1 Syntax

<table>
<thead>
<tr>
<th>format</th>
<th>start</th>
<th>end</th>
<th>inline</th>
<th>output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rnw</td>
<td><code>&lt;&lt;*&gt;&gt;=</code></td>
<td>@</td>
<td>\Sexpr{x}</td>
<td>TeX</td>
</tr>
<tr>
<td>Rmd</td>
<td><code>---(r *)</code></td>
<td><code>---</code></td>
<td><code>r x</code></td>
<td>MD</td>
</tr>
<tr>
<td>Rhtml</td>
<td><code>&lt;!--begin.rcode * end.rcode--&gt;</code></td>
<td><code>&lt;!--rinline x--&gt;</code></td>
<td>HTML</td>
<td></td>
</tr>
<tr>
<td>Rst</td>
<td><code>.. (r *)</code></td>
<td><code>...</code></td>
<td><code>r: </code>x`</td>
<td>reST</td>
</tr>
<tr>
<td>Rtex</td>
<td><code>% begin.rcode * % end.rcode</code></td>
<td><code>\rinline{x}</code></td>
<td>TeX</td>
<td></td>
</tr>
<tr>
<td>Rascl doc</td>
<td><code>// begin.rcode * // end.rcode</code></td>
<td><code>+r x+</code></td>
<td>AsciiDoc</td>
<td></td>
</tr>
<tr>
<td>Rtextile</td>
<td><code>### begin.rcode * ### end.rcode</code></td>
<td><code>@r x@</code></td>
<td>Textile</td>
<td></td>
</tr>
<tr>
<td>brew</td>
<td><code>% %</code></td>
<td><code>&lt;% x %&gt;</code></td>
<td>text</td>
<td></td>
</tr>
</tbody>
</table>

* denotes local chunk options, e.g. `<label, eval=FALSE>=`; x denotes inline R code, e.g. `'r 1+2'` (MD stands for [Markdown](https://daringfireball.net/projects/markdown/)).

2 Minimal Examples

2.1 Sweave (*.Rnw)

```r
\documentclass{article}
\begin{document}

Below is a code chunk.
\<<foo, echo=TRUE>>=
  z = 1+1
  plot(cars)
\""

The value of z is \Sexpr{z}.
\end{document}
```

2.2 R Markdown (*.Rmd)

```
Hi _markdown_!

```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```
```