## Class

**ClassInfo**
> Creates a structure to hold info describing a class, and reads the info from interactive popups or from a file. A single thingInfo holds the class name and description. Array lists of thingInfo hold the names, types, and descriptions of the fields, the constructors, and the methods.

## Fields

**classInfo**
> *type: thingInfo*
> Class name and description. Does not use the type field of thingInfo.

**fieldInfo**
> *type: ArrayList<thingInfo>*
> Array list of field names, types, and descriptions

**constructorInfo**
> *type: ArrayList<thingInfo>*
> Array list of constructor names and descriptions. Does not use the type field of thingInfo.

**methodInfo**
> *type: ArrayList<thingInfo>*
> Array list of method names, return types, and descriptions

**baseFilename**
> *type: String*
> Base file name constructed from the class name

**cssFilename**
> *type: String*
> File name for the style sheet

**baseDirname**
> *type: String*
> Name of directory holding the input file

**htmlFilename**
> *type: String*
> File name for the HTML file

**latexFilename**
> *type: String*
> File name for the LaTex file

**needsCSSwritten**
> *type: boolean*
> True unless an existing css file is found

**swapit**
> *type: boolean*
> Indicates order of the type and name information for fields and methods

## Constructors

**ClassInfo(String fname,String dirname,boolean swap)**
> Reads input from either a file or from popup dialogs

**ClassInfo(ClassInfo orig)**
> Copy constructor, but only copying strings so that they might be filtered for latex or html reserved characters

## Methods

**makeFilenames()** *Returns void*
    Creates the filenames and directories and determines whether a css file is present

**resetVals(ClassInfo orig)** *Returns void*
    Clones the strings from orig. Used to reset filtered strings to the original values

**SetParamVals(String msg,int maxit)** *Returns ArrayList<String>*
    Use a dialog to get a number of strings containing information

**breakItUp(String msg)** *Returns ArrayList<String>*
    Break up a string into a number of substrings at each forward slash delimiter