan Calculator", contains input fields for "Loan amount" (value 10000) and "Monthly payment" (value 250), a "Calculate" button, and the output "Loan paid off in 40 months.". The third panel, "Hailstone Sequence", contains an input field for "Starting value" (value 11), a "Start Sequence" button, and a text area displaying the sequence: "11 34 17 52 26 13 40 20 10 5 16 8 4 2 1".

Panel	Input	Output
X to the Nth Power	X: 3, N: 2	Answer: 9
Loan Calculator	Loan amount: 10000, Monthly payment: 250	Loan paid off in 40 months.
Hailstone Sequence	Starting value: 11	11 34 17 52 26 13 40 20 10 5 16 8 4 2 1