

SPRINT_2

Team ID	PNT2022TMID13797
Project Name	Smart Waste Management System for Metropolitan Cities

AIM:

To create a device for the Cloudant database and IoT Watson Cloud platform. Publish data on the IoT Watson Cloud platform, such as bin weight and bin status values(level). Store user information in the Cloudant database, including user name, password, and Gmail. Using a local Node-RED application, to design the workflow for IoT contexts.

COMPONENTS:

S.NO	COMPONENTS	QUANTITY
1	Ultrasonic sensor	2
2	Servo motor	1
3	ESP32 Microcontroller	1
4	HX711 Load Cell	1
5	LCD 16*2 Display	1

DESCRIPTION:

The below code will display the status of both bin level and bin load for a single smart bin. The status of the bin level is shown in the 16*2 LCD display. The data obtained from the sensors is then published to the IoT Watson Cloud Platform. The sensor data can be visually depicted using node-red application.

CODE:

```
#include <ESP32Servo.h>
#include <LiquidCrystal_I2C.h>
#include <HX711.h>
#define DATA_PIN 12
#define CLOCK_PIN 14
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
#define ORG "uuyxja"
#define DEVICE_TYPE "NodeMcu"
```

```

#define DEVICE_ID "12345"
#define TOKEN "23323850"
#define speed 0.034
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
void publishData();
const int trigpin=2;
const int echopin=15;
String command;
String data="";
long duration;
float dist;
LiquidCrystal_I2C LCD = LiquidCrystal_I2C(0x27, 16, 2);
Servo servo;
int trigPin1 = 2;
int echoPin1 = 15;
int trigPin2 = 18;
int echoPin2 = 5;
int duration1;
int distance1;
int duration2;
int distance2;
void setup()
{
  Serial.begin(115200);
  LCD.begin(16,2);
  LCD.init();
  LCD.backlight();
  LCD.clear();
  servo.attach(23);
  Serial.begin(115200);
  pinMode(trigPin1, OUTPUT);
  pinMode(echoPin1, INPUT);
  pinMode(trigPin2, OUTPUT);
  pinMode(echoPin2, INPUT);
  wifiConnect();
  mqttConnect();
}
void loop() {
  publishData();
  delay(500);
  if (!client.loop()) {
    mqttConnect();
  }
}

```

```

}
}
void wifiConnect() {
  Serial.print("Connecting to ");
  Serial.print("Wifi");
  WiFi.begin("Wokwi-GUEST", "", 6);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.print("WiFi connected, IP address: ");
  Serial.println(WiFi.localIP());
}
void mqttConnect() {
  if(!client.connected())
  {
    Serial.print("Reconnecting MQTT client to ");
    Serial.println(server);
    while (!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }
    initManagedDevice();
    Serial.println();
  }
}
void initManagedDevice() {
  if(client.subscribe(topic)) {
    Serial.println(client.subscribe(topic));
    Serial.println("subscribe to cmd OK");
  }
  else {
    Serial.println("subscribe to cmd FAILED"); }
}
void publishData()
{
  digitalWrite(trigPin1, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin1, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin1, LOW);
  duration1 = pulseIn(echoPin1, HIGH);
  distance1= duration1*0.034/2;
  delay(100);
  digitalWrite(trigPin2, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin2, HIGH);
  delayMicroseconds(10);
}

```

```

digitalWrite(trigPin2, LOW);
duration2 = pulseIn(echoPin2, HIGH);
distance2= duration2*0.034/2;
delay(100);

LCD.setCursor(0,1);
LCD.print("Fill Status ");
if(distance2>300 && distance2<=400){
    LCD.setCursor(12,1);
    LCD.print("25% ");
    String payload = "{\"Bin_Level\":\"";payload += "25";payload += "\",\"Bin_Load
\": \"";
    payload += "12.5";
    payload += "\"}";
    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if (client.publish(publishTopic, (char*) payload.c_str())) {
        Serial.println("Publish OK");
    } else {
        Serial.println("Publish FAILED");
    }
}
else if(distance2 > 200 && distance2<= 299){
    LCD.setCursor(12,1);
    LCD.print("50%");
    String payload = "{\"Bin_Level\":\"";payload += "50";payload += "\",\"Bin_Load
\": \"";
    payload += "25";payload += "\"}";
    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if (client.publish(publishTopic, (char*) payload.c_str())) {
        Serial.println("Publish OK");
    } else {
        Serial.println("Publish FAILED");
    }
}
else if(distance2 >50 && distance2<=199){
    LCD.setCursor(12,1);
    LCD.print("75%");
    String payload = "{\"Bin_Level\":\"";payload += "75";payload += "\",\"Bin_Load
\": \"";
    payload += "37.5";payload += "\"}";
    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if (client.publish(publishTopic, (char*) payload.c_str())) {

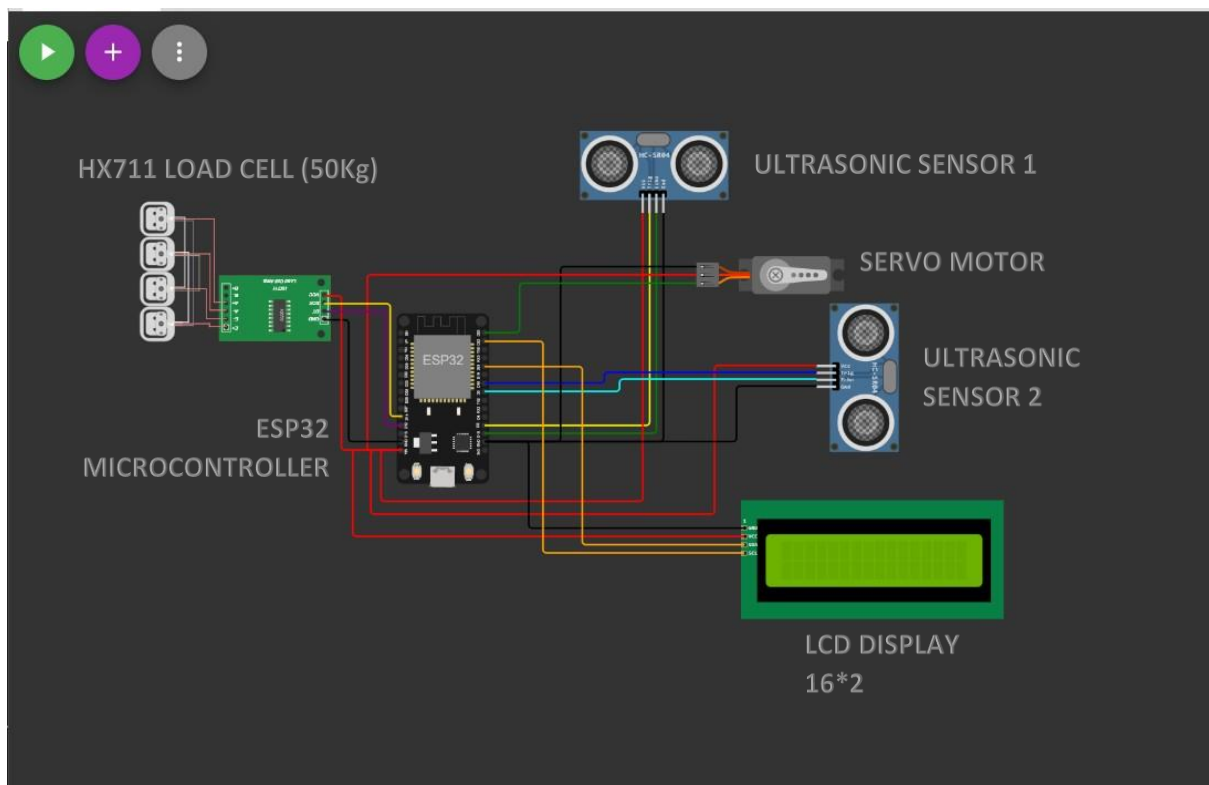
```

```

        Serial.println("Publish OK");
    } else {
        Serial.println("Publish FAILED");
    }
}
else{
    LCD.setCursor(12,1);
    LCD.print("100%");
    String payload = "{\"Bin_Level\":";
    payload += "100";payload += ",\"Weight \":"; payload += "50";payload +=
    "\"}";
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if (client.publish(publishTopic, (char*) payload.c_str())) {
        Serial.println("Publish OK");
    } else {
        Serial.println("Publish FAILED");
    }
}
if(distance1<=50){
    LCD.setCursor(0,0);
    LCD.print("Dustbin is open ");
    servo.write(90);
}
else{
    LCD.setCursor(0,0);
    LCD.print("Dustbin is close ");
    servo.write(0);
}
}
}

```

CIRCUIT DIAGRAM:



OUTPUT:

The screenshot shows a simulation window with the following components and output:

- Simulation Window:** Displays the circuit diagram and the output of the ESP32 microcontroller.
- Output Text:**

```
Connecting to Wifi...WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to dluuhi.messaging.internetofthings.ibmcloud.com
subscribe to cmd OK

Sending payload: {"Bin_Level":50,"Bin_Load ":25}
Publish OK

Sending payload: {"Bin_Level":50,"Bin_Load ":25}
Publish OK
```
- LCD Display:** Shows the output of the simulation: "Dustbin is open" and "Fill Status 25%".

IBM WATSON IoT PLATFORM:

The screenshot shows the IBM Watson IoT Platform dashboard. The top navigation bar includes tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is present with the text 'Search by Device ID'. The main content area displays a table of device information for device ID 12345, which is connected. The table has columns for Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. Below the table, there is a section for 'Recent Events' showing a stream of data points. The events are listed in a table with columns for Event, Value, Format, and Last Received. The events show a consistent pattern of data points with values like 'Bin_Level': 75 and 'Weight': 37.5.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12345	Connected	NodeMcu	Device	Nov 7, 2022 6:31 PM	

Event	Value	Format	Last Received
Data	{"Bin_Level":75,"Weight":37.5}	json	a few seconds ago
Data	{"Bin_Level":75,"Weight":37.5}	json	a few seconds ago
Data	{"Bin_Level":75,"Weight":37.5}	json	a few seconds ago
Data	{"Bin_Level":75,"Weight":37.5}	json	a few seconds ago
Data	{"Bin_Level":75,"Weight":37.5}	json	a few seconds ago

Node-red workflow:

The screenshot shows the Node-RED workflow editor. The main workspace displays a flow with the following components: an 'IBM IoT' node (connected), a 'msg.payload' node, and two 'Bin Level' and 'Weight' nodes. The flow is configured to process incoming data from the IBM IoT node and output the results to the 'Bin Level' and 'Weight' nodes. The left sidebar shows the 'common' and 'function' node categories. The right sidebar shows the 'debug' console, which displays the current flow's output, including the 'Bin_Level' and 'Weight' values.

```
graph LR; IoT[IBM IoT] --> Payload[msg.payload]; Payload --> BinLevel1[Bin Level]; Payload --> Weight[Weight]; BinLevel1 --> BinLevel2[Bin Level]; Weight --> Weight2[Weight];
```

Node-red WEB UI:

Form

Log in

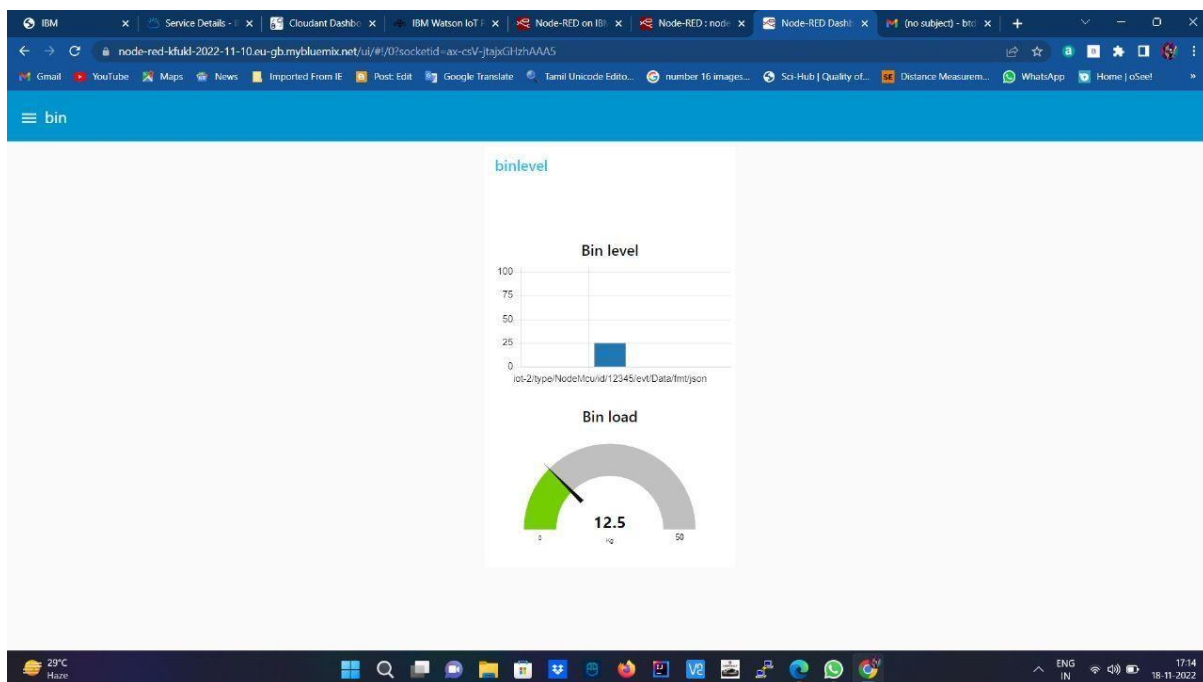
First name *
Loganath

Last name *
M

E-MAIL *
logan50@gmail.com

Password *

SUBMIT CANCEL



CLOUDANT DATABASE:

The screenshot shows the IBM Cloudant Databases dashboard. The browser address bar displays the URL: `3e2f065c-7fcd-41f7-a9a5-41f38dc523e9-bluemix.cloudant.com/dashboard.html#/all_dbs`. The dashboard title is "Databases". Below the title, there is a "Database name" dropdown and a "Create Database" button. The main content area shows a table of databases:

Name	Size	# of Docs	Partitioned	Actions
form	430 bytes	3	No	[Actions]
noderedkfuik20221110	31.9 KB	4	No	[Actions]

At the bottom of the dashboard, it says "Showing 1-2 of 2 databases. Databases per page: 20".

The screenshot shows the IBM Cloudant Databases dashboard with the "form" database selected. The browser address bar displays the URL: `3e2f065c-7fcd-41f7-a9a5-41f38dc523e9-bluemix.cloudant.com/dashboard.html#/database/form/_all_docs`. The dashboard title is "form". Below the title, there is a "Document ID" dropdown and a "Create Document" button. The main content area shows a table of documents:

id	key	value
07126c92f381a24766411b4c32e938e2	07126c92f381a24766411b4c32e938e2	{ "rev": "1-85222aa40384e1308370db62568..." }
31eb66dc121df2469d2c0b3b89337294	31eb66dc121df2469d2c0b3b89337294	{ "rev": "1-85222aa40384e1308370db62568..." }
fe5618060ea1cf64211f58319fbaf29	fe5618060ea1cf64211f58319fbaf29	{ "rev": "1-85222aa40384e1308370db62568..." }

At the bottom of the dashboard, it says "Showing document 1 - 3. Documents per page: 20".

IBM Watson IoT Platform | Service Details - IBM | Overview | IBM Cloud | Cloudant Dashboard | Node-RED: node-red | Node-RED Dashboard

3e2065c-7fed-41f7-a9a5-41f38dc523e9-blumixcloudant.com/dashboard.html#database/form/31eb66dc121df2469d2c0b3b89337294

form > 31eb66dc121df2469d2c0b3b89337294

Save Changes Cancel Upload Attachment Clone Document Delete

```
1-  
2-  
3-  
4-  
5-  
6-  
7-  
8-  
9-  
10-  
11-  
12-  
13-  
14-  
15-  
16-  
17-  
18-  
19-  
20-  
21-  
22-  
23-  
24-  
25-  
26-  
27-  
28-  
29-  
30-  
31-  
32-  
33-  
34-  
35-  
36-  
37-  
38-  
39-  
40-  
41-  
42-  
43-  
44-  
45-  
46-  
47-  
48-  
49-  
50-  
51-  
52-  
53-  
54-  
55-  
56-  
57-  
58-  
59-  
60-  
61-  
62-  
63-  
64-  
65-  
66-  
67-  
68-  
69-  
70-  
71-  
72-  
73-  
74-  
75-  
76-  
77-  
78-  
79-  
80-  
81-  
82-  
83-  
84-  
85-  
86-  
87-  
88-  
89-  
90-  
91-  
92-  
93-  
94-  
95-  
96-  
97-  
98-  
99-  
100-  
101-  
102-  
103-  
104-  
105-  
106-  
107-  
108-  
109-  
110-  
111-  
112-  
113-  
114-  
115-  
116-  
117-  
118-  
119-  
120-  
121-  
122-  
123-  
124-  
125-  
126-  
127-  
128-  
129-  
130-  
131-  
132-  
133-  
134-  
135-  
136-  
137-  
138-  
139-  
140-  
141-  
142-  
143-  
144-  
145-  
146-  
147-  
148-  
149-  
150-  
151-  
152-  
153-  
154-  
155-  
156-  
157-  
158-  
159-  
160-  
161-  
162-  
163-  
164-  
165-  
166-  
167-  
168-  
169-  
170-  
171-  
172-  
173-  
174-  
175-  
176-  
177-  
178-  
179-  
180-  
181-  
182-  
183-  
184-  
185-  
186-  
187-  
188-  
189-  
190-  
191-  
192-  
193-  
194-  
195-  
196-  
197-  
198-  
199-  
200-  
201-  
202-  
203-  
204-  
205-  
206-  
207-  
208-  
209-  
210-  
211-  
212-  
213-  
214-  
215-  
216-  
217-  
218-  
219-  
220-  
221-  
222-  
223-  
224-  
225-  
226-  
227-  
228-  
229-  
230-  
231-  
232-  
233-  
234-  
235-  
236-  
237-  
238-  
239-  
240-  
241-  
242-  
243-  
244-  
245-  
246-  
247-  
248-  
249-  
250-  
251-  
252-  
253-  
254-  
255-  
256-  
257-  
258-  
259-  
260-  
261-  
262-  
263-  
264-  
265-  
266-  
267-  
268-  
269-  
270-  
271-  
272-  
273-  
274-  
275-  
276-  
277-  
278-  
279-  
280-  
281-  
282-  
283-  
284-  
285-  
286-  
287-  
288-  
289-  
290-  
291-  
292-  
293-  
294-  
295-  
296-  
297-  
298-  
299-  
300-  
301-  
302-  
303-  
304-  
305-  
306-  
307-  
308-  
309-  
310-  
311-  
312-  
313-  
314-  
315-  
316-  
317-  
318-  
319-  
320-  
321-  
322-  
323-  
324-  
325-  
326-  
327-  
328-  
329-  
330-  
331-  
332-  
333-  
334-  
335-  
336-  
337-  
338-  
339-  
340-  
341-  
342-  
343-  
344-  
345-  
346-  
347-  
348-  
349-  
350-  
351-  
352-  
353-  
354-  
355-  
356-  
357-  
358-  
359-  
360-  
361-  
362-  
363-  
364-  
365-  
366-  
367-  
368-  
369-  
370-  
371-  
372-  
373-  
374-  
375-  
376-  
377-  
378-  
379-  
380-  
381-  
382-  
383-  
384-  
385-  
386-  
387-  
388-  
389-  
390-  
391-  
392-  
393-  
394-  
395-  
396-  
397-  
398-  
399-  
400-  
401-  
402-  
403-  
404-  
405-  
406-  
407-  
408-  
409-  
410-  
411-  
412-  
413-  
414-  
415-  
416-  
417-  
418-  
419-  
420-  
421-  
422-  
423-  
424-  
425-  
426-  
427-  
428-  
429-  
430-  
431-  
432-  
433-  
434-  
435-  
436-  
437-  
438-  
439-  
440-  
441-  
442-  
443-  
444-  
445-  
446-  
447-  
448-  
449-  
450-  
451-  
452-  
453-  
454-  
455-  
456-  
457-  
458-  
459-  
460-  
461-  
462-  
463-  
464-  
465-  
466-  
467-  
468-  
469-  
470-  
471-  
472-  
473-  
474-  
475-  
476-  
477-  
478-  
479-  
480-  
481-  
482-  
483-  
484-  
485-  
486-  
487-  
488-  
489-  
490-  
491-  
492-  
493-  
494-  
495-  
496-  
497-  
498-  
499-  
500-  
501-  
502-  
503-  
504-  
505-  
506-  
507-  
508-  
509-  
510-  
511-  
512-  
513-  
514-  
515-  
516-  
517-  
518-  
519-  
520-  
521-  
522-  
523-  
524-  
525-  
526-  
527-  
528-  
529-  
530-  
531-  
532-  
533-  
534-  
535-  
536-  
537-  
538-  
539-  
540-  
541-  
542-  
543-  
544-  
545-  
546-  
547-  
548-  
549-  
550-  
551-  
552-  
553-  
554-  
555-  
556-  
557-  
558-  
559-  
560-  
561-  
562-  
563-  
564-  
565-  
566-  
567-  
568-  
569-  
570-  
571-  
572-  
573-  
574-  
575-  
576-  
577-  
578-  
579-  
580-  
581-  
582-  
583-  
584-  
585-  
586-  
587-  
588-  
589-  
590-  
591-  
592-  
593-  
594-  
595-  
596-  
597-  
598-  
599-  
600-  
601-  
602-  
603-  
604-  
605-  
606-  
607-  
608-  
609-  
610-  
611-  
612-  
613-  
614-  
615-  
616-  
617-  
618-  
619-  
620-  
621-  
622-  
623-  
624-  
625-  
626-  
627-  
628-  
629-  
630-  
631-  
632-  
633-  
634-  
635-  
636-  
637-  
638-  
639-  
640-  
641-  
642-  
643-  
644-  
645-  
646-  
647-  
648-  
649-  
650-  
651-  
652-  
653-  
654-  
655-  
656-  
657-  
658-  
659-  
660-  
661-  
662-  
663-  
664-  
665-  
666-  
667-  
668-  
669-  
670-  
671-  
672-  
673-  
674-  
675-  
676-  
677-  
678-  
679-  
680-  
681-  
682-  
683-  
684-  
685-  
686-  
687-  
688-  
689-  
690-  
691-  
692-  
693-  
694-  
695-  
696-  
697-  
698-  
699-  
700-  
701-  
702-  
703-  
704-  
705-  
706-  
707-  
708-  
709-  
710-  
711-  
712-  
713-  
714-  
715-  
716-  
717-  
718-  
719-  
720-  
721-  
722-  
723-  
724-  
725-  
726-  
727-  
728-  
729-  
730-  
731-  
732-  
733-  
734-  
735-  
736-  
737-  
738-  
739-  
740-  
741-  
742-  
743-  
744-  
745-  
746-  
747-  
748-  
749-  
750-  
751-  
752-  
753-  
754-  
755-  
756-  
757-  
758-  
759-  
760-  
761-  
762-  
763-  
764-  
765-  
766-  
767-  
768-  
769-  
770-  
771-  
772-  
773-  
774-  
775-  
776-  
777-  
778-  
779-  
780-  
781-  
782-  
783-  
784-  
785-  
786-  
787-  
788-  
789-  
790-  
791-  
792-  
793-  
794-  
795-  
796-  
797-  
798-  
799-  
800-  
801-  
802-  
803-  
804-  
805-  
806-  
807-  
808-  
809-  
810-  
811-  
812-  
813-  
814-  
815-  
816-  
817-  
818-  
819-  
820-  
821-  
822-  
823-  
824-  
825-  
826-  
827-  
828-  
829-  
830-  
831-  
832-  
833-  
834-  
835-  
836-  
837-  
838-  
839-  
840-  
841-  
842-  
843-  
844-  
845-  
846-  
847-  
848-  
849-  
850-  
851-  
852-  
853-  
854-  
855-  
856-  
857-  
858-  
859-  
860-  
861-  
862-  
863-  
864-  
865-  
866-  
867-  
868-  
869-  
870-  
871-  
872-  
873-  
874-  
875-  
876-  
877-  
878-  
879-  
880-  
881-  
882-  
883-  
884-  
885-  
886-  
887-  
888-  
889-  
890-  
891-  
892-  
893-  
894-  
895-  
896-  
897-  
898-  
899-  
900-  
901-  
902-  
903-  
904-  
905-  
906-  
907-  
908-  
909-  
910-  
911-  
912-  
913-  
914-  
915-  
916-  
917-  
918-  
919-  
920-  
921-  
922-  
923-  
924-  
925-  
926-  
927-  
928-  
929-  
930-  
931-  
932-  
933-  
934-  
935-  
936-  
937-  
938-  
939-  
940-  
941-  
942-  
943-  
944-  
945-  
946-  
947-  
948-  
949-  
950-  
951-  
952-  
953-  
954-  
955-  
956-  
957-  
958-  
959-  
960-  
961-  
962-  
963-  
964-  
965-  
966-  
967-  
968-  
969-  
970-  
971-  
972-  
973-  
974-  
975-  
976-  
977-  
978-  
979-  
980-  
981-  
982-  
983-  
984-  
985-  
986-  
987-  
988-  
989-  
990-  
991-  
992-  
993-  
994-  
995-  
996-  
997-  
998-  
999-  
1000-
```

Log Out

27°C Haze

ENG IN 18-11-2022