Today's topics

- Subroutines
- Iteration
- Arrays

- Upcoming
  - More Java programming

Repeating code

- Repeating code is bad
- Writing repetitive code is tedious
- Debugging repetitive code is hard
- Avoid repeating code through:
  - Subroutines/methods
  - Loops

Loops

- If statements need to repeat, then you probably need a loop
- Describe portion of program as:
  - Repeat
  - Continue until
  - For each value from 1 to n
  - For every object of a set, do something

- We have already used iteration by using the buttons
  - How?

Problems

- We want to:
  - Print out all numbers from 0 up to 100 incrementing by 0.5 each time
  - Sum up the numbers from 1 to 100
  - ...

- New Java syntax
  - New object type TextArea which is basically a big scrolling textbox
  - tArea is 80 character wide and 20 rows high text box with 20 rows
    - TextArea tArea = new TextArea(20, 80);
  - Add characters to the end of the TextArea using append
    - tArea.append("Hello\n");
  - ‘\n’ is called a newline character which moves the next character to the next line
Anatomy of a while loop

- While loops are one way to get rid of repetitive code
- Print out numbers up to 100 by increments of 0.5

```
x = 0.0;
while (x < 100)
{
x = x + 0.5;
tArea.append("x = " + x);
tArea.append("\n");
}
```

Another loop

- Summing the numbers 1 ... 100
  ```
  int sum = 0;
  int k = 0;
  while (k < 100)
  {
    k = k + 1;
    sum = sum + 1;
  }
  ```

- Other Loop designs
  ➤ Count down
  ➤ Stopping and starting at computed values
  ➤ Data dependent loop

Arrays

- Aggregate data type
- Deal with items of same type
  ➤ Lists
  ➤ numbers
  ➤ words ...
- Analogies
  ➤ Mailboxes in post office
  ➤ CD racks with slots
- Simplifies naming
- Allows use of loops
- Required for many mathematical and statistical problems
- Multiple elements or cells

Using arrays

- *subscribe* or *index* to access element
  ```
x[5] = 20;
foo.setText("Result is " + x[5]);
```  
- Often used in loops
  ```
  int k = 0; sum = 0;
  while ( k < 10 )
  {
    k = k + 1;
    sum = sum + name[k];
  }
  ```
**Creating Arrays**

- **Declaration**
  ```java
double weights[];
```
- **Definition**
  ```java
weights = new double[50];
```
- **Combine**
  ```java
double weights[] = new double[50];
```

```java
int num[] = new int[6];
```

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

**Arrays & Loops**

```java
int k = 2;
while(k<6)
{
    num[k] = k*k;
    k = k+1;
}
```

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>4</td>
<td>9</td>
<td>16</td>
<td>25</td>
<td></td>
</tr>
</tbody>
</table>