# Lesson 13….ASCII and More on *char*

**Things you can't do:**

Character type *char* and *String* types can't be stored into each other. The following lines of code are **illegal**:

```
char ch = aString;  //where aString is a String…..illegal
char ch = "A";  //illegal

String x = xChar;  //where xChar is a char……..illegal
String x = 'X';  //illegal
```

**Surprisingly legal:**

Strangely enough the following **is** legal:

```
int x =1;
char ch = 'A';  //ASCII code for 'A' is 65… (more on ASCII below)
int y = x + ch;  //This is legal!
System.out.println(y);  //66

int z = ch;  //This is legal!
```

**Illegal!**

Storing an *int* type into a *char* is illegal.

```
char ch = j;  //Illegal…assuming j is an int
```

Why is this illegal? It's because *char* can take on Unicode values from 0 – 65536 (two bytes)while *int* types can go over 2 billion. The compiler justly complains about "possible loss of precision" and refuses to do it. Use casting as a way around this.

```
char ch = (char)j;   //Legal…assuming j is an int and less than 65,536
```

**ASCII (pronounced "ask-key") codes:**

Why does the code in middle section above work? It's because characters are just numbers. For example, capital A is stored as a 65. That's why we got 66 above. All characters (letters, numbers, symbols, etc) are stored as numbers. Some ASCII codes that you **should know** are:

| Character | ASCII | Character | ASCII | Character | ASCII |
|---|---|---|---|---|---|
| 0 | 48 | A | 65 | a | 97 |
| 1 | 49 | B | 66 | b | 98 |
| 2 | 50 | C | 67 | c | 99 |
| . . . | . . . | . . . | . . . | . . . | . . . |
| 8 | 56 | Y | 89 | y | 121 |
| 9 | 57 | Z | 90 | z | 122 |

For more on ASCII codes, see Appendix D.

**Conversion between *String*s and characters:**

Let's look back at the top section of this page. What do you do if you absolutely have to convert a *String* into a character or vice versa?

a. **Conversion of a *String* into a character**

```
String s = "W";
char a = s.charAt(0);  //a now equals 'W'
```

b. **Conversion of a character into a *String***

```
char a = 'X';
String s = "" + a;  //concatenation of a string and a character is permit-
                    //ed. The result is a String. The trick is to make the
                    //String we are concatenating an empty String ("").
```

**Conversion from capital to small:**

A way to convert capital-letter characters into small-letter characters is to add 32. Look in the chart above…capital A is 65……small a is 97…….a difference of 32.

```
char bigLetter = 'H';
char smallLetter = (char)(bigLetter + 32); //(bigLetter + 32) is an int that must be
                                           //cast…see # 3 on previous page.
System.out.println(smallLetter);  //h
```

**What are you? (just ask)**

We can ask the following questions of a character (answers are always *true* or *false*),

c. "are you a digit?"

```
char ch = 'a';
System.out.println( Character.isDigit(ch) );  //false

char ch = '3';
System.out.println( Character.isDigit(ch) );  //true
```

d. "are you a letter?"

```
char ch = 'a';
System.out.println( Character.isLetter(ch) );  //true

char ch = '3';
System.out.println(Character.isLetter(ch) );  //false
```

e. "are you a letter or a digit?"

```
char ch = 'a';
System.out.println( Character.isLetterOrDigit(ch) );  //true

char ch = '3';
System.out.println( Character.isLetterOrDigit(ch) );  //true
```

f. "are you whitespace?"….(new line character, space and tabs are whitespace)

```
char ch = '  ';
System.out.println( Character.isWhitespace(ch) );  //true

char ch = 'p';
System.out.println( Character.isWhitespace(ch) );  //false
```

g. "are you lowercase?"

```
char ch = 'a';
System.out.println( Character.isLowerCase(ch) );  //true

char ch = 'A';
System.out.println(Character.isLowerCase(ch) );  //false
```

h. "are you uppercase?"

```
char ch = 'a';
System.out.println( Character.isUpperCase(ch) );  //false

char ch = 'A';
System.out.println( Character.isUpperCase(ch) );  //true
```

**Conversion to upper case:**

We can convert a character to upper case as follows:

```
char ch = 'd';
char nn = Character.toUpperCase(ch);
System.out.println(nn);  //D
```

**Conversion to lower case:**

We can convert a character to lower case as follows:

```
char ch = 'F';
char nn = Character.toLowerCase(ch);
System.out.println(nn);  //f
```

# Exercise on Lesson 13

1. What is the ASCII code for ‘A’?

2. What is the ASCII code for ‘Z’?

3. What is the ASCII code for ‘a’?

4. What is the ASCII code for ‘z’?

5. How many letters are in the English alphabet?

6. What is the ASCII code for the character ‘0’ (this is the number 0 and not the letter O)?

7. What is the ASCII code for the character ‘9’?

8. What does the following code do?

```
char c;
for (int j = 97; j <= 122; j++)  {
        c = (char)(j –32);
        System.out.print(c);
}
```

9. What does the following code do?

```
String s = “Alfred E. Neuman”;
char ch;
for (int x = 0; x < s.length( ); x++)  {
        ch = s.charAt(x);
        if ( (ch <= 90) && (ch>=65) )
                ch = (char)(ch + 32);
        System.out.print(ch);
}
```

10. Write code that will convert *char a* into a *String*.

11. Write code that will convert *String p* into a character. (*p* consists of just one letter.)

12. Is this legal?

```
char ch = ‘V’;
String sd = ch;
```

13. Is this legal?

```
char ch = 'V';
char x = (char)(ch + 56);
```

14. Is this legal?

```
char aa = "X";
```

15. 
```
char k = 'B';
System.out.println(k + 3);  //What's printed?
```

16. 
```
char k = 'B';
System.out.println( (char)(k + 3) );  //What's printed?
```

17. Write code that will insure that an uppercase version of *char boy* is stored in *char cv*.

18. Write code that will insure that a lowercase version of *char boy* is stored in *char cv*.

19. If you have a character called *bv*, what could you do to determine if it's a digit?

20. If you have a character called *bv*, what could you do to determine if it's a letter?

21. If you have a character called *bv*, what could you do to determine if it's an uppercase character?

22. If you have a character called *bv*, what could you do to determine if it's either a letter or a digit?

23. If you have a character called *bv*, what could you do to determine if it's a lowercase character?

24. Describe what the following code does.

```
for(int j = 0; j <= 127; j++)
{
        char ch = (char)j;
        if  (Character.isWhitespace(ch) )
                System.out.println(j);
}
```