AP® Computer Science A

2026 EXAM REFERENCE INFORMATION

Name: _____

NOTE: You may use any blank space in this booklet for scratch work during the exam. **Proctors** should collect this reference information at the conclusion of the exam.

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

© 2025 College Board

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

© 2025 College Board