weatherModel.py README

In the main function, define the list of diseases to measure risk for as well as a threshold for the minimum number of consecutive favorable days needed for disease to propagate. Then, in a for loop, the following 4 methods are executed per disease:

1. sortCandidatePlots(disease):
   1. Method to remove days where weather conditions are unvfavorable for a given disease, per plot
   2. Input: disease parameter to be provided from list of diseases in main, WeatherData.xlsx, RicePestAndDisease.xlsx
   3. Output: dataframe of weather data (per plot) with entries that satisfy disease favorable conditions
2. countDiseaseDays(df):
   1. Method to find sequences of consecutive days + start date when disease conditions are favorable. For dates in a sequence, all dates except start date are removed
   2. Input: plots\_at\_risk dataframe, the output of sortCandidatePlots method (parameter)
   3. Output: dataframe of favorable dates entries, with StartDate and ConsecutiveDates columns added to show when sequence of consecutive dates began and how many consecutive dates there were in that sequence
3. outputRiskPlots(df, disease, threshold):
   1. Method to set threshold for minimum consecutive dates necessary to be considered risky and to sort plots into riskiness categories
   2. Input: dataframe output of countDiseaseDays, disease, threshold (parameters), and Paddy\_plots\_phase1.geojson for all plots in study area
   3. Output: dataframe of all plots in study area with risk score in Risk column
4. Validate(df, disease):
   1. Method to validate results of weatherModel and output them to geoJSON files per disease
   2. Input: merged dataframe - output of outputRiskPlots, disease (parameters), Yield\_reducing\_factors.xlsx for ground truth
   3. Output: 1 geoJSON for each disease with Risk column and Match column to show whether Risk prediction is correct