



Java

Amber Wagner, Birmingham-Southern College

CAC 210

Review

Decision Statements

```
// Statements that execute before the branches

if (expression) {
    // Statements that execute when expression is true (first branch)
}
else {
    // Statements that execute when expression is false (second branch)
}

// Statements that execute after the branches
```


If-else-if

```
import java.util.Scanner;

public class MultIfElseAnniv {
    public static void main(String[] args) {
        Scanner scnr = new Scanner(System.in);
        int numYears;

        System.out.print("Enter number years married: ");
        numYears = scnr.nextInt();

        if (numYears == 1) {
            System.out.println("Your first year -- great!");
        }
        else if (numYears == 10) {
            System.out.println("A whole decade -- impressive.");
        }
        else if (numYears == 25) {
            System.out.println("Your silver anniversary -- enjoy.");
        }
        else if (numYears == 50) {
            System.out.println("Your golden anniversary -- amazing.");
        }
        else {
            System.out.println("Nothing special.");
        }
    }
}
```

Loops

- ❖ Different types of loops...can you name them?

Loops

- ❖ Different types of loops...
 - ❖ while
 - ❖ do while
 - ❖ for

While Loops

```
while (expression) { // Loop expression
    // Loop body: Executes if expression evaluated to true
    // After body, execution jumps back to the "while"
}
// Statements that execute after the expression evaluates to false
```

While Loops

```
import java.util.Scanner;

public class CountUp {
    public static void main(String[] args) {
        Scanner scnr = new Scanner(System.in);
        int currPower;
        char userChar;

        currPower = 2;
        userChar = 'y';

        while (userChar == 'y') {
            System.out.println(currPower);
            currPower = currPower * 2;
            userChar = scnr.next().charAt(0);
        }

        System.out.println("Done");
    }
}
```

do while

```
do {  
    // Statements  
}while(Boolean_expression);
```


for loops

```
for (initialExpression; conditionExpression; updateExpression) {  
    // Loop body  
}  
// Statements after the loop
```

```
int i;  
  
i = 0;  
while (i < 5) {  
    // Loop body  
    i = i + 1;  
}
```

```
int i;  
  
for ( i = 0; i < 5; i = i + 1 ) {  
    // Loop body  
}
```


Let's write a program that reads in numbers from the user and prints out the sum and the average

Classes

What is a class?

First things first...

- ❖ We have to name the class. The filename and the class name must be the same.

```
public class _____ {
```

```
}
```

Next, private data

- ❖ What do you think private means?
- ❖ We need to think about what data we need for our class that should be kept private.
- ❖ For example:
 - ❖ BankAccount: balance, owner
 - ❖ Employee: Name, ID_Number, Salary

Constructor

- ❖ How do we construct the object?
- ❖ It is a public method...why?
- ❖ It has the same name as the class
- ❖ It can have parameters or not

Remind me:
What's
a parameter?

Constructor

- ❖ Default constructor
(no parameters)

```
public Employee() {  
    name = "";  
    idNumber = "";  
    salary = 0.0;  
}
```

- ❖ Non-default constructor
(parameters)

```
public Employee(String n, String num,  
double sal) {  
    name = n;  
    idNumber = num;  
    salary = sal;  
}
```

Methods

- ❖ What behaviors does the class need to be able to perform?
- ❖ Methods / functions can be public or private
- ❖ They can manipulate private data or just reference it
- ❖ What methods might we want for Employee?

Methods

```
public returnType name (parameters) {  
}
```

```
public void changeName(string newName) {  
    name = newName;  
}
```

```
public double getSalary() {  
    return salary;  
}
```


Let's write a bank account class and
a tester for the class