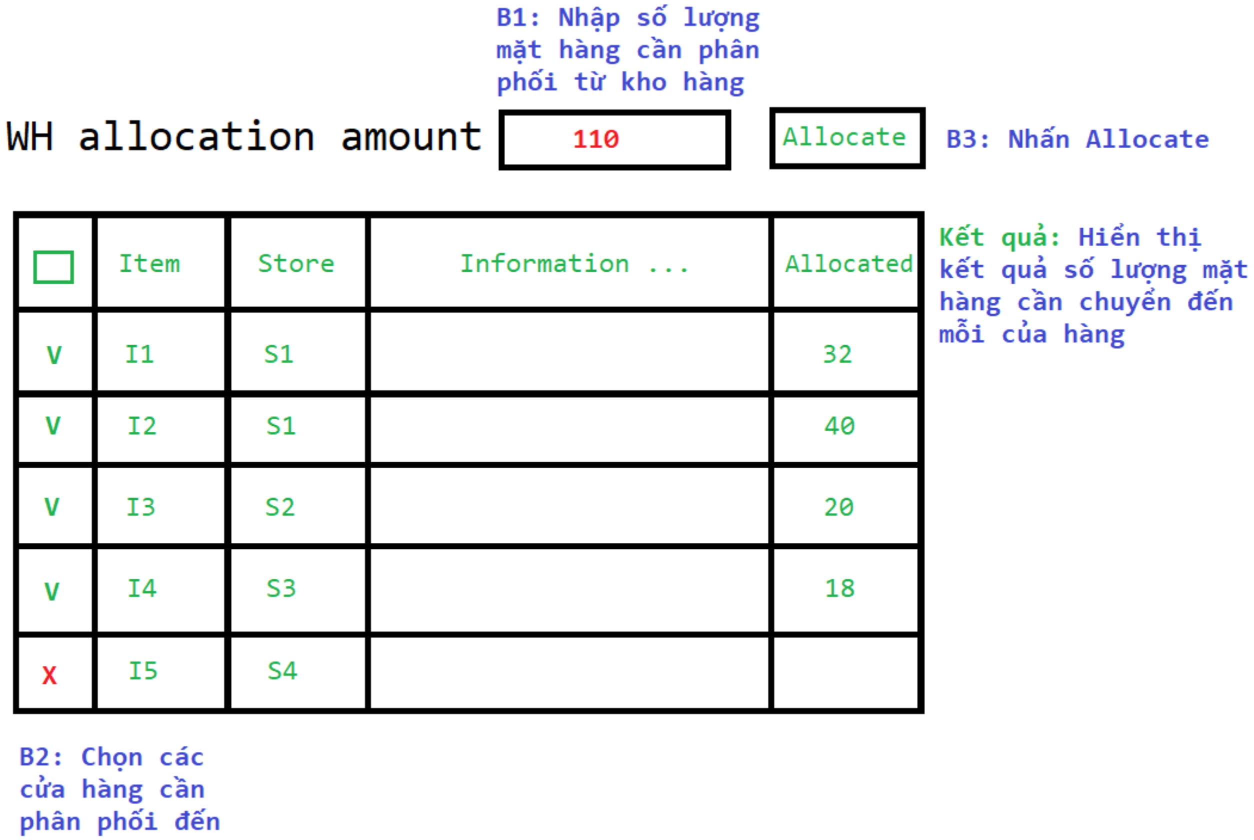
# LESSON 15 – KIEM TRA – D2

Trong các hệ thống siêu thị, mỗi ngày sẽ có sự phân phát hàng hóa từ các kho hàng đến các cửa hàng. Việc phân phát số lượng bao nhiêu sẽ dựa vào tình hình bán hàng thực tế tại mỗi của hàng trong thời gian gần đây và được quyết định bởi chủ kho hàng

Công việc tính toán số lượng mặt hàng cần phải chuyển đến cho mỗi cửa hàng phải trải qua rất nhiều bước tính toán. Thay vì việc tính toán lặp lại hàng ngày qua các tập tin word, excel rất mất thời gian và dễ sai sót. Chủ hệ thống yêu cầu chúng ta xây dựng ứng dụng như sau



B1: Nhập số lượng mặt hàng cần phân phối từ kho hàng

WH allocation amount 110 Allocate

B3: Nhấn Allocate

| □ | Item | Store | Information ... | Allocated |
|---|---|---|---|---|
| V | I1 | S1 |  | 32 |
| V | I2 | S1 |  | 40 |
| V | I3 | S2 |  | 20 |
| V | I4 | S3 |  | 18 |
| X | I5 | S4 |  |  |

Kết quả: Hiển thị kết quả số lượng mặt hàng cần chuyển đến mỗi của hàng

B2: Chọn các cửa hàng cần phân phối đến

**Thông tin Store:** storeId, referenceStoreId, stockPreviousDay, expectedSales, isSelected

Dữ liệu vào: Số lượng mặt hàng cần cấp phát từ kho hàng – Warehouse Allocation Amount(110)

Thực hiện: Tính toán qua 4 bước

Kết quả: Số lượng mặt hàng được cấp phát đến mỗi cửa hàng – Allocated Amount/Store

==============================================================

Việc tính toán được mô tả như sau:

Step One: Filling in missing “**Expected Sales”** values

Use Case 1: If a selected store is missing expected sales values and has a reference store maintained → use expected sales of reference store

Use Case 2: If a store selected is missing expected sales values and has

a) no reference store maintained OR

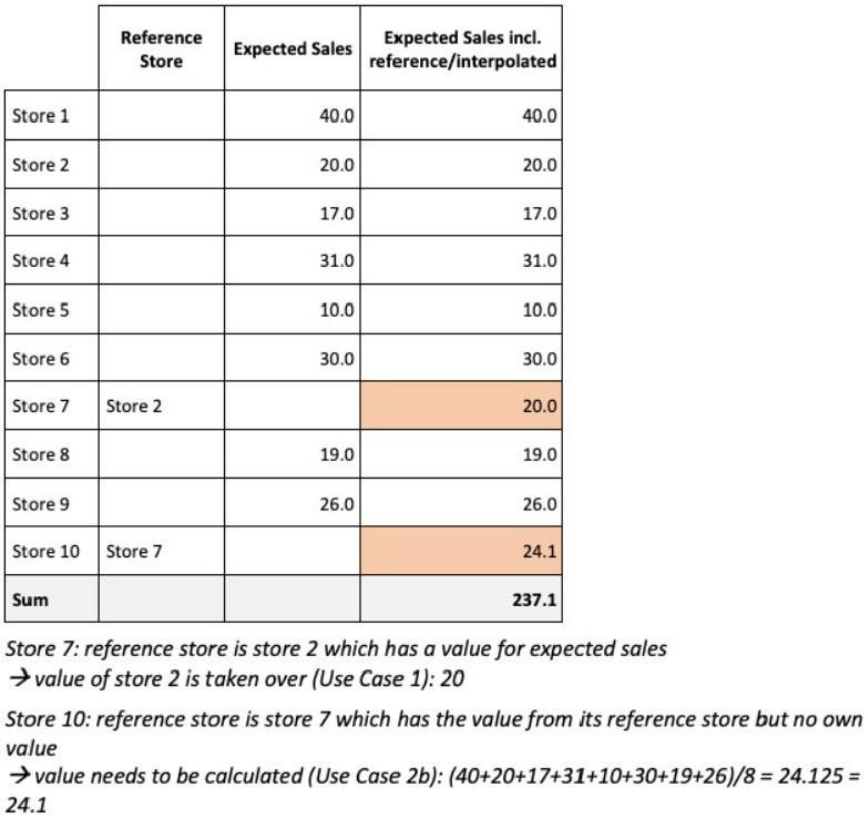
b) a reference store maintained which is also missing expected sales itself

only values of reference store itself count here. It is not considered if reference store would have a reference store itself.

→ take the average of all stores with “own” data (stores with data from reference store or calculated data are skipped for average calculation); commercially rounded to one decimal digit

→ expected sales calculated with average should be calculated and used for further calculation

## Example Step One



| | Reference Store | Expected Sales | Expected Sales incl. reference/interpolated |
|---|---|---|---|
| Store 1 | | 40.0 | 40.0 |
| Store 2 | | 20.0 | 20.0 |
| Store 3 | | 17.0 | 17.0 |
| Store 4 | | 31.0 | 31.0 |
| Store 5 | | 10.0 | 10.0 |
| Store 6 | | 30.0 | 30.0 |
| Store 7 | Store 2 | | 20.0 |
| Store 8 | | 19.0 | 19.0 |
| Store 9 | | 26.0 | 26.0 |
| Store 10 | Store 7 | | 24.1 |
| **Sum** | | | **237.1** |

*Store 7: reference store is store 2 which has a value for expected sales*
*→value of store 2 is taken over (Use Case 1): 20*

*Store 10: reference store is store 7 which has the value from its reference store but no own value*
*→value needs to be calculated (Use Case 2b): (40+20+17+31+10+30+19+26)/8 = 24.125 = 24.1*

If ALL selected stores have no expected sales and no reference store maintained (and thus a calculation of expected sales is not possible):

→ stop the calculation

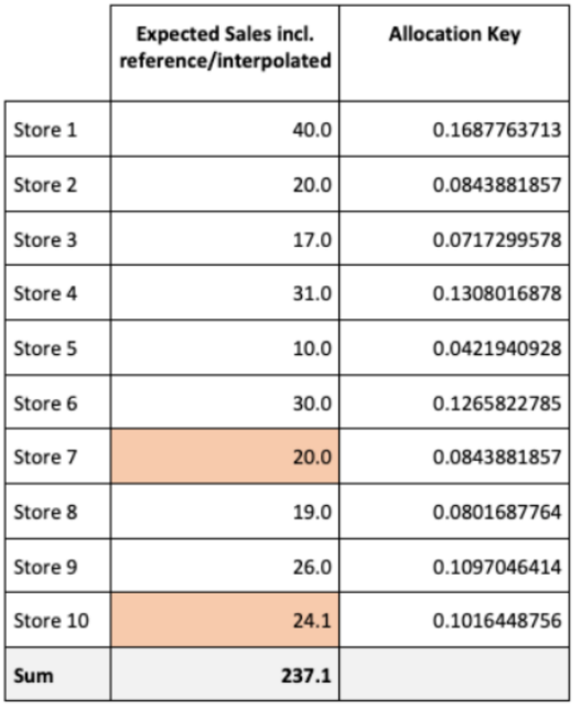
→ display a localizable error message in red: "Expected sales cannot be calculated. Please add a reference store or include stores with expected sales for interpolation"

### Step Two: Calculation of “**Allocation Key**”

Allocation Key Store = (Expected Sales incl. Reference Interpolated Store ) / (Σ Expected Sales incl. Reference Interpolated Store )

→ for further calculation results should be commercially rounded to max. 10 decimal digits

## Example Step Two



| | Expected Sales incl. reference/interpolated | Allocation Key |
|---|---|---|
| Store 1 | 40.0 | 0.1687763713 |
| Store 2 | 20.0 | 0.0843881857 |
| Store 3 | 17.0 | 0.0717299578 |
| Store 4 | 31.0 | 0.1308016878 |
| Store 5 | 10.0 | 0.0421940928 |
| Store 6 | 30.0 | 0.1265822785 |
| Store 7 | 20.0 | 0.0843881857 |
| Store 8 | 19.0 | 0.0801687764 |
| Store 9 | 26.0 | 0.1097046414 |
| Store 10 | 24.1 | 0.1016448756 |
| **Sum** | **237.1** | |

*Store 1: Allocation Key = 40.0/237 = 0.1687763713*

### Step Three: Calculation of “Amount Allocated”

Amount Allocated Store = Allocation Key Store * (Allocation Amount + Σ Stock Previous Day Store ) - Stock Previous Day Store

□ Allocation Amount is the amount the user added to the newly created field on top of the table

□ ΣStock of Previous Day Store is the sum over all stores that are selected via checkbox in Checkbox Column

□ results should be commercially rounded to an integer

□ negative results should be replaced with 0 and also considered as 0 when calculating further steps

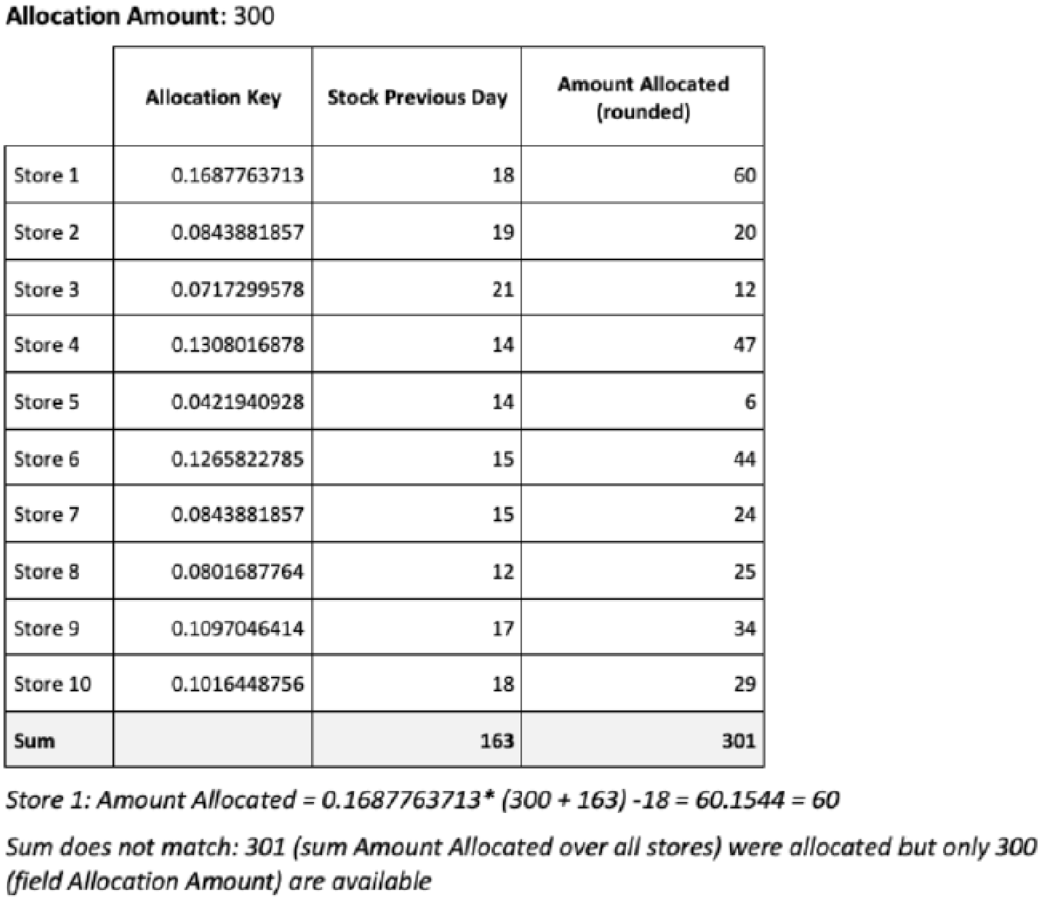
→ If sum of rounded Amounts Allocated over all stores checked via checkbox matches Allocation Amount added in field above the table:

□ add values to DC column of respective stores – show the result of store allocated amount in console

→ If sum does not match:

□ calculation needs to be continued with “Step Four”.

## Example Step Three



**Allocation Amount**: 300

| | Allocation Key | Stock Previous Day | Amount Allocated (rounded) |
|---|---:|---:|---:|
| Store 1 | 0.1687763713 | 18 | 60 |
| Store 2 | 0.0843881857 | 19 | 20 |
| Store 3 | 0.0717299578 | 21 | 12 |
| Store 4 | 0.1308016878 | 14 | 47 |
| Store 5 | 0.0421940928 | 14 | 6 |
| Store 6 | 0.1265822785 | 15 | 44 |
| Store 7 | 0.0843881857 | 15 | 24 |
| Store 8 | 0.0801687764 | 12 | 25 |
| Store 9 | 0.1097046414 | 17 | 34 |
| Store 10 | 0.1016448756 | 18 | 29 |
| **Sum** | | **163** | **301** |

*Store 1: Amount Allocated = 0.1687763713* (300 + 163) -18 = 60.1544 = 60*

*Sum does not match: 301 (sum Amount Allocated over all stores) were allocated but only 300 (field Allocation Amount) are available*

## Step Four: Fix Rounding Issues

Store Allocated Amount need to be corrected until the sum over all selected stores matches value in field Warehous Allocation Amount.

First Demand per store needs to be identified via following formula:

Demand Store = Expected Sales incl. Reference Interpolated Store - Stock Previous Day Store

□ results should be commercially rounded to an integer

□ negative results should be replaced by 0 and also considered as 0 when calculating further steps

Afterwards it needs to be determined if sum of Store Allocated Amount over all stores selected via checkbox is bigger or smaller than value in field Allocation Amount

□ Σ Amounts Allocated Store > Allocation Amount

□ Iterative: Subtract one allocation unit (one case) from the store which’s difference between Amount Allocated and Demand is biggest

□ if a store has “Amount Allocated” = 0, it is skipped and no difference is calculated

□ only positive values should be considered for difference; if difference of every store results into negative values, the one closest to 0 should be taken into account

□ If two or more stores have the same difference: take the one with smallest Demand

□ If two or more stores also have same Demand: take the one with smallest Expected Sales incl. reference/interpolated

□ If two or more stores also have same Expected Sales incl. reference/interpolated: take the one with smallest store number

□ After subtracting one allocation unit from a store, the difference between Amount Allocated and Demand is calculated again for this store and the iterative approach is continued until sum of Amounts Allocated over all stores selected via checkbox equals value of field Allocated Amount

□ Σ Amounts Allocated Store < Allocation Amount

□ Iterative: Add one allocation unit (one case) from the store which’s difference between Amount Allocated and Demand is smallest.

□ the smallest value, also negative value, should count as smallest difference

□ If two or more stores have the same difference: take the one with biggest Demand

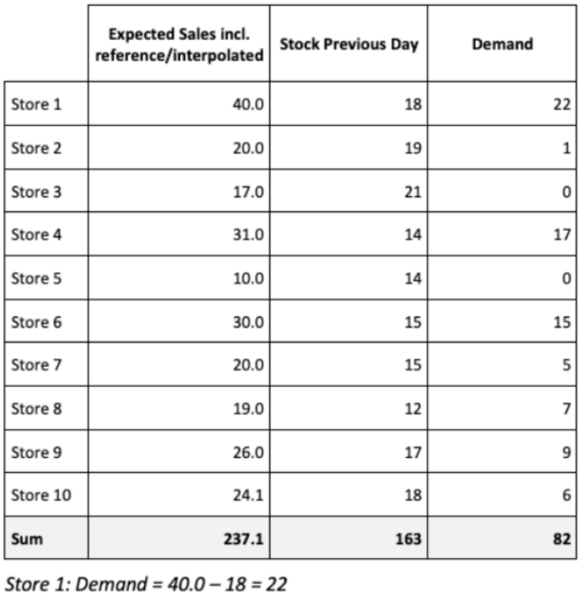
□ If two or more stores also have same Demand: take the one with biggest Expected Sales incl. reference/interpolated

□ If two or more stores also have same Expected Sales incl. reference/interpolated: take the one with smallest store number

□ After adding one allocation unit to a store, the difference between Amount Allocated and Demand is calculated again for this store and the iterative approach is continued until sum of Amounts Allocated over all stores selected via checkbox equals value of field Allocated Amount

□ final results should be commercially rounded to an integer

□ negative results should be replaced with 0 → add values to DC column of respective stores – show the result of store allocated amount in console



| | Expected Sales incl. reference/interpolated | Stock Previous Day | Demand |
|---|---|---|---|
| Store 1 | 40.0 | 18 | 22 |
| Store 2 | 20.0 | 19 | 1 |
| Store 3 | 17.0 | 21 | 0 |
| Store 4 | 31.0 | 14 | 17 |
| Store 5 | 10.0 | 14 | 0 |
| Store 6 | 30.0 | 15 | 15 |
| Store 7 | 20.0 | 15 | 5 |
| Store 8 | 19.0 | 12 | 7 |
| Store 9 | 26.0 | 17 | 9 |
| Store 10 | 24.1 | 18 | 6 |
| **Sum** | **237.1** | **163** | **82** |

*Store 1: Demand = 40.0 – 18 = 22*

==================================================

Reference to Excel file for calculation steps and testing result