```
-------------------------------------------------------------------------
-------------------------------------------------------------
PROGRAMMING INFORMATION COMPILATION

RESOURCES:

        Adafruit_PN532.h - https://github.com/adafruit/Adafruit-PN532/
blob/master/Adafruit_PN532.h
        NDEF Formatting  - https://learn.adafruit.com/adafruit-pn532-
rfid-nfc/ndef
        RFID Tag Info    - https://www.adafruit.com/product/359


-------------------------------------------------------------------------
-------------------------------------------------------------









********************************************** GENERAL PROGRAMMING
*************************************************************************

-------------------------------------------------------------------------
-------------------------------------------------------------
Serial Monitor Library and RFID shield library
-------------------------------------------------------------------------
-------------------------------------------------------------

        #include <Wire.h>
        #include <Adafruit_PN532.h>



-------------------------------------------------------------------------
-------------------------------------------------------------
GPIO PIN DECLERATIONS (2 is default but can be changed)
-------------------------------------------------------------------------
-------------------------------------------------------------

        #define PN532_IRQ   (2)
        #define PN532_RESET (3)  // Not connected by default on the
```

NFC Shield

OBJECT DELCERATION: Required to have an object for RFID class use, pass it the pin assignments above
---

```
Adafruit_PN532 nfc(PN532_IRQ, PN532_RESET);
```

********** LIBRARY FUNCTIONS **********
*

FORMATING TAG: New RFID tag must be formatted to a communication proticol NDEF (NFC Data Exchange Format)
Authenticate the block before formating (block 0 only used for formatting!!)


```
uint8_t mifareclassic_FormatNDEF(void);
```


AUTHENTICATING EEPROM: Each block must be authenticated before attemting to write data including formating tag
---

```
        uint8_t mifareclassic_AuthenticateBlock(uint8_t *uid, uint8_t
uidLen,uint32_t blockNumber,uint8_t keyNumber, uint8_t *keyData);



------------------------------------------------------------------------
------------------------------------------------------------
WRITING TO BLOCK:  Valid blocks 4-63 (blocks 0-4 contains system data,
DONT TOUCH!)
                    Verify the block before writing data
------------------------------------------------------------------------
------------------------------------------------------------


        uint8_t mifareclassic_WriteDataBlock(uint8_t blockNumber,
uint8_t *data);



------------------------------------------------------------------------
------------------------------------------------------------
ADVANCED WRITING WITH PREFIXES:  Store website, phone number prefixes
and many more to tag EEPROM
------------------------------------------------------------------------
------------------------------------------------------------


        uint8_t mifareclassic_WriteNDEFURI(uint8_t sectorNumber,
uint8_t uriIdentifier, const char *url);



------------------------------------------------------------------------
------------------------------------------------------------
READING FROM BLOCK:  Can read any block and authentication is not
required before read
------------------------------------------------------------------------
------------------------------------------------------------


        uint8_t mifareclassic_ReadDataBlock(uint8_t blockNumber,
uint8_t *data);


------------------------------------------------------------------------
------------------------------------------------------------
ADDITIONAL FUNCTIONS:
------------------------------------------------------------------------
------------------------------------------------------------
```

```
        bool mifareclassic_IsFirstBlock(uint32_t uiBlock);
        bool mifareclassic_IsTrailerBlock(uint32_t uiBlock);
```

********** TAG INFORMATION **********

TAG DATA: Outlines the relevent tag details

---

1. 13.56MHz RFID/NFC card
2. The card is "blank" (not formatted to a communication proticol)
3. Uses a ISO/IEC 14443 Type A chipset
4. Chips can be written to & store up to 1 KB of data in writable EEPROM divided into "Sectors"
5. Sectors are composed of blocks
6. There is also a permanent 4-byte ID burned into the chip that you can use to identify one tag from another - the ID number cannot be changed.

ACCESS KEYS: Used for Authenticating NDEF Messages in Mifare Sectors of EEPROM (Key Assigned in NDEF formatting)

Clean Tag Default Key (used when no key has been assigned to tag)

```
uint8_t keya[6] = { 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF };
```

Public KEY A of NFC Sectors

BYTE 0	BYTE 1	BYTE 2	BYTE 3	BYTE 4	BYTE 5
0xD3	0xF7	0xD3	0xF7	0xD3	0xF7

```
uint8_t keyA[6] = { 0xD3, 0xF7, 0xD3, 0xF7, 0xD3, 0xF7 };
```