SHA256

- Sha 256 is secure hash algorithm generating 256-bit hash value.
- Takes the message input of 512-bit block and then produces 256-bit output.
- There are steps involved to generate the hash value.
    1. Message padding
        - After giving the inputs, it will give the equivalent hexadecimal value and add single 1 bit at the end of the input.
        - Add zeros after that till 448 locations excluding last 64 bits of the 512.
        - The last 64 bits consists the length of the original input.

Input message	added 1-bit	added zeros	length of the input

Total 512 bit

2. 64 words
    - Making 512-bit block as 64 words having each of 32-bits
    - i.e w [0] ………………….. w [63]
      where w [0] =32-bit
    - The initial w [0] to w [15] will have 512-bit messages. The remaining words are filled using the formula
      From 16 - 63
      S0 = (w[i-15] right rotate 7) ^ (w[i-15] right rotate 18) ^ (w[i-15] right shift 3)
      S1 = (w[i-2] right rotate 17) ^ (w[i-2] right rotate 19) ^ (w[i-2] right shift 10)
      W [i] = w [i-16] + S0 + w [i-7] + S1
3. Setting initial hash values/ working variables and round constants.
    - H0 - H7 = The square root of the initial prime numbers where the fractions part of that written in hexadecimal value.
      K0 – K63 = The cube root of the initial prime numbers where the fractions part of that written in hexadecimal value.
4. 64 round Function
    - Run 64 rounds in that perform these operations
      Sigma0, sigma1, ch, Maj, Temp1, Temp2.

First initialise with the 8 working variables

a = H [0] ……………………………… h = H [7]

Sigma0= (a)right rotate 2 ^ (a)right rotate 13 ^ (a)right rotate 22

Sigma1= (e)right rotate 6 ^ (e)right rotate 11 ^ (e)right rotate 25

Ch (e, f, g) = (e & f) ^ ((~e) & g)

Working like a 2:1 mux

Maj (a, b, c) = (a & b) ^ (a & c) ^ (b & c)

This is majority checker

Temp1 = h + Sigma1(e) + ch (e, f, g) + k [t] + w[t]
Temp2 = Sigma0(a) + Maj (a, b, c)

Then update the working variable for the next round
a= Temp1 + Temp2
b = a
c = b
d = c
e = d + Temp1
f = e
g = f
h = g

5. After 64 rounds update the 8 working variables
    - Formula is
      H0 = H0 + a
      H1 = H1 + b
      H2 = H2 + c
      H3 = H3 + d
      H4 = H4 + e
      H5 = H5 + f
      H6 = H6 + g
      H7 = H7 + h

6. 256-bit hash value
    - Concatenating the updated 8 working variable
      Data_out = {H0, H1, H3, H4, H5, H6, H7}

These are the steps