Decisions

- Relational and Logical Operators
- If Blocks
- Select Case Blocks

Relational and Logical Operators

- Relational Operators
- ANSI Values
- Logical Operators
- Boolean Data Type

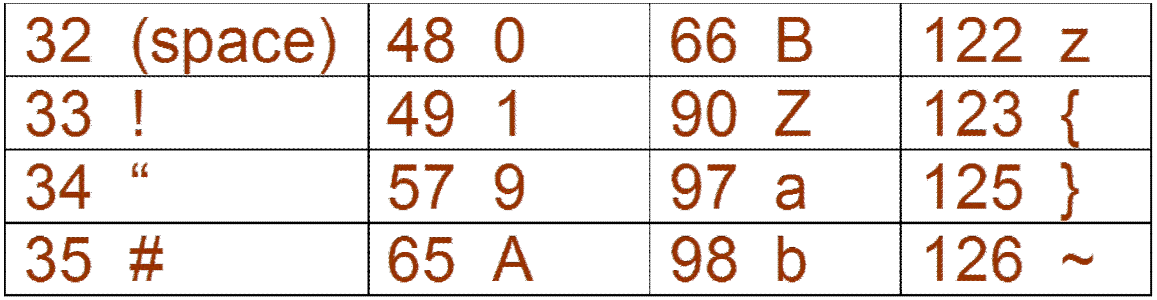
Relational Operators

<	less than
<=	less than or equal to
>	greater than
>=	greater than or equal to
=	equal to
<>	not equal to

ANSI values are used to decide order for strings.

ANSI Character Set

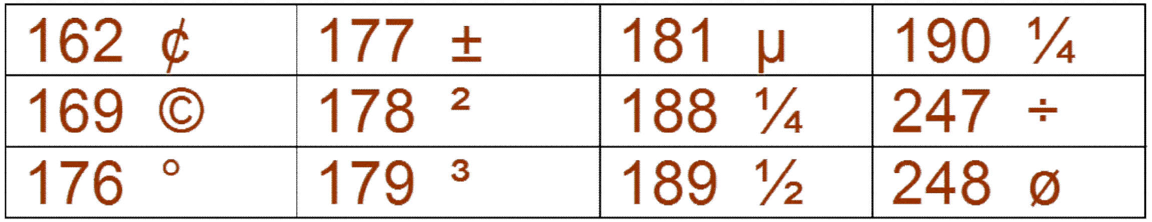
- A numeric representation for every key on the keyboard and for other assorted characters.



32 (space)	48 0	66 B	122 z
33 !	49 1	90 Z	123 {
34 “	57 9	97 a	125 }
35 #	65 A	98 b	126 ~

ANSI Character Set: continued

- A numeric representation for every key on the keyboard and for other assorted characters.



162 ¢	177 ±	181 µ	190 ¼
169 ©	178 ^{2}	188 ¼	247 ÷
176 °	179 ^{3}	189 ½	248 ø

Chr Function

For n between 0 and 255,

```
Chr(n)
```

is the string consisting of the character with ANSI value *n*.

EXAMPLES: `Chr(65) is "A"`

`Chr(162) is "¢"`

Asc Function

For a string *str*,

Asc(str)

is ANSI value of the first character of *str*.

EXAMPLES: **Asc("A") is 65**

Asc("¢25") is 162

Boolean Data Type

- An expression or variable that evaluates to either True or False is said to have Boolean data type.
- *Example:*

  The statement

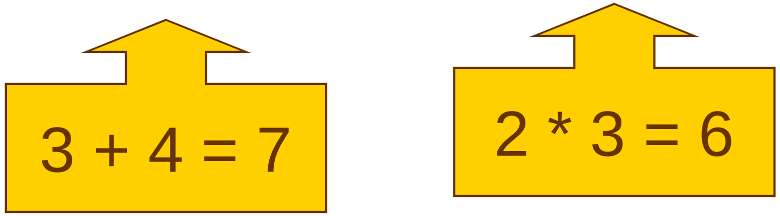
  ```
  txtBox.Text = (2+3)<6
  ```

  displays **`True`** in the text box.

Example

When a = 3, b = 4

(a + b) < 2 * a



3 + 4 = 7

2 * 3 = 6

7 is NOT less than 6 and the value of the expression is False

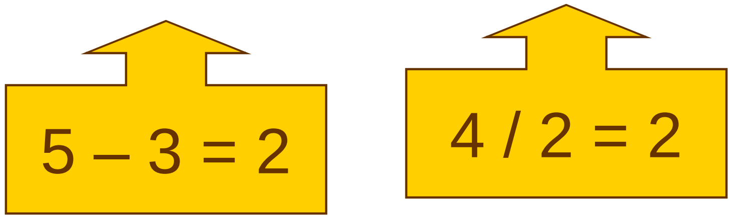
Another Example

a = 4

b = 3

c = 5

(c – b) = (a / 2)



5 – 3 = 2

4 / 2 = 2

True because 2 equals 2

Relational Operator Notes

- Relational operators require an operand on both sides of the operator
- Value of a relational expression will always be True or False
- Expressions are evaluated from left to right with no order of operations