# Konark IoT Experiments - Extracted Codes

## Experiment 1: MySQL Installation on Raspberry Pi

USE exampledb;  
  
CREATE TABLE students (  
 id INT PRIMARY KEY,  
 name VARCHAR(100),  
 marks INT  
);  
  
INSERT INTO students VALUES (1, 'Alice', 85);  
INSERT INTO students VALUES (2, 'Bob', 90);  
  
SELECT \* FROM students;  
  
DELETE FROM students WHERE name='Bob';

## Experiment 2: Publish Simulated Temperature Data to MQTT Broker (Python)

import random  
import time  
import paho.mqtt.client as mqtt  
  
broker = "broker.hivemq.com"  
topic = "iotlab/temperature"  
  
client = mqtt.Client()  
client.connect(broker, 1883, 60)  
  
while True:  
 temperature = round(random.uniform(20.0, 35.0), 2)  
 client.publish(topic, str(temperature))  
 print(f"Published Temperature: {temperature} °C")  
 time.sleep(2)

## Experiment 3: MQTT Subscriber for Temperature Data (Python)

import paho.mqtt.client as mqtt  
  
def on\_message(client, userdata, msg):  
 print(f"Received Temperature: {msg.payload.decode()} °C")  
  
broker = "broker.hivemq.com"  
topic = "iotlab/temperature"  
  
client = mqtt.Client()  
client.connect(broker, 1883, 60)  
client.subscribe(topic)  
client.on\_message = on\_message  
  
print("Subscribed to topic:", topic)  
client.loop\_forever()

## Experiment 4: TCP Server for Humidity Data (Python)

import socket  
import random  
  
HOST = '0.0.0.0'  
PORT = 65432  
  
server\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  
server\_socket.bind((HOST, PORT))  
server\_socket.listen(1)  
  
print("TCP Server started. Waiting for connections...")  
  
while True:  
 conn, addr = server\_socket.accept()  
 print(f"Connected by {addr}")  
 humidity = round(random.uniform(40.0, 70.0), 2)  
 conn.sendall(f"Humidity: {humidity}%".encode())  
 conn.close()

## Experiment 5: UDP Server for Humidity Data (Python)

import socket  
import random  
  
HOST = "0.0.0.0"  
PORT = 54321  
  
server\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_DGRAM)  
server\_socket.bind((HOST, PORT))  
  
print("UDP Server started...")  
  
while True:  
 data, addr = server\_socket.recvfrom(1024)  
 humidity = round(random.uniform(40.0, 70.0), 2)  
 server\_socket.sendto(f"Humidity: {humidity}%".encode(), addr)

## Experiment 6: DHT11 Sensor Simulation (Python)

import random  
import time  
  
while True:  
 temperature = round(random.uniform(20.0, 35.0), 2)  
 humidity = round(random.uniform(40.0, 70.0), 2)  
 print(f"Temperature: {temperature}°C, Humidity: {humidity}%")  
 time.sleep(2)

## Experiment 7: Motor with Relay and Push Button (Arduino - C++)

int buttonPin = 2;   
int relayPin = 8;   
int buttonState = 0;   
  
void setup() {  
 pinMode(buttonPin, INPUT);  
 pinMode(relayPin, OUTPUT);  
}  
  
void loop() {  
 buttonState = digitalRead(buttonPin);  
 if (buttonState == HIGH) {  
 digitalWrite(relayPin, HIGH); // Motor ON  
 } else {  
 digitalWrite(relayPin, LOW); // Motor OFF  
 }  
}

## Experiment 7: Motor with Relay and Push Button (Raspberry Pi Pico - MicroPython)

from machine import Pin  
import time  
  
button = Pin(14, Pin.IN, Pin.PULL\_DOWN)  
relay = Pin(15, Pin.OUT)  
  
while True:  
 if button.value() == 1:  
 relay.value(1)  
 else:  
 relay.value(0)  
 time.sleep(0.1)

## Experiment 8: Motor Relay Simulation (Arduino - Wokwi)

int buttonPin = 2;  
int relayPin = 7;  
int buttonState = 0;  
  
void setup() {  
 pinMode(buttonPin, INPUT);  
 pinMode(relayPin, OUTPUT);  
 Serial.begin(9600);  
}  
  
void loop() {  
 buttonState = digitalRead(buttonPin);  
 if (buttonState == HIGH) {  
 digitalWrite(relayPin, HIGH);  
 Serial.println("Motor ON");  
 } else {  
 digitalWrite(relayPin, LOW);  
 Serial.println("Motor OFF");  
 }  
 delay(500);  
}

## Experiment 9: OLED Display with DHT11 (Arduino)

#include <Adafruit\_SSD1306.h>  
#include <Adafruit\_GFX.h>  
#include <DHT.h>  
  
#define DHTPIN 2  
#define DHTTYPE DHT11  
  
DHT dht(DHTPIN, DHTTYPE);  
Adafruit\_SSD1306 display(128, 64, &Wire, -1);  
  
void setup() {  
 dht.begin();  
 display.begin(SSD1306\_SWITCHCAPVCC, 0x3C);  
 display.clearDisplay();  
}  
  
void loop() {  
 float temp = dht.readTemperature();  
 float hum = dht.readHumidity();  
   
 display.clearDisplay();  
 display.setTextSize(1);  
 display.setTextColor(WHITE);  
 display.setCursor(0, 0);  
 display.print("Temp: ");  
 display.print(temp);  
 display.println(" C");  
 display.print("Humidity: ");  
 display.print(hum);  
 display.println(" %");  
 display.display();  
 delay(2000);  
}

## Experiment 10: Bluetooth Module with DHT11 (Arduino)

#include <DHT.h>  
  
#define DHTPIN 2  
#define DHTTYPE DHT11  
  
DHT dht(DHTPIN, DHTTYPE);  
  
void setup() {  
 Serial.begin(9600);  
 dht.begin();  
}  
  
void loop() {  
 float temp = dht.readTemperature();  
 float hum = dht.readHumidity();  
   
 Serial.print("Temp: ");  
 Serial.print(temp);  
 Serial.print(" °C, Humidity: ");  
 Serial.print(hum);  
 Serial.println(" %");  
 delay(2000);  
}