CS 149
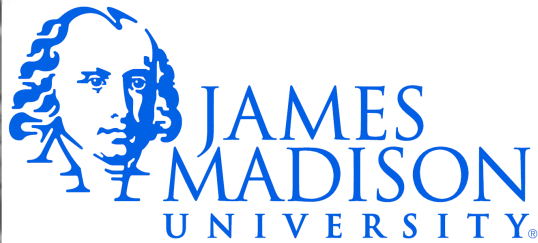
Professor: Alvin Chao

CS149 – Strings
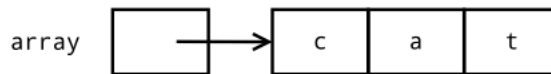
The primitive type **char** is used to store a single character, which can be a letter, a number, or a symbol. In contrast, the reference type String encapsulates an array of characters.
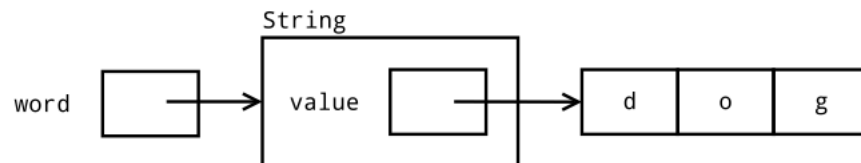
```
char letter;
letter = 'A';

char[] array;
array = new char[]
        {'c', 'a', 't'};

String word;
word = "dog";
```
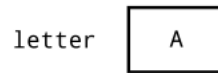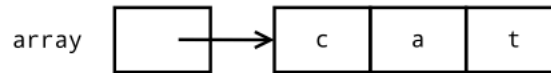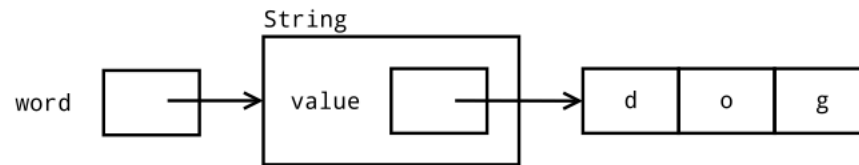
```
char letter;                    letter  [ A ]
letter = 'A';

char[] array;
array = new char[]              array   [  ]→[ c | a | t ]
        {'c', 'a', 't'};

                                        String
String word;                            [ value [  ]→[ d | o | g ] ]
word = "dog";                   word    [  ]→
```
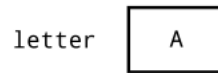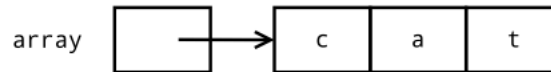
1. How is the syntax of character literals and string literals different?

2. What is the index of 'd' in the string above? What is the index of 'g'? In general, what is the index of the last character of a string?

3. Based on the diagram, what does it mean for a class to encapsulate data? How do you access data inside of a class?

```
char letter;                letter  [ A ]
letter = 'A';

char[] array;               array  [ ]→[ c | a | t ]
array = new char[]
        {'c', 'a', 't'};
                                    String
String word;                       [ value [ ]→[ d | o | g ] ]
word = "dog";               word  [ ]→
```

4. Why can you use the String class in Java programs without having to import it first?

5. What is the value of a char variable? What is the value of an array variable? What is the value of a String variable?

- Draw a memory diagram for the given code. Each variable should be a name next to a box containing its value.

```
String str;
str = "Hi!";

char let;
let = 'X';

int num;
num = -1;

double foo;
foo = num;

String hmm;
hmm = str;
```

7. Recall that the == operator compares the value of two variables. What does it mean for two char variables to be ==? What does it mean for two String variables to be ==?

8. How could you determine whether two character arrays have the same contents? In other words, how does the Arrays.equals method work internally?

Method	Returns	Description
charAt(int)	char	Returns the char value at the specified index of this string.
indexOf(String)	int	Returns the index within this string of the first occurrence of the specified substring.
length()	int	Returns the length of this string.
substring(int, int)	String	Returns a new string that is a substring of this string (from beginIndex to endIndex - 1).
toUpperCase()	String	Returns a copy of this string with all the characters converted to upper case.

9. If str contains the string "hello world", then what is the return value of the following
method calls?
a) str.charAt(6)
b) str.indexOf("ll")
c) str.length()
d) str.substring(4, 7)
e) str.toUpperCase()

Model 2 String Methods

Method	Returns	Description
charAt(int)	char	Returns the char value at the specified index of this string.
indexOf(String)	int	Returns the index within this string of the first occurrence of the specified substring.
length()	int	Returns the length of this string.
substring(int, int)	String	Returns a new string that is a substring of this string (from beginIndex to endIndex - 1).
toUpperCase()	String	Returns a copy of this string with all the characters converted to upper case.

10. Explain what precedes the . (dot) operator in the expressions above. What does it have to do with the keyword this in the model?

11. How many arguments does each method call in #9 have? (Hint: None of them have zero.)

a)

b)

c)

d)

e)

12. To compare strings, you must use either the String.equals or String.compareTo method.
Predict the output of the following code.

```java
String name1 = "Mark";
String name2 = "Ma" + "rk";
String name3 = "Mary";

// compare name1 and name2
if (name1 == name2) {
    System.out.println("name1 and name2 are identical");
} else {
    System.out.println("name1 and name2 are NOT identical");
}


// compare "Mark" and "Mark"
if (name1.equals(name2)) {
    System.out.println("name1 and name2 are equal");
} else {
    System.out.println("name1 and name2 are NOT equal");
}


// compare "Mark" and "Mary"
if (name1.equals(name3)) {
    System.out.println("name1 and name3 are equal");
} else {
    System.out.println("name1 and name3 are NOT equal");
}
```

Program A	Program B

`String greeting = "hello world";`
`greeting.toUpperCase();`
`System.out.println(greeting);`
	`Scanner in = new Scanner(System.in);`
	`String line = in.nextLine();`
	`char letter = line.charAt(1);`
	`System.out.println(letter);`

16. Write the output of each program in the space under the table above. What is the logic error you see when you run Program A?

17. In Program A, what is returned by the string method? What happens to the return value?

18. Describe two different ways you can fix the logic error in #17.

19. In general, what happens to this string when calling the methods in Model 2?

20. In what cases will Program B crash? What is the error message displayed?

21. Describe two different ways you can fix the run-time error in #20.

- **Acknowledgements**

Parts of this activity are based on materials developed by Chris Mayfield and Nathan Sprague.

</end>