

## Java Quick Reference

This table contains accessible methods from the Java library that may be included on the AP Computer Science A Exam.

| Class Constructors and Methods                               | Explanation  |
|--|--|
| <b>String Class</b>  |  |
| <code>String(String str)</code>                              | Constructs a new <code>String</code> object that represents the same sequence of characters as <code>str</code>  |
| <code>int length()</code>                                    | Returns the number of characters in a <code>String</code> object   |
| <code>String substring(int from, int to)</code>              | Returns the substring beginning at index <code>from</code> and ending at index <code>to - 1</code>   |
| <code>String substring(int from)</code>                      | Returns <code>substring(from, length())</code>   |
| <code>int indexOf(String str)</code>                         | Returns the index of the first occurrence of <code>str</code> ; returns <code>-1</code> if not found   |
| <code>boolean equals(Object other)</code>                    | Returns <code>true</code> if <code>this</code> corresponds to the same sequence of characters as <code>other</code> ; returns <code>false</code> otherwise   |
| <code>int compareTo(String other)</code>                     | Returns a value <code>&lt; 0</code> if <code>this</code> is less than <code>other</code> ; returns zero if <code>this</code> is equal to <code>other</code> ; returns a value <code>&gt; 0</code> if <code>this</code> is greater than <code>other</code> . Strings are ordered based upon the alphabet. |
| <code>String[] split(String del)</code>                      | Returns a <code>String</code> array where each element is a substring of <code>this String</code> , which has been split around matches of the given expression <code>del</code>   |
| <b>Integer Class</b>   |  |
| <code>Integer.MIN_VALUE</code>                               | The minimum value represented by an <code>int</code> or <code>Integer</code>   |
| <code>Integer.MAX_VALUE</code>                               | The maximum value represented by an <code>int</code> or <code>Integer</code>   |
| <code>static int parseInt(String s)</code>                   | Returns the <code>String</code> argument as an <code>int</code>  |
| <b>Double Class</b>  |  |
| <code>static double parseDouble(String s)</code>             | Returns the <code>String</code> argument as a <code>double</code>  |
| <b>Math Class</b>  |  |
| <code>static int abs(int x)</code>                           | Returns the absolute value of an <code>int</code> value  |
| <code>static double abs(double x)</code>                     | Returns the absolute value of a <code>double</code> value  |
| <code>static double pow(double base, double exponent)</code> | Returns the value of the first parameter raised to the power of the second parameter   |
| <code>static double sqrt(double x)</code>                    | Returns the nonnegative square root of a <code>double</code> value   |
| <code>static double random()</code>                          | Returns a <code>double</code> value greater than or equal to <code>0.0</code> and less than <code>1.0</code>   |
| <b>ArrayList Class</b>                                       |  |
| <code>int size()</code>                                      | Returns the number of elements in the list   |
| <code>boolean add(E obj)</code>                              | Appends <code>obj</code> to end of list; returns <code>true</code>   |
| <code>void add(int index, E obj)</code>                      | Inserts <code>obj</code> at position <code>index</code> ( <code>0 &lt;= index &lt;= size</code> ), moving elements at position <code>index</code> and higher to the right (adds 1 to their indices) and adds 1 to size   |
| <code>E get(int index)</code>                                | Returns the element at position <code>index</code> in the list   |
| <code>E set(int index, E obj)</code>                         | Replaces the element at position <code>index</code> with <code>obj</code> ; returns the element formerly at position <code>index</code>  |
| <code>E remove(int index)</code>                             | Removes element from position <code>index</code> , moving elements at position <code>index + 1</code> and higher to the left (subtracts 1 from their indices) and subtracts 1 from size; returns the element formerly at position <code>index</code>   |

| File Class                                |  |
|---|--|
| <code>File(String pathname)</code>        | The <code>File</code> constructor that accepts a <code>String</code> pathname  |
| Scanner Class                             |  |
| <code>Scanner(File f)</code>              | The <code>Scanner</code> constructor that accepts a <code>File</code> for reading  |
| <code>int nextInt()</code>                | Returns the next <code>int</code> read from the file or input source if available. If the next <code>int</code> does not exist or is out of range, it will result in an <code>InputMismatchException</code> .                        |
| <code>double nextDouble()</code>          | Returns the next <code>double</code> read from the file or input source. If the next <code>double</code> does not exist, it will result in an <code>InputMismatchException</code> .  |
| <code>boolean nextBoolean()</code>        | Returns the next <code>boolean</code> read from the file or input source. If the next <code>boolean</code> does not exist, it will result in an <code>InputMismatchException</code> .  |
| <code>String nextLine()</code>            | Returns the next line of text as a <code>String</code> read from the file or input source; can return the empty string if called immediately after another <code>Scanner</code> method that is reading from the file or input source |
| <code>String next()</code>                | Returns the next <code>String</code> read from the file or input source  |
| <code>boolean hasNext()</code>            | Returns <code>true</code> if there is a next item to read in the file or input source; <code>false</code> otherwise  |
| <code>void close()</code>                 | Closes this scanner  |
| Object Class                              |  |
| <code>boolean equals(Object other)</code> |  |
| <code>String toString()</code>            |  |