What Is Sha-256?

SHA stands for Secure Hash Algorithm and the 256 simply means that it outputs a 256-bit long string.

SHA-256 is simply one form of hash function in the SHA family, which was initially proposed by the National Security Agency in 1993 with SHA-0. After two years, the National Institute of Standards and Technology created the SHA-1 to address the security flaws discovered in its predecessor.

However, flaws in SHA-1 were discovered, leading to the establishment of the SHA-2 family in 2001.It is referred to as a family because it comprises of six hash functions, each with a distinct digest, or hash value, or in layman's words, size, as indicated by their numbers..

So, SHA-224 is 224-bit long, SHA-256 is 256-bit long, and so on.
SHA-512-224 and SHA-512-256 are just truncated versions of the others.
So, how is the SHA-2 more secure compared to its predecessors?

In order to answer that, I will first give a brief overview of how the Secure Hash Algorithm works.

I will focus on SHA-256 primarily because it is the most widely used type in authentication protocols and is actually the one employed by Bitcoin, which I have discussed earlier.
So, how does the Secure Hash Algorithm work?

SHA always accepts a variable length input and turns it to a fixed length output. The message size, or input size limit, and message digest size, or output length, called hash value, are both determined by the kind of SHA.In the instance of SHA-256, it may accept input of any length between 0 and 2 increased to 64 and will always output a 256-bit hash result. A single letter, a word, a phrase, a paragraph, or even an entire book as input will all result in the same hash length of 256 bits.
It converts data into an alphanumeric hash in hexadecimal representation, meaning its hash value is composed of a total of 64 characters that are either letters or numbers.

You may be asking why there are 64 characters.
One byte is equivalent to eight bits, and one character in hex may be represented with a total of four bits. Because the hash value is 256 bits long, 256 divided by 4 bits per character equals 64 characters. Because of the following qualities, the Secure Hash Algorithm is regarded as a secure cryptographic hash function.

1 .

It is deterministic, which means that for the same input, it will always output the same hash or set of 64 alphanumeric characters. This is significant because if the output altered each time, we would have no means of tracking the input data that we hashed.

2 .

It is quick to compute. SHA is not an encryption algorithm.
It is intended to be used as an authentication mechanism rather than an encryption that must be decoded, which brings me to its third attribute, pre-image resistance.

3.

Pre-image resistance means that it is infeasible to determine the original input data from the output hash. I use the word "infeasible" because finding the input is not completely impossible because the hash created for a certain input will always be the same.
However, because the only means to detect the input is by brute force or by attempting every conceivable combination of the input data, this danger is rendered unimportant.

4 .

The avalanche effect.
This implies that even a minor alteration, such as converting a letter from lowercase to uppercase and vice versa, will totally alter the hash. This attribute protects the hash pre-image.

5 .
It is resistance to collision.
Collision occurs when two distinct inputs have the same hash value. As previously stated, the size limit of the input value for the SHA256 algorithm is 2 multiplied by 64, which is indefinitely huge. This means that SHA256 may be used to hash nearly anything, and that everything, regardless of size, will be transformed to a 256-bit string.
So how is it possible for every hash output to be unique?
Well, the answer to that is no, it is not possible. In fact, no hash function is completely collision-free.
However, like the explanation that I provided for the pre-image resistance, it is highly unlikely to find two input values that will yield the same hash value. It will simply take too long, which makes it an insignificant threat.

Going back to my question earlier, how is the SHA2 more secure compared to its predecessors?
This is because SHA0 and SHA1 both produce only 160-bit long hashes.
Having a shorter length, the hash values that they produce have lower number of different combinations, which makes them more susceptible to collision.