📊 Data Format Specification - RainfallFussion

Input Data Format Requirements

File Type & Extensions

- **Accepted:** `.xlsx` (Excel 2007+)
- **NOT Accepted:** `.xls` (old format), `.csv`, `.txt`, `.json`

How to Convert to .xlsx:

- Open file di Excel → Save As → Format: Excel Workbook (.xlsx)
- Google Sheets: Download → Microsoft Excel (.xlsx)

Excel File Structure

Basic Requirements

1. Single Sheet per File

- Aplikasi membaca sheet pertama (default)
- Jika ada multiple sheets, hanya pertama yang diketahui
- Nama sheet tidak penting (bisa "Data", "Sheet1", dll)

2. Header Row (Row 1)

```
EXACT FORMAT (case-sensitive untuk Month names):

A1: Year
B1: Day
C1: Jan
D1: Feb
E1: Mar
F1: Apr
G1: May
H1: Jun
I1: Jul
J1: Aug
K1: Sep
L1: Oct
M1: Nov
N1: Dec
```text

**⚠ PENTING:** Month names harus singkat (Jan-Dec), bukan full names.

#### 3. **Data Rows (Row 2 onwards)**

Setiap row data harus memiliki format:

```text
| Year (numeric) | Day (numeric) | Jan | Feb | ... | Dec |
|    COLUMN A    |   COLUMN B    |  C  |  D  | ... |  N  |
```text

---

## Data Value Specifications

### Year Column (Column A)

- **Format:** Integer (full year, e.g., 2020, 2021, 2022)
- **Range:** Recommended 1900-2100
- **Examples:** ✓ 2020, ✓ 2021 | ✗ 20, ✗ '2020 (text)

### Day Column (Column B)

- **Format:** Integer (day number in year)
- **Range:** 1-365 (normal year) or 1-366 (leap year)
- **Note:** Sequential order recommended
- **Examples:** ✓ 1, ✓ 182, ✓ 365 | ✗ 0, ✗ 366 (non-leap)

### Monthly Data (Columns C-N)

- **Format:** Numeric (integer or decimal)
- **Unit:** Millimeters (mm) - universal rainfall measurement
- **Range:** 0 - 500 (extreme cases up to 1000 for typhoon)
- **Decimal Places:** 1-2 recommended (e.g., 12.5, 0.8)
- **Missing Values:** Leave EMPTY CELL (blank), do NOT use:
  - Text: "N/A", "NA", "missing", "-", "—"
  - Zero for unknown values
  - Error values like "#N/A"

**Valid Examples:**

```text
✓ 12.5    (rainy day)
✓ 0.1     (trace rain)
✓ 250.3   (heavy rain)
✓ [blank] (missing data)
```text

**Invalid Examples:**

```text
✗ "N/A"      (text)
✗ "missing"  (text)
✗ #N/A       (error)
✗ -5         (negative)
✗ 1000       (unrealistic)
```text

````

Complete Example - Valid Data

Example 1: Surakarta Rainfall Station (2020)

```text
Year | Day | Jan   | Feb   | Mar   | Apr   | May   | Jun   | Jul   | Aug   | Sep
| Oct   | Nov   | Dec
2020 | 1   | 0.0   | 5.2   | 12.5  | 15.3  | 22.1  |       | 28.9  | 26.5  | 23.2
| 18.9  | 10.5  | 2.3
2020 | 2   |       | 4.8   | 11.9  | 14.5  | 21.8  | 24.3  | 27.6  | 25.8  | 22.9
| 18.5  | 10.2  | 2.1
2020 | 3   | 1.2   | 4.5   | 11.3  | 13.9  | 21.5  | 23.8  | 27.1  | 25.2  | 22.6
| 18.1  | 9.8   | 1.9
2020 | 4   | 2.1   | 4.2   | 10.8  |       | 21.2  | 23.3  | 26.5  | 24.6  | 22.2
| 17.7  | 9.5   | 1.7
2020 | 5   | 3.5   |       | 10.2  | 12.9  | 20.9  | 22.8  | 25.9  | 24.1  | 21.9
| 17.3  | 9.1   | 1.5
```text

**Karakteristik Data Valid:**

- ✓ Year column: konsisten (2020)
- ✓ Day column: sequential (1,2,3,4,5)
- ✓ Data values: numerikal dengan desimal
- ✓ Missing values: completely blank (tidak ada text/zero)
- ✓ Range: semua dalam 0-30 mm (reasonable untuk daily rainfall)

---

## Data Validation Rules

### Must-Have Requirements

1. **File format:** `.xlsx`
2. **Header row:** Exactly as specified
3. **Data completeness:** Minimal 5 rows data
4. **Column count:** Exactly 14 columns (Year through Dec)
5. **Data type:** Numeric (except Year/Day which are integers)

### Strongly Recommended

1. **Data coverage:** Minimal 1 full tahun (365 days)
2. **Consistent year:** Jika multiple years, organize by blocks atau sort
3. **Missing pattern:** < 30% missing values per stasiun
4. **Outlier handling:** Remove unrealistic values (>500mm daily)
5. **Data quality:** Cross-check dengan neighboring stations

### Validation Checklist

```text
Before Upload:
☐ File extension is .xlsx
````
````

````
☐ Header row matches: Year, Day, Jan-Dec
☐ No empty rows di tengah data
☐ No merged cells
☐ All values are numbers (except missing=blank)
☐ No text values seperti "N/A" atau "missing"
☐ No formula cells (convert to values)
☐ Year column contains 4-digit years
☐ Day column in range 1-366
☐ Monthly values non-negative (<500 usually)
☐ Minimal 5 rows, ideally 365+ rows
```text

---

## Common Data Issues & Solutions

### Issue 1: Mixed Text & Numbers in Data

```text
WRONG:
Year | Day | Jan
2020 | 1   | 12.5
2020 | 2   | N/A         ← TEXT tidak bisa dibaca!
2020 | 3   | 14.3

CORRECT:
Year | Day | Jan
2020 | 1   | 12.5
2020 | 2   |             ← BLANK untuk missing
2020 | 3   | 14.3
```text

**Solution:** Replace semua "N/A", "missing", dll dengan blank cell

---

### Issue 2: Inconsistent Date Format

```text
WRONG:
Year | Day
2020 | 1
2020 | 2/1      ← Format date, bukan day number!
2020 | 3

CORRECT:
Year | Day
2020 | 1
2020 | 2
2020 | 3
```text

**Solution:** Day harus pure numerikal (1-365), bukan format tanggal
````
````

````
---

### Issue 3: Wrong Column Headers

```text
WRONG:
Year | Date | January | February | ...  ← Full month names!

CORRECT:
Year | Day | Jan | Feb | ...
```text

**Solution:** Use exact month abbreviations (Jan-Dec)

---

### Issue 4: Multiple Sheets

```text
WRONG:
[Sheet1: Data tahun 2020]
[Sheet2: Data tahun 2021]    ← Aplikasi hanya baca sheet pertama!
[Sheet3: Summary]

CORRECT:
[Sheet "AllData": Data 2020-2021 combined]
Atau buat file terpisah per tahun
```text

**Solution:** Combine data dalam 1 sheet, atau upload file terpisah

---

### Issue 5: Zero vs Missing Values

```text
WRONG:
Year | Day | Jan
2020 | 1   | 0        ← Bisa berarti "no rain" atau "missing"
2020 | 2   | 0        ← Ambiguous!

CORRECT (Option A - No rain):
Year | Day | Jan
2020 | 1   | 0.0      ← Clear zero/trace rain
2020 | 2   | 0.1      ← Measured tiny amount

CORRECT (Option B - Unknown/Missing):
Year | Day | Jan
2020 | 1   |          ← BLANK untuk unknown
2020 | 2   |
```text

**Solution:** Define clearly: 0 untuk no rain, blank untuk missing/unknown
````

````
---

### Issue 6: Decimal Separator

```text
WRONG (in some locales):
Year | Day | Jan
2020 | 1   | 12,5     ← Comma decimal (local format)

CORRECT:
Year | Day | Jan
2020 | 1   | 12.5     ← Period decimal (standard)
```text

**Solution:** Save Excel dengan regional setting English (US)

---

## Data Quality Guidelines

### Recommended Checks Before Upload

#### 1. **Range Check**

```text
For daily rainfall in tropical region:
- 0-400 mm: Normal range
- 400-500 mm: Extreme, possible
- >500 mm: Very rare (typhoon), verify
- Negative: Always error
```text

#### 2. **Consistency Check**

```text
Within station:
- Monthly total = sum of daily values
- Seasonal pattern should make sense
- Extreme values should be documented

Between stations:
- Spatial correlation: nearby stations should correlate
- Distance effect: farther stations may differ more
```text

#### 3. **Completeness Check**

```text
For 3 stations (Target + Ref1 + Ref2):
- Target: 30-50% missing OK (that's why we impute)
- Ref1, Ref2: Should be 90%+ complete
- Common dates: Should overlap well

Target: 60% complete, Ref1: 95%, Ref2: 98% → GOOD
````
````

````
Target: 95% complete → Not worth imputing
Both Ref: 50% complete → Cannot use for imputation
```text

---

## Examples of Real Data

### Example 1: High Quality Data

```text
Year | Day | Jan | Feb | Mar
2020 | 1   | 5.2 | 8.3 | 12.1
2020 | 2   | 4.8 | 7.9 | 11.8
2020 | 3   |     | 7.5 | 11.5    ← 1 missing dalam 800+ values
Year | 4   | 4.1 | 7.1 | 11.1

Status: ✓ EXCELLENT - >99% complete
```text

### Example 2: Medium Quality Data

```text
Year | Day | Jan | Feb | Mar
2020 | 1   | 5.2 |     | 12.1
2020 | 2   |     | 7.9 | 11.8
2020 | 3   | 3.8 |     | 11.5    ← Some missing scattered
2020 | 4   |     | 6.1 |

Status: ✓ ACCEPTABLE - ~90% complete
        Can use as reference station
```text

### Example 3: Poor Quality Data

```text
Year | Day | Jan    | Feb | Mar
2020 | 1   | 5.2    |     | 12.1
2020 | 2   |        |     |
2020 | 3   | 3.8    |     |         ← Missing in clusters
2020 | 4   |        | 6.1 |
2020 | 5   |        |     | 13.2
...  | ... | ...    | ... | ...

Status: ✗ NOT SUITABLE - <70% complete
        Cannot use as reference
```text

### Example 4: Problematic Values

```text
Year | Day | Jan    | Feb  | Mar
2020 | 1   | 5.2    | 8.3  | 12.1
2020 | 2   | N/A    | 7.9  | 11.8    ← TEXT! Invalid
````

```
2020 | 3   | 3.8    | -5.0 | 11.5    ← NEGATIVE! Invalid
2020 | 4   | 1200   | 6.1  | 11.1    ← EXTREME! Verify

Status: ✗ MUST FIX before upload
        Replace N/A with blank
        Remove negative values
        Verify extreme values
```text

---

## Multi-Year Data Handling

### Scenario: 3 Years of Data (2019-2021)

**Option A: Concatenate Rows**

```excel
Year | Day | Jan   | Feb   | Mar
2019 | 1   | 5.2   | 8.3   | 12.1
2019 | 2   | 4.8   | 7.9   | 11.8
...
2019 | 365 | 6.1   | 9.2   | 13.5
2020 | 1   | 5.5   | 8.5   | 12.3
2020 | 2   | 5.0   | 8.1   | 12.0
...
2021 | 365 | 6.3   | 9.5   | 13.8

Total rows: 3×365 = 1095 data points
Status: ✓ Recommended format
```text

**Option B: Single Year Highest Quality**

```excel
Year | Day | Jan   | Feb   | Mar
2020 | 1   | 5.2   | 8.3   | 12.1
2020 | 2   | 4.8   | 7.9   | 11.8
...
2020 | 365 | 6.1   | 9.2   | 13.5

Total rows: 365 data points
Status: ✓ Also acceptable
        Less data but higher quality
```text

---

## Data Preparation Workflow

### Step 1: Source Data
Source dari BMKG, official weather stations, atau monitoring networks

### Step 2: Initial Cleaning

```text
□ Remove duplicates
□ Fix formatting issues
□ Handle typos atau text values
□ Convert to consistent decimal format
□ Validate ranges
```text

### Step 3: Structure Conversion

```text
Original format (BMKG):
Date | Station_A | Station_B
2020-01-01 | 12.5 | 14.3
2020-01-02 | 11.2 |

Convert to:
Year | Day | Jan | Feb | Mar | ...
2020 | 1   | 12.5 |
2020 | 2   | 11.2 |
```text

### Step 4: Excel Export

```text
□ Save format: .xlsx (Excel 2007+)
□ Verify header row
□ Check data types
□ Confirm no merged cells
□ Validate total rows/columns
```text

### Step 5: Pre-Upload Validation

```text
□ Open file: Confirm readable
□ Check first & last rows visible
□ Scroll through: Look for anomalies
□ Column alignment: Verify headers
□ Data range: Check values are reasonable
```text

### Step 6: Upload to RainfallFussion

```text
□ Add coordinates (accuracy ±0.01°)
□ Upload Target file
□ Upload Ref1 file
□ Upload Ref2 file
□ Confirm all loaded
□ Process & download results
```text

````

Advanced: DateTime to Day Number Conversion

If you have dates but need "Day" number:

Excel Formula (Column B):

```text
=A2
(where A2 contains date)

Then format as integer
This gives day number within year
```text

**Python Script:**

```python
from datetime import datetime

date = datetime(2020, 1, 15)
day_number = date.timetuple().tm_yday  # Returns 15
```text

**Google Sheets:**

```text
=SEQUENCEIF matching date in column A
Or use helper column with DAYOFYEAR()
```text

---

## File Size & Performance

### Data Volume Guidelines

```text
Rows    | Years | Daily Points | Processing Time | File Size
365     | 1     | 365          | <100 ms        | 50 KB
1095    | 3     | 1095         | 150 ms         | 120 KB
3650    | 10    | 3650         | 500 ms         | 400 KB
7300    | 20    | 7300         | 1000 ms        | 800 KB
```text

**Recommended:** 365-3650 rows (1-10 years)
**Maximum tested:** 10000+ rows (works but slow)

---

## Troubleshooting Data Upload Errors

### Error: "Error membaca file"
````

```text
Most likely causes:
1. File format bukan .xlsx
2. Header row tidak sesuai spesifikasi
3. Empty rows di tengah data
4. Merged cells ada di data area
5. File corrupted/incomplete

Fix:
- Re-save dalam .xlsx format
- Check header exact match
- Remove empty rows
- Unmerge cells
- Copy-paste data ke file baru
```text

### Error: "Nilai invalid atau NaN di hasil"

```text
Causes:
1. Data columns tidak align (month data dimulai dari column D bukan C)
2. Negative atau extremely large values
3. Month columns dalam order salah

Fix:
- Verify column alignment (C=Jan, D=Feb, ... N=Dec)
- Check dan remove illogical values
- Ensure month order left-to-right
```text

### Error: "File terunduh tapi tidak terbuka"

```text
Causes:
1. File tidak fully download
2. Browser download settings
3. File corrupted saat transfer

Fix:
- Clear browser cache
- Try different browser
- Check file size setelah download
- Re-run process dan download ulang
```text

---

## Summary Checklist

Quick checklist sebelum upload data:

```text
Format & Structure:
```

```
☐ File is .xlsx format
☐ Exactly 1 sheet (or data di sheet first)
☐ Header row: Year | Day | Jan | Feb | ... | Dec
☐ Data starts di row 2

Data Quality:
☐ Year column: 4-digit integers
☐ Day column: 1-366 integers
☐ Monthly values: nonnegative numbers
☐ Missing values: completely BLANK
☐ NO text values like "N/A", "missing"
☐ NO negative values
☐ NO extreme outliers without verification

Completeness:
☐ Minimal 5 rows data (ideally 365+)
☐ Target station: ≥30% complete
☐ Reference 1: ≥80% complete
☐ Reference 2: ≥80% complete
☐ Data overlaps across 3 stations well

Formatting:
☐ No merged cells
☐ No empty rows dalam data
☐ Decimal separator: period (.)
☐ Currency or special formatting removed
```text

---

**Dokumentasi Format Data Selesai!**

Untuk bantuan lebih lanjut:
- Lihat contoh data di folder `/examples/`
- Hubungi admin untuk validasi data
- Gunakan template yang disediakan
```
```