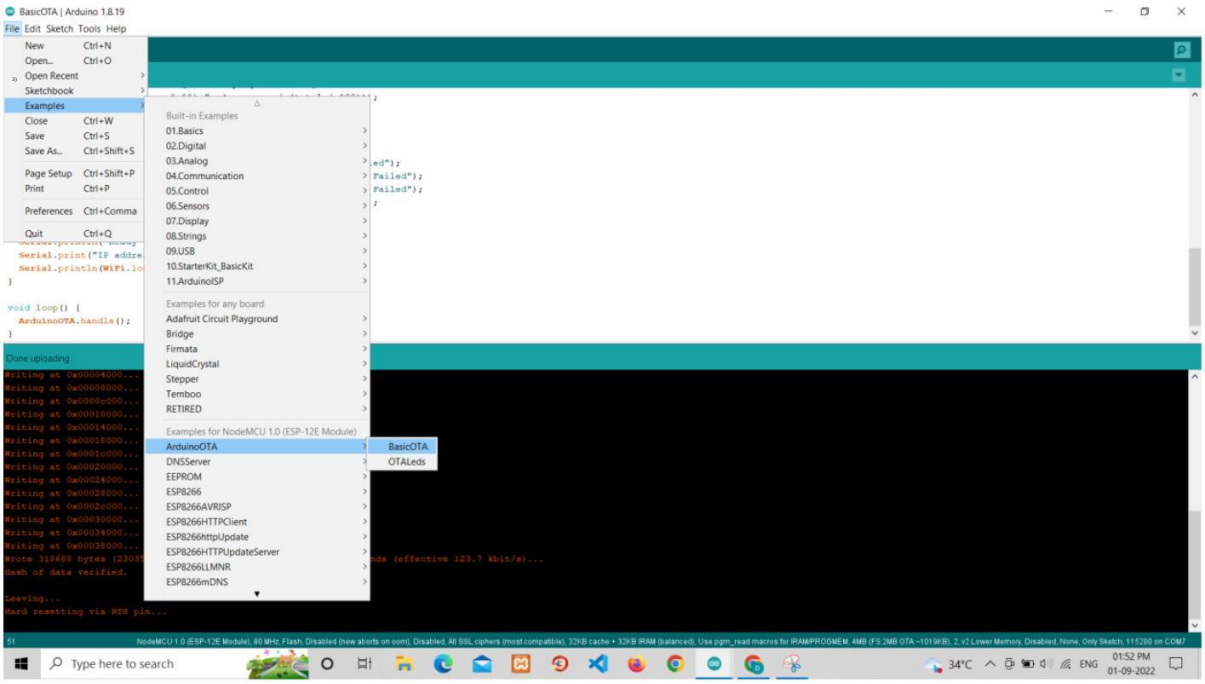
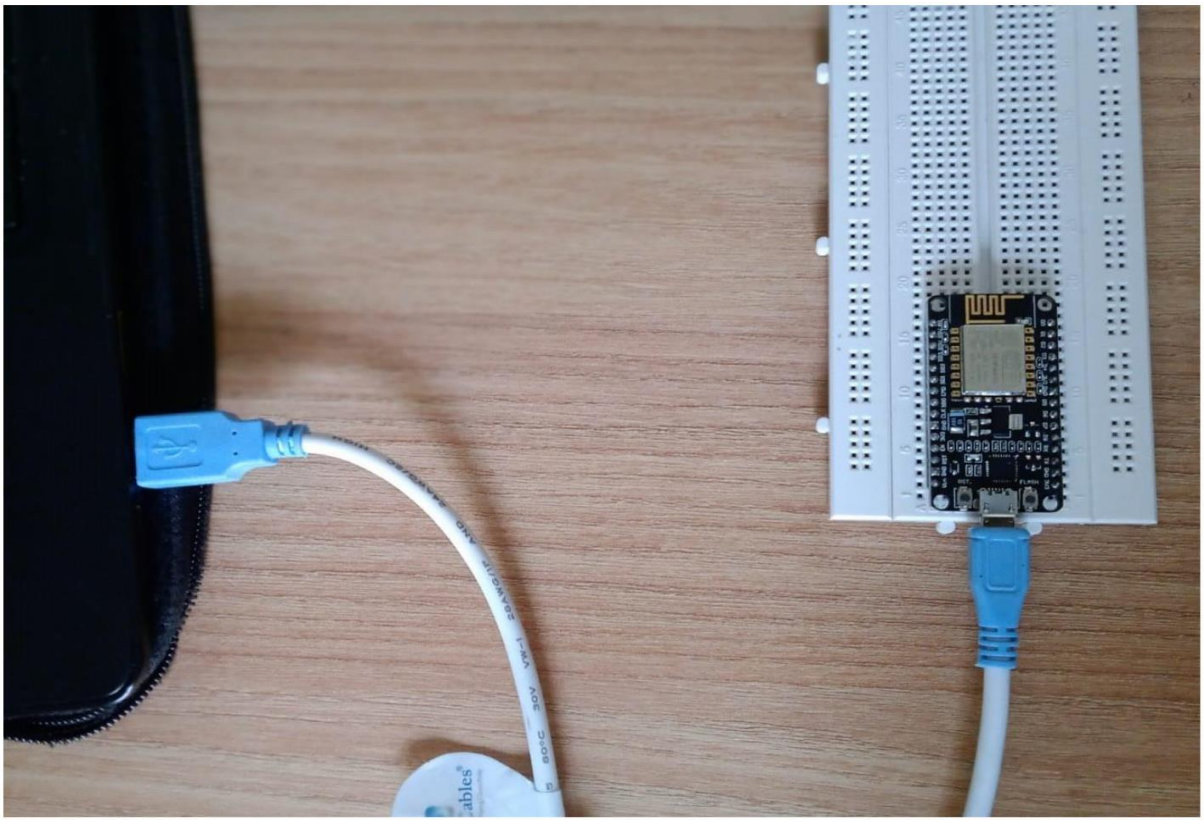
Practical : 11

AIM :- Configure ESP8266 OTA(Over the Air) Updates and test it.

1) You have to upload firmware to your ESP8266 wirelessly.
   So Open the BasicOTA.ino: **File > Examples > Arduino OTA > BasicOTA.ino.**





CODE:

```
#include <ESP8266WiFi.h>
#include <ESP8266mDNS.h>
#include <WiFiUdp.h>
#include <ArduinoOTA.h>

// Replace with your network credentials
const char* ssid = "Mansi";
const char* password = "Mansi@2105";

const int ESP_BUILTIN_LED = 2;

void setup()
{
  Serial.begin(115200);
  Serial.println("Booting");
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  while (WiFi.waitForConnectResult() != WL_CONNECTED) {
    Serial.println("Connection Failed! Rebooting...");
    delay(5000);
    ESP.restart();
  }

  // Port defaults to 8266
  // ArduinoOTA.setPort(8266);

  // Hostname defaults to esp8266-[ChipID]
  // ArduinoOTA.setHostname("myesp8266");
```

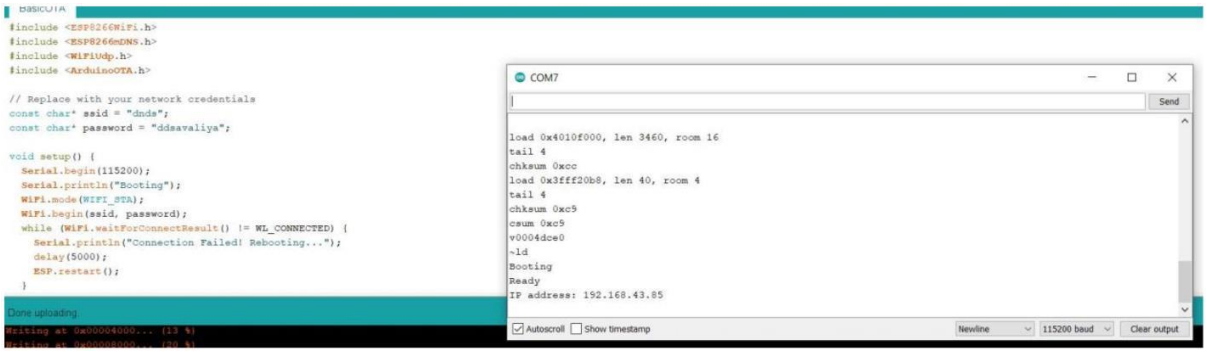
```
  // No authentication by default
  // ArduinoOTA.setPassword((const char *)"123");

  ArduinoOTA.onStart([]()
  {
    Serial.println("Start");
  });
  ArduinoOTA.onEnd([]()
  {
    Serial.println("\nEnd");
  });
  ArduinoOTA.onProgress([](unsigned int progress, unsigned int total) {
    Serial.printf("Progress: %u%%\r", (progress / (total / 100)));
  });
  ArduinoOTA.onError([](ota_error_t error) {
    Serial.printf("Error[%u]: ", error);
    if (error == OTA_AUTH_ERROR) Serial.println("Auth Failed");
    else if (error == OTA_BEGIN_ERROR) Serial.println("Begin Failed");
    else if (error == OTA_CONNECT_ERROR) Serial.println("Connect Failed");
    else if (error == OTA_RECEIVE_ERROR) Serial.println("Receive Failed");
    else if (error == OTA_END_ERROR) Serial.println("End Failed");
  });
  ArduinoOTA.begin();
  Serial.println("Ready");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
  pinMode(ESP_BUILTIN_LED, OUTPUT);
}

void loop()
```

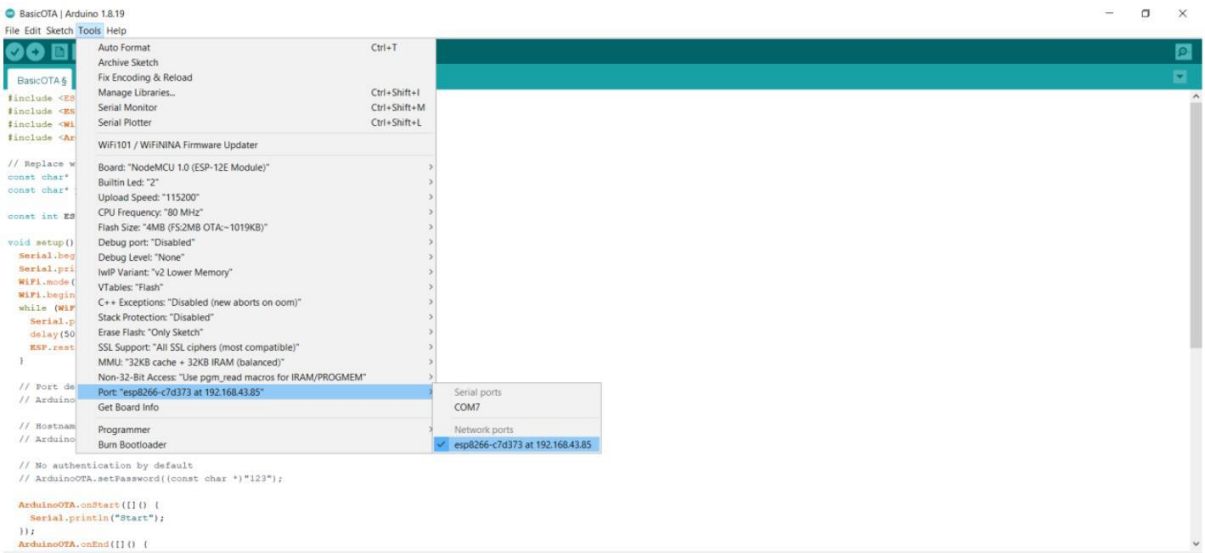
```
{
  ArduinoOTA.handle();
}
```

OUTPUT:



2) ESP8266 is ready to receive OTA firmware updates.

So Go to your Arduino IDE. Open **Tools** tab select the **Port** option and you should see something like this: **esp8266-xxxxxx** at **your_esp_ip_address.**



CODE:

```
#include <ESP8266WiFi.h>
#include <ESP8266mDNS.h>
#include <WiFiUdp.h>
#include <ArduinoOTA.h>

// Replace with your network credentials
const char* ssid = "Mansi";
const char* password = "Mansi@2105";

const int ESP_BUILTIN_LED = 2;

void setup()
{
  Serial.begin(115200);
  Serial.println("Booting");
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  while (WiFi.waitForConnectResult() != WL_CONNECTED) {
    Serial.println("Connection Failed! Rebooting...");
    delay(5000);
    ESP.restart();
  }

  // Port defaults to 8266
  // ArduinoOTA.setPort(8266);

  // Hostname defaults to esp8266-[ChipID]
  // ArduinoOTA.setHostname("myesp8266");
```

```
  // No authentication by default
  // ArduinoOTA.setPassword((const char *)"123");

  ArduinoOTA.onStart([]()
  {
    Serial.println("Start");
  });
  ArduinoOTA.onEnd([]()
  {
    Serial.println("\nEnd");
  });
  ArduinoOTA.onProgress([](unsigned int progress, unsigned int total) {
    Serial.printf("Progress: %u%%\r", (progress / (total / 100)));
  });
  ArduinoOTA.onError([](ota_error_t error) {
    Serial.printf("Error[%u]: ", error);
    if (error == OTA_AUTH_ERROR) Serial.println("Auth Failed");
    else if (error == OTA_BEGIN_ERROR) Serial.println("Begin Failed");
    else if (error == OTA_CONNECT_ERROR) Serial.println("Connect Failed");
    else if (error == OTA_RECEIVE_ERROR) Serial.println("Receive Failed");
    else if (error == OTA_END_ERROR) Serial.println("End Failed");
  });
  ArduinoOTA.begin();
  Serial.println("Ready");
  Serial.print("IP address: ");
  Serial.println(WiFi.localIP());
  pinMode(ESP_BUILTIN_LED, OUTPUT);
}

void loop()
```

```
{
  ArduinoOTA.handle();
  digitalWrite(ESP_BUILTIN_LED, LOW);
  delay(1000);
  digitalWrite(ESP_BUILTIN_LED, HIGH);
  delay(1000);
}
```

OUTPUT: