

PRINCIPAIS APIs de JAVA4 (lang, util, io)

INTERFACES

public interface Collection = Set

```
public abstract boolean add(Object o)
public abstract boolean addAll(Collection c)
public abstract void clear()
public abstract boolean contains(Object o)
public abstract boolean containsAll(Collection c)
public abstract boolean equals(Object o)
public abstract boolean isEmpty()
public abstract Iterator iterator()
public abstract boolean remove(Object o)
public abstract boolean removeAll(Collection c)
public abstract boolean retainAll(Collection c)
public abstract int size()
public abstract Object[] toArray()
```

public interface List extends Collection

```
public abstract boolean add(Object o)
public abstract boolean add(int index, Object o)
public abstract boolean addAll(Collection c)
public abstract boolean addAll(int index, Collection c)
public abstract void clear()
public abstract boolean contains(Object elem)
public abstract boolean containsAll(Collection c)
public abstract boolean equals(Object o)
public abstract Object get(int index)
public abstract int indexOf(Object o)
public abstract boolean isEmpty()
public abstract Iterator iterator()
public abstract int lastIndexOf(Object o)
public abstract Object remove(int index)
public abstract boolean remove(Object o)
public abstract boolean removeAll(Collection c)
public abstract boolean retainAll(Collection c)
public abstract Object set(int index, Object o)
public abstract int size()
public abstract List subList(int from, int to)
public abstract ListIterator listIterator()
```

public interface Iterator

```
public abstract boolean hasNext()
public abstract Object next()
public abstract void remove()
```

public interface ListIterator extends Iterator

// métodos abstractos adicionais aos de Iterator

```
public abstract boolean hasPrevious()  
public abstract Object previous()  
public abstract int nextIndex()  
public abstract int previousIndex()
```

public interface Enumeration

```
public abstract boolean hasMoreElements()  
public abstract Object nextElement()
```

public interface Comparable

```
public abstract int compareTo(Object o)
```

public interface Comparator

```
public abstract int compare(Object o1, Object o2)  
public abstract boolean equals(Object o)
```

public interface Map

```
public abstract void clear()  
public abstract boolean containsKey(Object key)  
public abstract boolean containsValue(Object value)  
public abstract Set entrySet()  
public abstract boolean equals(Object o)  
public abstract Object get(Object key)  
public abstract boolean isEmpty()  
public abstract Set keySet()  
public abstract Object put(Object key, Object value)  
public abstract void putAll(Map t)  
public abstract Object remove(Object key)  
public abstract int size()  
public abstract Collection values()
```

public interface Map.Entry (usar com entrySet)

```
public abstract boolean equals(Object o)  
public abstract Object getKey()  
public abstract Object getValue()
```

CLASSES

public abstract class java.util.AbstractCollection implements Collection

public String toString()

public abstract class java.util.AbstractMap implements Map

public String toString()

public class java.util.ArrayList extends AbstractList

implements Cloneable, List, Serializable

ArrayList()

ArrayList(int capacidade)

ArrayList(Collection c)

public boolean add(Object o)

public boolean add(int index, Object o)

public boolean addAll(Collection c)

public boolean addAll(int index, Collection c)

public boolean contains(Object elem)

public boolean containsAll(Collection c)

public boolean equals(Object o)

public Object get(int index)

public int indexOf(Object o)

public boolean isEmpty()

public Iterator iterator()

public int lastIndexOf(Object o)

public Object remove(int index)

public boolean remove(Object o)

public boolean removeAll(Collection c)

public boolean retainAll(Collection c)

public Object set(int index, Object o)

public int size()

public List subList(int from, int to)

public ListIterator listIterator()

public Object clone()

public class java.util.Vector extends AbstractList

implements Cloneable, List, Serializable

Vector()

Vector(int dim)

Vector(Collection c)

public int size()

public boolean isEmpty()

public Enumeration elements()

public void addElement(Object obj)

public void insertElementAt(Object elem, int index)

public boolean contains(Object elem)

public int indexOf(Object elem)

public int lastIndexOf(Object elem, int index)

public Object elementAt(int index)

public Object firstElement()

```
public Object lastElement()
public boolean removeElement(Object obj)
public boolean removeElementAt(int index)
public void removeAllElements(Object elem)
public Object clone()
```

// e ainda todos os métodos de List

***public class java.util.LinkedList extends AbstractSequentialList
implements Cloneable, List, Serializable***

```
LinkedList()
LinkedList(Collection c)
public void addFirst(Object o)
public void addLast(Object o)
public Object getFirst()
public Object getLast()
public Object removeFirst()
public Object removeLast()
```

// e ainda todos os métodos de List

***public class java.util.HashMap extends AbstractMap
implements Cloneable, Map, Serializable***

```
HashMap()
HashMap(int dim)
HashMap(Map t)
public void clear()
public int size()
public boolean isEmpty()
public Set entrySet()
public Set keySet()
public Collection values()
public boolean containsValue(Object value)
public boolean containsKey(Object key)
public Object get(Object key)
public Object put(Object key, Object value)
public void putAll(Map t)
public Object remove(Object key)
public Object clone()
```

***public class java.util.HashSet extends AbstractSet
implements Cloneable, Set, Serializable***

```
HashSet()
HashSet(int dim)
HashSet(Collection c)
public boolean add(Object o)
public void clear()
public boolean contains(Object o)
public boolean isEmpty()
public Iterator iterator()
```

```
public boolean remove(Object o)
public int size()
public Object clone()
```

public class java.lang.Integer implements Serializable

```
Integer(String str) throws NumberFormatException
Integer(int value)
public static final int MAX_VALUE;
public static final int MIN_VALUE;
public static Integer getInteger(String s)
public static int parseInt(String s)
    throws NumberFormatException
public static Integer valueOf(String s)
    throws NumberFormatException
public primitivo primitivoValue()
public int compareTo(Integer io)
public int compareTo(Object o) // Comparable
public String toString()
```

public class java.lang.StringBuffer implements Serializable

```
StringBuffer()
StringBuffer(int dim)
StringBuffer(String str)
public StringBuffer append(Object o | primitivo)
public char charAt(int index)
public StringBuffer delete(int start, int end)
public StringBuffer deleteCharAt(int index)
public StringBuffer insert(int offset, Object | primitivo)
public int length()
public StringBuffer replace(int start, int end, String str)
public StringBuffer reverse()
public void setCharAt(int index, char c)
public String substring(int index)
public String substring(int start, int end)
public String toString()
```

public class java.util.StringTokenizer implements Enumeration

```
StringTokenizer(String str, String delim)
public boolean hasMoreTokens()
public String nextToken()
public String nextToken(String delim)
public int countTokens()
```

public class java.io.PrintWriter extends java.io.Writer

```
public PrintWriter(OutputStream out ou Writer out)
public void print ou println(qualquer tipo primitivo ou Object ou char [])
```

```
public void flush()
public void close()
public void write(String|int|char [])
```

public class java.io.BufferedReader extends java.io.Reader

```
BufferedReader(Reader in)
public void close()
public int read()
public String readLine()
```

public class java.util.StringTokenizer implements Enumeration

```
StringTokenizer(String str, String delim)
public boolean hasMoreTokens()
public String nextToken()
public String nextToken(String delim)
public int countTokens()
```

public class java.io.ObjectOutputStream extends OutputStream

```
ObjectOutputStream(OutputStream os)
public void writeObject(Object obj) throws IOException
public void reset()
public void flush() throws IOException
public void close() throws IOException
public void writeInt(int i) throws IOException
public void writeXXX(XXXprimitivo p) throws IOException
```

Obs : O mesmo com *read* para *ObjectInputStream*.

public class java.lang.Class

```
public static Class forName(String classname)
public Object newInstance()
public boolean isInstance(Object obj)
    // equivalente dinâmico ao operador instanceof
public boolean isInterface()
public Class getSuperClass()
public Class[ ] getInterfaces()
```

public class java.lang.Math

```
public static double E; // 2,718281828...
public static double PI; // 3,14159265358
public static int|long|float|double abs(x|x|x|x value)
public static double asin(double d)
public static double acos(double d)
public static double atan(double d)
public static double sin(double d)
public static double cos(double d)
```

```
public static double tan(double d)
public static double log(double d)
public static double pow(double d1, double d2)
public static double sqrt(double d)
public static double random()
public static int|long|float|double max(x|x|x|x value, x|x|x|x
value)
public static int|long|float|double min(x|x|x|x value, x|x|x|x
value)
public static int round(float a)
public static long round(double d)
public static double toDegrees(double angrad)
public static double toRadians(double angdeg)
```
