

ANNOUNCEMENTS

- ❖ No posting of code in the forum
- ❖ Check class announcements daily

INCREMENT/DECREMENT OPERATORS

❖ $++ \rightarrow$ increases value by one

❖ $x++ \rightarrow x = x + 1$

❖ $-- \rightarrow$ decreases value by one

❖ $x-- \rightarrow x = x - 1$

❖ Pre/post version

❖ $++x$ VS. $x++$

❖ $--x$ VS. $x--$

DO WHILE STATEMENT

- ❖ do while statement – Allows repetition of a set of statements
 - ❖ Can be seen as “do as long as”

Basic Form

```
do {  
    statement // executed as long as expression is true  
} while (expression);
```

- ❖ Notice the semicolon after the expression in parenthesis
- ❖ Executes the statement at least once
- ❖ You don't need the { } if you only need to execute one statement
- ❖ **Example:** DoWhileNumbers.html
- ❖ **Example:** DoWhile.html
- ❖ Any type of statements (including do whiles) in a do while
- ❖ When to use a do while?
- ❖ When to use a while?

EMPTY STATEMENT

- ❖ Empty statement: represented by a semicolon
- ❖ It does nothing
- ❖ Example:

```
if (x === 100)
    ;
else {
    // task
}
```

- ❖ Notice we don't want to have code like the one above. We could rewrite it as follows

```
if (x !== 100)
    // task
```

- ❖ If you add a semicolon to a while statement (not a do while) you may generate an infinite loop. Be careful

CODING EXAMPLE (PASSWORD READING)

- ❖ Let's write a program that reads a password and allows a maximum of two attempts
 - ❖ We will start with just reading the password and looping until a valid password is provided

IN-CLASS LAB (20 MINUTES)
