Strings, char, and final

CST 231 - Week 13

Today's Topics:

char datatype

Strings and String

methods

final variables

Part A - char Datatype

char Holds Character Data... Sort Of

Actually: char stores a small integer (0 to 65,535)

```
char b = 'd';   // variable b holds value 100
char c = 100;   // variable c holds value 100
```

Both store the same value! When printed, Java uses ASCII/Unicode to display as a character.

ASCII values you need to know:

'0'-'9' start at 48

'A'-'Z' start at 65

char and ASCII for Friday's Lab

Friday's tic-tac-toe lab needs this pattern:

```
// Convert position 1-9 to character '1'-'9'
int position = 5;
char display = (char)(position + 48);  // '5'
```

Why 48? Because '0' is ASCII 48

1 + 48 = 49 = '1'

5 + 48 = 53 = '5'

9 + 48 = 57 = '9'

Part B - Strings

Strings Are Arrays of char... Sort Of

- String is a **class** (more in CST 238)
- String variables have useful **methods**
- Strings are **immutable** (cannot be changed)

Useful String Methods

```
String s = "hello";

// Comparison
s.equals("hello")             // true
s.equalsIgnoreCase("HELLO")   // true

// Get a character
char c = s.charAt(1);         // 'e'

// Transform
String upper = s.toUpperCase();  // "HELLO"

// Search
s.contains("ll")              // true
int loc = s.indexOf("ll");    // 2
```

Strings Are Immutable

```
String s = "hello";
s = "goodbye";
```

This does **NOT** change the string!

It creates a

new string and stores it in s.
Replacing is not the same as changing.
Everything in the old string is lost.

Converting Between String and char Array

String to char array:

```
String s = "mark";
char[] ca = s.toCharArray();
```

char array to String:

```
char[] ca = {'m', 'a', 'r', 'k'};
String s = new String(ca);
```

Why? char arrays are mutable, Strings are not!

Part C - final Variables

What Are final Variables?

Sometimes we don't want a variable's value to change.

```
final int FOO = 5;
```

- Always named in ALL_CAPS
- Can be assigned **once** and never changed
- Trying to change it won't compile

final Variables Cannot Be Changed

```
final int FOO = 5;
FOO = 6;  // won't compile!

final int BAR;
BAR = 5;   // OK - first assignment
BAR = 6;   // won't compile!
```

final Variables Outside main

```
public class MyClass {
static final int SIZE = 10;

public static void main(String[] args) {
int[] a = new int[SIZE];
// ...
}

public static void someMethod() {
System.out.println(SIZE);  // can use SIZE here too
}
}
```

Review - What You Should Know

char:

- Stores integers, prints as characters
- Convert int to char for Friday's lab

Strings:

- Immutable
- Have useful methods (.equals, .charAt, .contains, etc.)
- Can convert to/from char arrays

final:

- Variables that can't be changed
- Named in ALL_CAPS
- Use static final for class-level constants

Now Let's Code!

We'll practice:

- char and ASCII conversion (for Friday's lab)
- String methods
- final variables