bool.dev | Unique ID Generation Cheat Sheet

Identifier	Sample value	Uniqueness	Scalability	Performance	Orderability	Size
Database Auto-Increment	1, 2, 3	Simple and ensures uniqueness within a single table	Easy to implement in single-node applications	Efficient for small-scale applications	Sequentially ordered	32-bit (int) 64-bit (bigint)
Twitter Snowflake	13572484591234567	Guaranteed unique within the distributed system	Highly scalable, can generate millions of IDs per second	Fast generation speed	Time-ordered, useful for chronological sorting	64-bit
UUID v4 (Universally Unique Identifier)	550e8400-e29b-41d4-a716-446655440000	Globally unique without the need for a central authority	Can be generated independently on multiple systems without coordination	Fast generation, no need for network communication	No inherent orderability, random distribution	128-bit
UUID v7	01890c8e-bc4d-7b3f-915e-0d4d4e310e67	Globally unique, combining randomness and timestamp	Can be generated independently on multiple systems	Fast generation	Designed to be sortable by incorporating timestamps	128-bit
ULID (Universally Unique Lexicographically Sortable Identifier)	01ARZ3NDEKTSV4RRFFQ69G5FAV	Globally unique, combining randomness and timestamp	Can be generated independently on multiple systems	Fast generation, suitable for high-throughput systems	Lexicographically sortable, useful for chronological ordering	128-bit
KSUID (K-Sortable Unique Identifier)	0ujsszwN8NRY24YaXiTIEEIo7K0	Combines timestamp and randomness, globally unique	Can be generated independently without coordination	Fast generation and verification, suitable for high-throughput systems	K-sortable, suitable for time-based sorting	160-bit
MongoDB ObjectID	507f1f77bcf86cd799439011	Combines timestamp, machine ID, process ID, and counter for uniqueness	Can be generated independently on multiple nodes	Fast generation, used natively in MongoDB	Roughly ordered based on creation time	96-bit
CUID (Collision-resistant Unique Identifier)	cjld2cyuq0006s1rxy8123456	Highly unique with collision resistance even in high concurrency	Suitable for distributed systems with high-traffic	Fast generation, designed for high-concurrency environments	Not naturally ordered, designed primarily for uniqueness	129-bit
NanoID	V1StGXR8_Z5jdHi6B	Secure and highly unique with customizable size and alphabet	Suitable for distributed systems, highly scalable	Fast generation with high security	No inherent orderability	128-bit
Sonyflake	1132088477364927953	Highly unique with low risk of collisions in a single data center	Optimized for single data center deployment, highly scalable	Fast generation with efficient use of 64-bit storage	Time-ordered, useful for chronological sorting	64-bit
FlakeID	4zqG3B2TnMs57S1PvQ	Combines timestamp, machine ID, and randomness for high uniqueness	Suitable for distributed systems, highly scalable	Fast generation, suitable for high-throughput systems	Roughly ordered based on timestamp	128-bit
Base62	1B2M2Y8AsgTpgAmY7PhCfg	Unique and URL-friendly, avoids special characters	Suitable for web applications needing URL-friendly IDs	Fast generation with compact encoding	No inherent orderability	131-bit

Legend

5

Uniqueness - Guaranteed uniqueness across distributed systems or globally, no risk of collisions.
Scalability - Extremely scalable, ideal for high-throughput and highly distributed systems.
Performance - Very high performance, negligible impact on system speed.
Orderability - Excellent orderability, inherently time-ordered or sequentially sorted.
Storage Efficiency - Extremely efficient storage, minimal ID size.

4

Uniqueness - High likelihood of uniqueness with a well-managed system, very low risk of collisions.
Scalability - Highly scalable, suitable for large applications and some distributed systems.
Performance - High performance, minimal impact on system speed.
Orderability - Good orderability, useful for chronological or sequential sorting.
Storage Efficiency - High storage efficiency, compact ID size.

3

Uniqueness - Good likelihood of uniqueness within a limited scope, minimal risk of collisions.
Scalability - Reasonably scalable, suitable for medium-sized applications.
Performance - Good performance, suitable for most applications.
Orderability - Some inherent orderability, useful in specific contexts.
Storage Efficiency - Reasonable storage efficiency, suitable for most applications.

2

Uniqueness - Moderate likelihood of uniqueness, some risk of collisions.
Scalability - Limited scalability, may struggle under high load or in distributed environments.
Performance - Below-average performance, some impact on system speed.
Orderability - Limited orderability, may require additional sorting mechanisms.
Storage Efficiency - Below-average storage efficiency, larger than necessary IDs.

1

Uniqueness - Low likelihood of uniqueness, high risk of collisions.
Scalability - Not scalable, suitable only for small-scale or single-node applications.
Performance - Slow generation, significant performance impact.
Orderability - No inherent orderability, IDs appear random.
Storage Efficiency - Very inefficient storage, large ID size.