Pumping Lemma for Regular Languages

If $L$ is a regular language, then there is a number $p$ (called a pumping length for $L$) such that any string $s \in L$ with $|s| \geq p$ can be split into $s = xyz$ so that the following conditions are satisfied:

1. for each $i \geq 0$, $xy^iz \in L$,
2. $|y| > 0$, and
3. $|xy| \leq p$.

Remarks:

- Condition 2 is equivalent to requiring that $y$ be non-empty.
- If $y$ were allowed to be $\epsilon$, then all the strings $xy^iz$ would be equal to the original string $s$ and the result would be trivial.
- Because of condition 2, $p$ must be at least 1.
- If $p$ is a pumping length for $L$, then so is any $p' > p$, since any string satisfying $|s| \geq p'$ must also satisfy $|s| \geq p$ when $p' > p$. This is why we call $p$ a pumping length for $L$ and not the pumping length for $L$.
- Using $i \geq 2$ in condition 1 is called “pumping up” the string $s$.
- Using $i = 0$ in condition 1 is called “pumping down” the string $s$.
- The Pumping Lemma may be satisfied vacuously, if there are no strings longer than a certain length (which can happen only when $L$ is finite). In this case, any $p$ larger than the length of the longest string in $L$ is a pumping length for $L$. 