Properties of Regular Languages
Closure under simple set operations- union, intersection, concatenation, complementation and star-closure.

Closure properties on regular languages are defined as certain operations on regular language which are guaranteed to produce regular language. Closure refers to some operation on a language, resulting in a new language that is of same "type" as originally operated on i.e., regular.

Regular languages are closed under following operations.
In an automata theory, there are different closure properties for regular languages. They are as follows −

- Union
- Intersection
- concatenation
- Kleene closure
- Complement

Union
If L1 and If L2 are two regular languages, their union L1 U L2 will also be regular.

Example
L1 = $\{a^n \mid n > 0\}$ and L2 = $\{b^n \mid n > 0\}$
L3 = L1 U L2 = $\{a^n \cup b^n \mid n > 0\}$ is also regular.

Intersection
If L1 and If L2 are two regular languages, their intersection L1 ∩ L2 will also be regular.

Example
L1= $\{a^m b^n \mid n > 0 \text{ and } m > 0\}$ and
L2= $\{a^m b^n \cup b^n a^m \mid n > 0 \text{ and } m > 0\}$
L3 = L1 ∩ L2 = $\{a^m b^n \mid n > 0 \text{ and } m > 0\}$ are also regular.

Concatenation
If L1 and If L2 are two regular languages, their concatenation L1.L2 will also be regular.

Example
L1 = $\{a^n \mid n > 0\}$ and L2 = $\{b^n \mid n > 0\}$
L3 = L1.L2 = $\{a^m . b^n \mid m > 0 \text{ and } n > 0\}$ is also regular.

Kleene Closure
If L1 is a regular language, its Kleene closure L1* will also be regular.

Example
L1 = (a U b)
L1* = (a U b)*

Complement
If L(G) is a regular language, its complement L'(G) will also be regular. Complement of a language can be found by subtracting strings which are in L(G) from all possible strings.

Example

L(G) = $\{a^n \mid n > 3\}$ L'(G) = $\{a^n \mid n <= 3\}$

Note − Two regular expressions are equivalent, if languages generated by them are the same. For example, (a+b)* and (a+b)* generate the same language. Every string which is generated by (a+b*)* is also generated by (a+b)* and vice versa.*