iSell Tool Documentation

- **Mapping Files**:
  Folder Name: masters
  1) InputConditionMaster.xlsx:

  It the main master file with different conditions. From this file all conditions are picked up and mapped in tool. It contains **4 sheets** as follows:

  i) **Accounts:** This sheet contains all account names with account manager, channel manager, hotel capacity, grid conditions, data conditions.
  ii) **Monthly_MinRates:** It contains minimum rate to be set for pricing grid and recommendations.
  iii) **Monthly_MaxRates:** It is max rate to be set for recommendation
  iv) **Monthly_Jump:**

  2)'cm_master.xlsx':

	L	M	N	O
	RezNext	**BookingHotel**	**TravelClick**	**stdname**
	ChannelName	BookingSource	Channel Name	Channel
us	Status	Booking Status	Reservation Status	Status
e	ArrivalDate	CheckInDate	Checkin Date	CheckIn
te	DepartureDate	CheckOutDate	Checkout Date	CheckOut
	NoofRooms	No Of Rooms	Rooms	No_of_Rooms
	NetAmt	Amount to pay	Revenue USD	Total_Amount

This file maps the required Column names in raw data with standard names. While adding new channel manager to above mapping, get respective columns from the **OTA or Transactional data** and add required column names to left hand side of the **stdname** column. Ensure the column '**stdname'** should present at the end. The column '**stdname'** contains the finalized standard names.

3)'DateFormats.xlsx':
Every channel manager contains different date formats. We need to specify those date formats in the DateFormats.xlsx mapping file. Formats should be provided as per the python standard. You can check correct date formats present in file by opening it in **notepad ++.** Wrong date format update in mapping will result wrong calculations. Be careful while updating the date formats.
The leftmost column **(CM**) tells date column name for which the format to be specified.

	A	B	C	D
1	**CM**	**Staah**	**AxisRooms**	**Maximojo**
2	CheckIn	%Y-%m-%d	%d/%m/%Y	%d-%b-%Y
3	CheckOut	%Y-%m-%d	%d/%m/%Y	%d-%b-%Y
4	extension	xlsx	csv	xlsx
5				

4)**'ezeedate_format**.xlsx': It is the mapping file which we maintain separately for eZee Channel manager as the OTA data is having different date formats for the hotels on eZee channel manager.

	A	B	C
	Hotel	Checkin	Checkout
	Lar Amorosa	%d-%m-%Y	%d-%m-%Y
	Varanda Do Mar, Panaji	%d/%m/%Y	%d/%m/%Y
	Vedifarm a Mountain Resort	%d/%m/%Y	%d/%m/%Y
	Hotel Park N	%d-%m-%Y	%d-%m-%Y
	Hotel El Paso	%d/%m/%Y	%d/%m/%Y

5)statuscodes.xlsx:

	A	B
1	**status**	**code**
2	C	1
3	M	1
4	B	0
5	Confirmed	1
6	Cancelled	0
7	Modified	1
8	On-hold	1
9	Provisional	1
10	Received	1
11	No Show	0

It contains Status name mapping, we map 0 for **Cancelled and NoShow status**, rest Status are set to 1(Confirmed, Modified, Provisional, …, etc). It is same as YPR_masters statuscodes file. Ensure both files have the same status mapped with same codes. Update statuscodes in isell masters as soon as you update statuscodes in YPR masters. Both isell and YPR statuscodes master files should be same.

6) DropColumns.xlsx:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
	CM	UK	TravelBoo	eZee	Maximojo	Djubo	AxisRooms	Staah	SiteMinde	AsiaTech	BookingC	ResAvenu	RezNext	BookingH	TravelClick
	1	No	No			No			No	No	No				
	2	OTA_Sold	OTA_Sold			Rate on CM			Rate on CI	Rate on CI	Rate on CM				
	3														
	4														
	5														

This mapping file contains the information of column names to be deleted from the iSell report. If no column to be deleted, then also you need to specify the channel manager name in this file.

7) zFactors.xlsx:

	A	B	C	D	E
1	JumpName	Standard	Aggressive	VeryAggressive	Val
2	Base	2	7	10	1
3	Short	3	8	15	2
4	High	4	9	20	3
5	Long	5	10	25	4
6					

This file contains information about Jumps and percent increase in the adjacent rates. By default, **Standard** is set to "**JumpType**" column in input condition master. Val is JumpName code i.e 1->Base, 2->Short, 3->High, 4->Long

8) Format2_iSells.xlsx:

	A	B	C	D
1	HotelNames			
2	YO1 India's Holistic Wellness Center			
3				
4				
5				

It contains the hotelnames which are having 2nd format of iSell in which OTA and Hotel Information is available in one report.

- **Modules**:
  'AccountSplitMerge.py',
  'beautiMode.py',
  'CMAs.py',
  'directRecs.py',
  'GRID.py',
  'iSellFormat2.py',
  'iSell_fun_02.py',
  'Leaf_Module.py',
  'monthlymodule.py',
  'ProcessFlow.py',
  'seasonalmodule.py',
  'simpleRecs.py',

  Execution Steps:

  1.Trigger Script(Create_iSell_Tool.py):
  It is the main file or Tool to run iSell Report. It contains standard paths of Folder Structure and values for some conditions.

  std paths:
  srcpth = Path of source code (git-hub)
  stdpth = Path of All_In_One_iSell folder
  path = Path of masters folder

  conditional values:
  Account_Managers = It is list which contain Standard Account Manager Name.
  logflag = Logging flag for DEBUG or INFO
  use DEBUG for debugging purpose, default is INFO.
  LR_date = It takes input from user , Last Report Run Date to get last report and calculate pickup comparing with Last Report OTA_Sold.

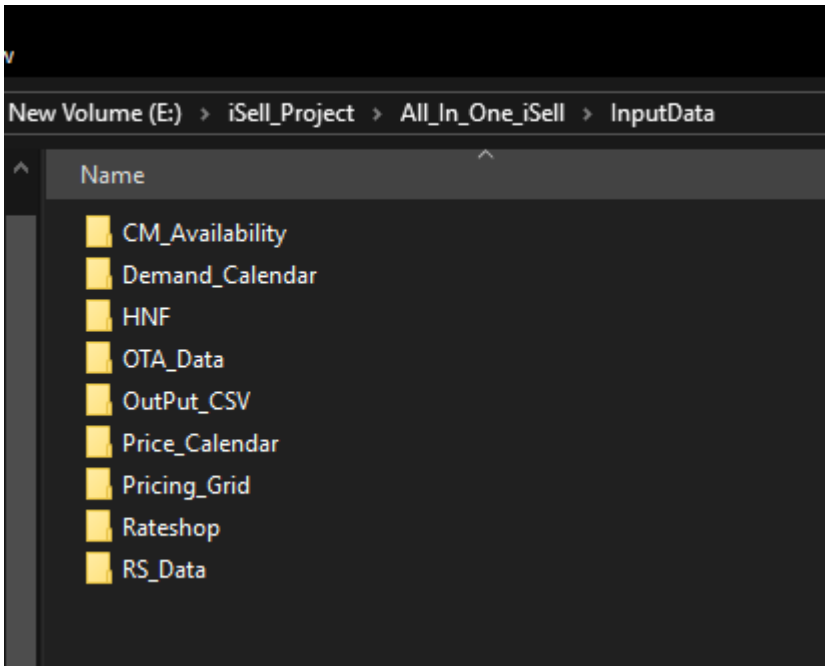
  By setting all above values in Trigger Script, python module **ProcessFlow** is called.

2. ProcessFlow:

The name indicates, It contains the flow of the process. The steps are as follows:

1. Logging conditions are set.
2. All master files read here and created all dictionaries to map with the data.
3. It will go in to **for loop** with the **hotelname** column present in the InputCondition Master.

Folder Structure of input files(InputData):



Process Flow For loop steps:

1. **format2flag:** It is binary flag**,** It will set format2flag by checking '_OTA' in name. This format is 2nd iSell format which contains OTA and Total Level information.
2. **Dynamic Dictionaries Creation:** It sets dictionaries and lists using standard hotel name.
   Calculation of variables and reading all input files:
   1. clustName : gets cluster name of the hotel in loop.
   2. htl_dowWt: It is hotel -> dow_weights dictionary.
   3. Isellrange : It is the isell window (180,365,etc), taken from **'isellwindow'** column in inputConditionsMaster.
   4. It will check Channel Manager from **name_chman** dictionary and read data files required for creating iSell.
      staahfile : It is variable, which reads **OTA data** for each channel manager.
      cmdata : It is CM_Availabilty data, contains availability information, read from CM_Availability folder.
      pcdata : It is Price_Calander data, contains rates information and read from Price_Calendar folder.
      rsfile : Rate shopping data read from RS_Data folder, it is file with hotel and competitors rateshop data with filtered conditions.
      rateshopfile : It is complete rateshop with hotel and competitors, read from RateShop folder.
      dc : Demand calendar file read from Demand_Calendar folder**.**

df_PG : Pricing grid read from folder Pricing_Grid folder, if use_Grid = 1 in input conditions master file . Otherwise it will create Pricing Grid in tool itself and put in iSell.

All files are not required for all channel managers, in such case the variables are assigned **' '** value.

3. **Occupancy Conversion of OTA Data**:
   Occupancy conversion function dfconv present in iSell_fun_02 module, is called in this step. Cm_master, statuscodes, DateFormats file read and dictionaries (statuscode, stdcols, dtformat) created. Set the dateformats using dtformat dictionary.

   Occupancy conversion: Date range created using checkin and checkout dates.
   Ex. Checking ='1-Jul-2019' and checkout ='5-Jul-2019',
   then Occupancy dates would be 1-Jul-2019,2-Jul-2019,3-Jul-2019,4-Jul-2019,5-Jul-2019.
   Those extra entries will be created in data during occupancy conversion.

   Extra Column Calculations:

   dtDif column added: where dtDif = CheckOut – occupancydate. **Arrivals** column added. Replaced Blank **Channel** values with 'YourWeb'. Replaced **Blank Status** values with 'Cancelled'. Mapped statuscode mapping on Status column.

   Data Filtered as per dtDif > 0 , statuscode == 1.

   LOS column added to dataframe, where LOS = CheckOut – CheckIn. RevPD column added, where RevPD = Total_Amount/LOS. ADR column added, where ADR = Total_Amount/(LOS x No_of_Rooms).

   ttlsold, df_total, df_ota Calculations:

   a) **ttlsold : It contains** No_of_Rooms summed on occupancydate.
   b) **df_total : It contains** No_of_Rooms and RevPD summed on occupancydate.
   c) **df_ota : It contains** No_of_Rooms summed on summed on ['occupancydate','Channel'].

   These three dataframes are returned by **dfconv** function.

   df_ota2 = Pivoted the channel names.

   No_of_Rooms (OTA_Sold) and RevPD(OTA Revenue) merged on Date in ddff dataframe.

   Then otabreak, otasold and otarev dataframes are sliced from ddff.

   Demand calendar merged and capacity added to **dc3** dataframe.

4. **Attach Rooms Availability and Flow through Availability:**
   CM availability calculations are done in CMAs module.
   CMAs Module:
   It contains modules for all channel managers to calculate Rooms Availability, Flow through Availability and RateOnCM.

CM_Avails function: It returns two dataframe (dfa,dfb)

dfa(rmsavail): It contains rooms availability and flow through availability merged on date column. dfb(cmdf): It contains Rate on CM column merged on date column.

iSelldf1 : dc3 <- rmsavail (availability merged on dc3)

iSelldf2 : iSelldf1 <- otasold

***Hybrid check**:

If **GridType** is **Hybrid**:

HNF created on the fly and dumped in today's folder under HNF, by calculating Sold = capacity-availability. **HNF** column should be **Yes** in InputConditionsMaster. It is HNF based Pricing.

Last_OTASOLD and **LAvg** calculations:

dfLR function in iSell_fun_02 , takes **last iSell** report and returns **Last OTA_Sold** and **Last Rateshop Average(LAvg)**.

Last_szrates(last Seasonal Rates): Read **SeasonalRate_y** column from last iSell and renamed this column with Last_szrates, to compare recommendations with this column for the channel managers not having RateOnCM column (cmflag = 0).

Pickup Calculation:

Pickup = OTA_SOLD (current) - Last_OTASOLD

Revenue merged with **iSelldf2.**

ADR_OTB = OTA Revenue / OTA_Sold

5. **monthlymodule.py:**

PricingType should be **Monthly**:
Module monthlymodule.py,
This module returns min_rate,max_rate calculations.
Returns two dataframes , one is iSell with min,max rates and other for grid calculation.

Monthly Minimum Rate, Monthly jump dictionaries calculated.
min_rate calculated using **dow_weights,**
where min_rate = Min_Rate * dow_weights

a) Use_MaxRate = 0:
   max_rate calculated using compound_interest function using min_rate, Jump factor, interval.
b) Use_MaxRate = 1:
   This condition needs to calculate the jump factor first, jump factor is calculated using ciRate function using Min_Rate, Max_Rate, interval. By using those calculated jump factors, max_rate is calculated.

Two dataframes returned to process flow.

6. **Pricing Algorithms for Recommendations:**
   These are used to reflect recommended rate in iSell. What and which rate from Grid should be reflected in iSell.

   Pricing is divided in to three types:
   1) Simple
   2) HNF Based
   3) Non HNF Based

   All algos are explained in iSell InputCondition and Pricing_Document.

7. **RateShop Addition:**
   Lowest rate of client hotel, Rateshopping Table with conditions are merged to the iSell, from Rateshop module.

8. **Market_Trend:**
   It is the difference between average of all rates (client and competitor rates) from last report and current report average rates. Market Trend = Current Average- Last Average

9. **Merging of OTA Channel Names:**
   OTA information (**otabreak**) is merged with iSelldf7. This contains information of channel names with number of confirmed rooms.

10. **Drop columns which are not required.**

11. **Adoption:**
    It indicates percentage of rates being adopted. Recommendations matching with the Rate On CM column are kept blank. Adoption for 180 and 90 days is shown in Copy iSell on the top.
    Adoption dataframe is returned from **Adopcal** function in **iSell_fun_02.py** module**.** Before passing to the daysadop function in Adopcal module, the dataframe is filtered with two conditions,
    Pickup should be 0 and Recommended Rate column should not be blank.

12. **Rate on CM Check and iSell CSV dump:**
    Rate on CM Check: Sometimes the master rate code changes in CM Availability file, the Rate on CM column reflects as blank. If the Rate On CM column is blank in iSell dataframe, it is dumped in the Output_CSV with name **BAD**. It can be fixed by changing the rate plan code in InputConditionMaster by copying the rate code name from CM_Availability file.

    iSell CSVs with today's date are dumped under OutPut_CSV inside todays date folder.

13. **iSell Beautification:**
    Separate module for iSell beautification(beautiMode.py)
    It contains two sub modules: 1. isellbeautify, 2. beautify

    1. **isellbeautify:** Precalculations like 10% of hotel capacity value, isell dataframe slicing as per the isell window and client window is done here. It will create todays folder in iSell folder and account manager name folder inside today's folder. After that the isell dataframe is passed to **beautify** function for beautification. **beautify** function is called inside **isellbeautify** function**.**

2. **beautify:**

Here two modules for excel workbook manipulation are used.
1. Openpyxl, 2. Xlsxwriter

Firstly, dataframe is converted to Workbook Objects.
Conditional formatting done using column names. So column name and position dictionaries are created for map the location and vice-versa.

Writer object created for client and internal isell.
Dataframe directly pushed to 6th row by keeping first 5 rows for iSell image and logo.
Different formats for conditional formatting are defined:

Conditional Formatted Columns:
Dow: Saturday and Sunday are highlighted as **dark pink background colour (format4)**

Event: Set Event column width to 30

Rooms Avail to Sell Online: Column values with zero and less than zero values are highlighted as **Red background colour (format1)**

Hotel Availability: The values less than 10% of hotel availability are highlighted as yellow background colour (**format8**)

Flow_through: Column values with zero and less than zero values are highlighted as **Red background colour (format1)**

OTA_Sold: added **dark pink background colour (format4)** for all values.

Pickup: Values greater than zero are highlighted **as green background colour (format2) i.e Pickup indication.** Values less than zero are highlighted **as red background colour (format1) i.e wash indication.**

OTA Revenue to Lowest_Rate : Background colour set to **thin Orange (format5)**

Rateshopping : Column width set 15 and Background colour set to **thin blue (format7)**

Market_Trend_Arrows: If the trend value is high +ve it shows up arrow, -ve it shows down arrow. Intermediate will show horizontal arrow.

Columns after Market Trend: Background colour set to thin green(format6).

border_form : It sets border for each column and alignment to right.

iSell image and logo: inserts iSell image and logo from logo folder with specified positions.

Glossary addition to Glossary sheet:

Creates Glossary sheet to writer object and sets column widths for A,B,C as 5,32 and 46 respectively. Sets border formatting to glossary using **gformat**.

Bold the header row cells (in Header Formatting) by specifying that row. Format **boldcell** is used to bold the header rows in glossary.

Grid Addition to Copy iSell :

Creates new Grids sheet to writer object and puts Grid dataframe as it is.

Rateshop sheet Addition to iSell :

Creates new **Rateshop** sheet to writer object and puts Rateshop dataframe as it is.

14. **Format2 iSell:**

    It checks **format2flag**, this flag is set at the starting of for loop. If it finds '_OTA' string in hotelname then sets flag as 1. In channel manager condition there are two separate conditions to slice OTA data for **format2flag =1** and **format2flag=0**.

    if it is 1**:** It goes for **total_ota_merging** module.
    Two iSells (OTA level and Total Level) are merged in format2 iSell.
    The merged iSell is again passed to beautification module. It will dump in to iSell folder with **_Combine_iSell** name.