

AES ENCRYPT/DECRYPT:

1.)

The screenshot shows the 'Recipe' application interface for AES encryption. The 'To Hex' section has 'Delimiter' set to 'Space' and 'Bytes per line' set to '0'. The 'AES Encrypt' section has a 'Key' field containing 'mZq4t7w!z%C*F-JaNdRgUjXn2r5u8x/A?D(G+KbPeShVmYp3s6...', an 'IV' field, and 'Mode' set to 'ECB'. The 'Input' field contains the text 'This is Ryan J. Skelly from CS 527!'. The 'Output' field displays the encrypted hex string: 475f1b8552351c505436c48481e986c5a39bc26433876b6da6eba8f0c71371c748d6b376e8c68cdb59eb68cf82eaabec2534c5733e966444b7e604eca89eafb30d2d42a7283a87db9ca07ded6ab22c677b508fde398ea440aa39398906575240aee57c8b65ef7ee7e5e02d611231510c. The interface also shows a 'Recipe' tab, a 'To Hex' section, and an 'AES Encrypt' section.

KEY:

mZq4t7w!z%C*F-JaNdRgUjXn2r5u8x/A?D(G+KbPeShVmYp3s6v9y\$B&E)H@McQfTjWnZr4t7w
Iz%C*F-JaNdRgUkXp2s5v8x/A?D(G+KbP

IN: This is Ryan J. Skelly from CS 527!

OUT:

475f1b8552351c505436c48481e986c5a39bc26433876b6da6eba8f0c71371c748d6b376e8c68c
db59eb68cf82eaabec2534c5733e966444b7e604eca89eafb30d2d42a7283a87db9ca07ded6ab2
2c677b508fde398ea440aa39398906575240aee57c8b65ef7ee7e5e02d611231510c

2.)

The screenshot shows the 'Recipe' application interface for AES decryption. The 'AES Decrypt' section has a 'Key' field containing 'mZq4t7w!z%C*F-JaNdRgUjXn2r5u8x/A?D(G+KbPeShVmYp3s6...', an 'IV' field, and 'Mode' set to 'ECB'. The 'Input' field contains the encrypted hex string: 475f1b8552351c505436c48481e986c5a39bc26433876b6da6eba8f0c71371c748d6b376e8c68cdb59eb68cf82eaabec2534c5733e966444b7e604eca89eafb30d2d42a7283a87db9ca07ded6ab22c677b508fde398ea440aa39398906575240aee57c8b65ef7ee7e5e02d611231510c. The 'Output' field displays the decrypted text: 'This is Ryan J. Skelly from CS 527!'. The interface also shows a 'Recipe' tab, an 'AES Decrypt' section, and an 'Output' section.

KEY:

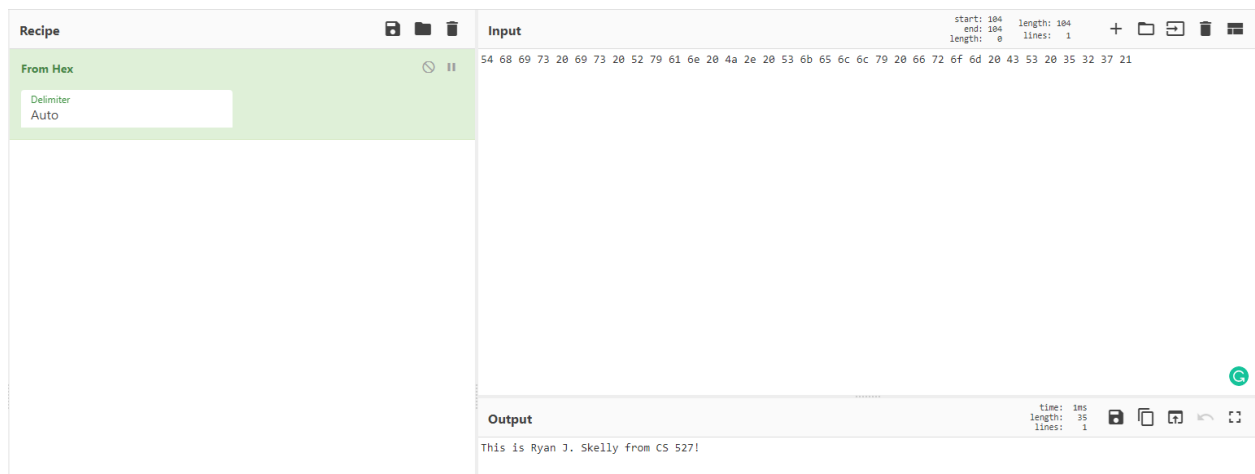
mZq4t7w!z%C*F-JaNdRgUjXn2r5u8x/A?D(G+KbPeShVmYp3s6v9y\$B&E)H@McQfTjWnZr4t7w
!z%C*F-JaNdRgUkXp2s5v8x/A?D(G+KbP

IN:

475f1b8552351c505436c48481e986c5a39bc26433876b6da6eba8f0c71371c748d6b376e8c68c
db59eb68cf82eaabec2534c5733e966444b7e604eca89eafb30d2d42a7283a87db9ca07ded6ab2
2c677b508fde398ea440aa39398906575240aee57c8b65ef7ee7e5e02d611231510c

OUT: 54 68 69 73 20 69 73 20 52 79 61 6e 20 4a 2e 20 53 6b 65 6c 6c 79 20 66 72 6f 6d 20 43
53 20 35 32 37 21

The Decryption method can only output in raw 16bit text, or Hex text, hence why the
Output has hex formatted text.



IN: 54 68 69 73 20 69 73 20 52 79 61 6e 20 4a 2e 20 53 6b 65 6c 6c 79 20 66 72 6f 6d 20 43
53 20 35 32 37 21

OUT: This is Ryan J. Skelly from CS 527!

Since the decryption method would output Hex formatted text, I had to convert it back to
Unicode.

HASH TEXT:

1.)

Recipe

SHA2

Size256

Rounds64

Input

This is Ryan J. Skelly from CS 527!

Output

84fd03306916dd9601bee89cbb7ff4d99f5ae6c26d3deb79045bfd1c7e7eb646

2.)

Recipe

SHA2

Size256

Rounds64

Input

This is Ryan J. Skelly from CS 527?

Output

c5720a444d0442ae954b5e232d2a9d50705e78d63842f48d133cf13eeee47244

The output text changes drastically due to the nature of hash functions scrambling the original text so it “can’t” be decoded.