**What Is a Naming Convention?**

A naming convention is a rule to follow as you decide what to name your identifiers (e.g. class, package, variable, method, etc..).

**Why Use Naming Conventions?**

Different Java programmers can have different styles and approaches to the way they program. By using standard Java naming conventions they make their code easier to read for themselves and for other programmers. Readability of Java code is important because it means less time is spent trying to figure out what the code does, leaving more time to fix or modify it.

To illustrate the point it's worth mentioning that most software companies will have a document that outlines the naming conventions they want their programmers to follow. A new programmer who becomes familiar with those rules will be able to understand code written by a programmer who might have left the company many years before hand.

**Picking a Name for Your Identifier**

When choosing a name for an identifier make sure it's meaningful. For instance, if your program deals with customer accounts then choose names that make sense to dealing with customers and their accounts (e.g., customerName, accountDetails). Don't worry about the length of the name. A longer name that sums up the identifier perfectly is preferable to a shorter name that might be quick to type but ambiguous.

**A Few Words About Cases**

Using the right letter case is the key to following a naming convention:

* **Lowercase** is where all the letters in a word are written without any capitalization (e.g., while, if, mypackage).
* **Uppercase** is where all the letters in a word are written in capitals. When there are more than two words in the name use underscores to separate them (e.g., MAX\_HOURS, FIRST\_DAY\_OF\_WEEK).
* **CamelCase** (also known as Upper CamelCase) is where each new word begins with a capital letter (e.g., CamelCase, CustomerAccount, PlayingCard).
* **Mixed case** (also known as Lower CamelCase) is the same as CamelCase except the first letter of the name is in lowercase (e.g., hasChildren, customerFirstName, customerLastName).

**Standard Java Naming Conventions**

The below list outlines the standard Java naming conventions for each identifier type:

* **Projects/Packages:** Names should be in lowercase. With small projects that only have a few packages it's okay to just give them simple (but meaningful!) names:
* package pokeranalyzer

package mycalculator

In software companies and large projects where the packages might be imported into other classes, the names will normally be subdivided. Typically this will start with the company domain before being split into layers or features:

package com.mycompany.utilities

package org.bobscompany.application.userinterface

* **Classes:** Names should be in CamelCase. Try to use nouns because a class is normally representing something in the real world:
* class Customer

class Account

* **Interfaces:** Names should be in CamelCase. They tend to have a name that describes an operation that a class can do:
* interface Comparable

interface Enumerable

Note that some programmers like to distinguish interfaces by beginning the name with an "I":

interface IComparable

interface IEnumerable

* **Methods:** Names should be in mixed case. Use verbs to describe what the method does:
* void calculateTax()

string getSurname()

* **Variables:** Names should be in mixed case. The names should represent what the value of the variable represents:
* string firstName

int orderNumber

Only use very short names when the variables are short lived, such as in for loops:

for (int i=0; i<20;i++)

{

//i only lives in here

}

* **Constants:** Names should be in uppercase.
* static final int DEFAULT\_WIDTH

static final int MAX\_HEIGHT