欄位名稱	說明
type	裝置類型（tree / bin / water）
imei	Sensor IMEI（唯一識別）
imei_short	IMEI 後 8 碼（前端介面顯示用）
code	裝置代碼（sen_code）
owner	資產編號（lot_no）
lot_id	資產主鍵
name	資產名稱（如：Tree 531）
sub_type	感測器子類型（如：Body/Root/Paper/General Waste/Level）
value_json	原始感測 JSON 數據（含 X/Y/Z、C1 等讀數）
fill	Bin 填充百分比 (0-100)，其餘類型為 NULL
level	Water 水位百分比 (0-100)，其餘類型為 NULL
depth	Water 槽深 (cm)，其餘類型為 NULL
connected	1 為在線，0 為離線
battery_percent	由函數計算之 0-100 電量百分比
status	狀態代碼（urgent, important, warning, low_battery, offline, normal）

status_text	狀態顯示文字（如：Urgent, Low Battery, Full, High 等）
severity_value	嚴重度核心數值：Tree 為傾斜角(°)、Bin 為填充率(%)、Water 為水位率(%)
rpTime	最後回報時間（YYYY-MM-DD HH:MM:SS）
lng / lat	感測器經緯度座標
quality_json	訊號品質 JSON (RSRP/SNR)
extra_json	額外資訊（如 clearTime 清除時間）
is_real	1=正式資料, 0=測試資料
worker_name	負責人姓名（NULL 代表未指派）
worker_lat / worker_lng	負責人之即時位置座標
distance_km	感測器與負責 Worker 間的 Haversine 直線距離（單位：km）

1. 離線優先 **(Offline)**: 若 `connected` 為 0, 優先判定為 **offline**。
2. 異常警示 **(Severity)**: 根據 `severity_value` 判斷是否達到 **urgent**、**important** 或 **warning** 之閾值。
3. 低電量 **(Low Battery)**: 若數值正常但 `battery_percent` 過低, 判定為 **low_battery**。
4. 正常 **(Normal)**: 未觸發上述條件則判定為 **normal**。
5. 各類型嚴重度標準：
    - **Tree (**路樹**)**: 監控傾斜角度（以度為單位）, 數值越大代表倒塌風險越高。
    - **Bin (**垃圾桶**)**: 監控填充百分比（Fill %）, 數值越大代表越接近滿溢。
    - **Water (**水體**)**: 監控水位百分比（Level %）, 數值越大代表水位越高。

Q1: 目前有哪些需要立即處理的？

```sql
SELECT type, name, sub_type, imei, status_text, severity_value, worker_name,
distance_km
FROM v_sensor_status
WHERE status = 'urgent'
ORDER BY type, severity_value DESC;
```

預期數量：Tree 60 + Bin 665 + Water 72 = **797**

Q2: 按輕重緩急安排路線（指定 worker）

```sql
SELECT type, name, sub_type, status, severity_value, lng, lat, distance_km
FROM v_sensor_status
WHERE worker_name = 'Chris'
  AND status NOT IN ('normal')
ORDER BY
  FIELD(status, 'urgent', 'important', 'warning', 'low_battery', 'offline'),
  distance_km ASC;
```

Q3: 目前 Tree 最嚴重的有多少？

```sql
SELECT COUNT(*) AS urgent_tree_count
FROM v_sensor_status
WHERE type = 'tree' AND status = 'urgent';
```

預期：**60**

Q4: 目前垃圾桶有多少快要滿的？

```sql
SELECT COUNT(*) AS full_bin_count
FROM v_sensor_status
WHERE type = 'bin' AND status = 'urgent';
```

預期：**665**

Q5: 目前水位有多少是快要滿的？

```sql
SELECT COUNT(*) AS high_water_count
FROM v_sensor_status
WHERE type = 'water' AND status = 'urgent';
```

預期：**72**

Q6: 各類型 × 狀態統計（全景）

```sql
SELECT type, status, COUNT(*) AS cnt
FROM v_sensor_status
GROUP BY type, status
ORDER BY type, FIELD(status, 'urgent', 'important', 'warning', 'low_battery', 'offline',
'normal');
```

Q7: 各 worker 負責多少異常裝置？

```sql
SELECT worker_name,
    COUNT(*) AS total,
    SUM(status = 'urgent') AS urgent,
    SUM(status = 'important') AS important,
    SUM(status = 'warning') AS warning,
    SUM(status = 'low_battery') AS low_battery,
    SUM(status = 'offline') AS offline
FROM v_sensor_status
WHERE status != 'normal'
GROUP BY worker_name
ORDER BY urgent DESC, important DESC;
```

Q8: 哪個 worker 最忙（異常最多）？

```sql
SELECT worker_name, COUNT(*) AS abnormal_count
FROM v_sensor_status
WHERE status != 'normal'
GROUP BY worker_name
ORDER BY abnormal_count DESC
LIMIT 5;
```

Q9: 離 worker 最遠的 urgent（可能需要調度）

```sql
SELECT type, name, status, severity_value, worker_name, distance_km
FROM v_sensor_status
WHERE status = 'urgent' AND distance_km > 5
ORDER BY distance_km DESC;
```

Q10: 電量快沒的裝置

```sql
SELECT type, name, imei, battery_percent, status, worker_name
FROM v_sensor_status
WHERE battery_percent <= 15
ORDER BY battery_percent ASC;
```

Q11: 斷線裝置清單

```sql
SELECT type, name, imei, worker_name, distance_km
FROM v_sensor_status
WHERE status = 'offline'
ORDER BY type, distance_km ASC;
```

Q12: 某個 worker 今天要跑的完整路線（含座標）

```sql
SELECT type, name, sub_type, status, status_text, severity_value,
```

```
       lng, lat, distance_km
FROM v_sensor_status
WHERE worker_name = 'Chris'
  AND status NOT IN ('normal')
ORDER BY
  FIELD(status, 'urgent', 'important', 'warning', 'low_battery', 'offline'),
  distance_km ASC;
```

Q13: 派誰去最近？（找離某 sensor 最近的 worker）

```sql
SELECT worker_name, distance_km
FROM v_sensor_status
WHERE owner = 'RBIN_001'
LIMIT 1;
```

或反過來：某 urgent sensor 周圍哪些 worker 最近：

```sql
SELECT s.name, s.lng, s.lat,
       w.dw_name,
       ROUND(6371 * ACOS(LEAST(1,
         COS(RADIANS(w.dw_lat)) * COS(RADIANS(s.lat)) * COS(RADIANS(s.lng) -
RADIANS(w.dw_lng))
         + SIN(RADIANS(w.dw_lat)) * SIN(RADIANS(s.lat))
       )), 2) AS dist_km
FROM v_sensor_status s
CROSS JOIN tb_dispatch_worker w
WHERE s.imei = '867637485053274'
  AND w.dw_enabled = 1
ORDER BY dist_km ASC
LIMIT 3;
```

Q14: Dashboard 等價統計（按 owner 分組，取最嚴重狀態）

```sql
SELECT worker_name,
       COUNT(DISTINCT owner) AS total_assets,
       COUNT(DISTINCT CASE WHEN worst != 'normal' THEN owner END) AS
abnormal_assets,
```

```
      COUNT(DISTINCT CASE WHEN worst = 'normal' THEN owner END) AS
normal_assets
FROM (
   SELECT owner, worker_name,
          CASE
             WHEN SUM(status = 'urgent') > 0 THEN 'urgent'
             WHEN SUM(status = 'important') > 0 THEN 'important'
             WHEN SUM(status = 'warning') > 0 THEN 'warning'
             WHEN SUM(status = 'low_battery') > 0 THEN 'low_battery'
             WHEN SUM(status = 'offline') > 0 THEN 'offline'
             ELSE 'normal'
          END AS worst
   FROM v_sensor_status
   GROUP BY owner, worker_name
) t
GROUP BY worker_name
ORDER BY abnormal_assets DESC;
```

Chris 預期：total=30, abnormal=6, normal=24

Q15: 某站所有 sensor 明細（如 TRX1 Station 三個桶）

```sql
SELECT sub_type, fill, status, status_text, battery_percent, rpTime
FROM v_sensor_status
WHERE name = 'TRX1 Station'
ORDER BY sub_type;
```
