CODE FOR TRANSMITTER:

```
//Include Libraries
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

//create an RF24 object
RF24 radio(9, 8);  // CE, CSN

//address through which two modules communicate.
const byte address[6] = "00001";

void setup()
{
  radio.begin();

  //set the address
  radio.openWritingPipe(address);

  //Set module as transmitter
  radio.stopListening();
}
void loop()
{
  //Send message to receiver
  const char text[] = "Hello World";
  radio.write(&text, sizeof(text));

  delay(1000);
```

```
}
```

CODE FOR RECIEVER :

```
//Include Libraries
#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

//create an RF24 object
RF24 radio(9, 8);  // CE, CSN

//address through which two modules communicate.
const byte address[6] = "00001";

void setup()
{
  while (!Serial);
    Serial.begin(9600);

  radio.begin();

  //set the address
  radio.openReadingPipe(0, address);

  //Set module as receiver
  radio.startListening();
}

void loop()
{
  //Read the data if available in buffer
```

```
  if (radio.available())
  {
    char text[32] = {0};
    radio.read(&text, sizeof(text));
    Serial.println(text);
  }
}
```

NRF CODE :

```
#define DT A0
#define SCK A1
#define sw 2

long sample=0;
float val=0;
long count=0;

unsigned long readCount(void)
{
  unsigned long Count;
  unsigned char i;
  pinMode(DT, OUTPUT);
  digitalWrite(DT,HIGH);
  digitalWrite(SCK,LOW);
  Count=0;
  pinMode(DT, INPUT);
  while(digitalRead(DT));
  for (i=0;i<24;i++)
  {
    digitalWrite(SCK,HIGH);
```

```
    Count=Count<<1;
    digitalWrite(SCK,LOW);
    if(digitalRead(DT))
    Count++;
  }
  digitalWrite(SCK,HIGH);
  Count=Count^0x800000;
  digitalWrite(SCK,LOW);
  return(Count);
}

void setup()
{
  Serial.begin(9600);
  pinMode(SCK, OUTPUT);
  pinMode(sw, INPUT_PULLUP);
}

void loop()
{
  count= readCount();
  int w=(((count-sample)/val)-2*((count-sample)/val));
  Serial.print("weight:");
  Serial.print((int)w);
  Serial.println("g");


  if(digitalRead(sw)==0)
  {
    val=0;
```

```
    sample=0;
    w=0;
    count=0;
    calibrate();
  }
}

void calibrate()
{

  for(int i=0;i<100;i++)
  {
    count=readCount();
    sample+=count;
    Serial.println(count);
  }
  sample/=100;
  Serial.print("Avg:");
  Serial.println(sample);

  count=0;
  while(count<1000)
  {
    count=readCount();
    count=sample-count;
    Serial.println(count);
  }

  delay(2000);
  for(int i=0;i<100;i++)
```

```
  {
    count=readCount();
    val+=sample-count;
    Serial.println(sample-count);
  }
  val=val/100.0;
  val=val/100.0;

}
```