

## Examen parcial

<pre> if (condición) {     sentencias; }  if (condición) {     sentencias; } else {     sentencias; }  if (condición) {     sentencias; } else if (condición) {     sentencias; } else {     sentencias; } </pre>	<pre> for (inicialización; condición; actualización) {     sentencias; }  while (condición) {     sentencias; }  public static void nombre(parametros) {     sentencias; }  public static type nombre(parametros) {     sentencias;     return expresión; } </pre>
<p><b>Clase String</b></p> <p>contains(<b>str</b>): true si la cadena contiene los caracteres de la otra</p> <p>endsWith(<b>str</b>), startsWith(<b>str</b>): true si la cadena empieza o acaba con los caracteres de la otra</p> <p>equals(<b>str</b>): true si una cadena es igual a la otra</p> <p>equalsIgnoreCase(<b>str</b>): true si una cadena es igual a la otra, sin tener en cuenta mayúsculas</p> <p>indexOf(<b>str</b>): índice en la cadena de caracteres por el que empieza el string dado (-1 si no está)</p> <p>length(): número de caracteres del string</p> <p>substring(<b>i</b>, <b>j</b>): caracteres que van desde el índice i (incluido) hasta j (excluido)</p> <p>substring(<b>i</b>): caracteres que van desde el índice i (incluido) hasta el final</p> <p>toLowerCase(), toUpperCase(): string con todas sus letras en minúsculas o en mayúsculas</p> <p>charAt(<b>i</b>): carácter con índice i</p>	
<p><b>Clase Scanner</b></p> <p>nextInt(), hasNextInt(): lee un dato como int y comprueba que un dato se puede leer como int</p> <p>next(), hasNext(): lee un dato como String y comprueba que es posible leerlo</p> <p>nextDouble(), hasNextDouble(): lee un dato como double y comprueba que es posible leerlo</p> <p>nextLine(), hasNextLine(): lee una línea como String y comprueba que es posible leerla</p>	
<p><b>Clase Math</b></p> <p>Math.abs(<b>valor</b>): valor absoluto</p> <p>Math.min(<b>v1</b>, <b>v2</b>): valor menor de los 2 dados</p> <p>Math.max(<b>v1</b>, <b>v2</b>): valor mayor de los 2 dados</p> <p>Math.round(<b>valor</b>): número entero más cercano</p> <p>Math.pow(<b>b</b>, <b>e</b>): b elevado a e</p>	<p><b>Clase PrintStream</b></p> <pre> println(texto); print(texto); </pre> <p><b>Clase File</b></p> <p>canRead(): true si el fichero existe y se puede leer</p>
<p><b>Clase Random</b></p> <p>nextInt(<b>max</b>): entero aleatorio entre 0 y max - 1</p> <p><b>Objetos</b></p> <p>Type objectName = new Type(parameters);</p>	<p><b>Arrays</b></p> <pre> tipo[] nombre = new tipo[longitud]; tipo[] nombre = {VAL1, VAL2, VAL3, ...}; nombre[indice] = valor; nombre.length // Número de elementos del array </pre>

