ECOMMERCE DAX FORMULAS

1. month_name = FORMAT(date_table[Date],"mmm")
2. month_number = MONTH(date_table[Date])
3. year = YEAR(date_table[Date])
4. profit colour = IF([yoy profit]>0,"green","red")
5. profit icon = var positive_icon =UNICHAR(9650)
   VAR negative_icon=UNICHAR(9660)
   VAR result=IF([yoy profit]>0,positive_icon,negative_icon)
   RETURN result
6. profit margin
   =SUM(ecommerce_data[profit_per_order])/SUM(ecommerce_data[sales_per_order])
7. profit margin colour = IF([yoy profit margin]>0,"green","red")
8. pydt profit =
   CALCULATE(SUM(ecommerce_data[profit_per_order]),DATESYTD(SAMEPERIODLASTYEAR(date_table[Date])))
9. pydt profit margin = CALCULATE([profit margin],DATESYTD(SAMEPERIODLASTYEAR(date_table[Date])))
10. pydt quantity =
    CALCULATE(SUM(ecommerce_data[order_quantity]),DATESYTD(SAMEPERIODLASTYEAR(date_table[Date])))
11. pydt sales =
    CALCULATE(SUM(ecommerce_data[sales_per_order]),DATESYTD(SAMEPERIODLASTYEAR(date_table[Date])))
12. quanity icon = var positive_icon =UNICHAR(9650)
    VAR negative_icon=UNICHAR(9660)
    VAR result=IF([yoy quantity]>0,positive_icon,negative_icon)
    RETURN result
13. quantity colour = IF([yoy quantity]>0,"green","red")
14. sales colour = IF([yoy sales]>0,"green","red")
15. sales profit margin icon = var positive_icon =UNICHAR(9650)
    VAR negative_icon=UNICHAR(9660)
    VAR result=IF([yoy profit margin]>0,positive_icon,negative_icon)
    RETURN result
16. trend = var positive_icon =UNICHAR(9650)
    VAR negative_icon=UNICHAR(9660)
    VAR result=IF([yoy sales]>0,positive_icon,negative_icon)
    RETURN result
17. yoy profit = ([ytd profit]-[pydt profit])/[pydt profit]
18. yoy profit margin = ([ytd profit margin]-[pydt profit margin])/[pydt profit margin]
19. yoy quantity = ([ytd quantity]-[pydt quantity])/[pydt quantity]
20. yoy sales = ([ytd sales]-[pydt sales])/[pydt sales]
21. ytd concanunated quantity = CONCATENATE("#",FORMAT([ytd quantity]/1000,"0.0,k"))
22. ytd profit = TOTALYTD(SUM(ecommerce_data[profit_per_order]),date_table[Date])
23. ytd profit margin = TOTALYTD([profit margin],date_table[Date])
24. ytd quantity = TOTALYTD(SUM(ecommerce_data[order_quantity]),date_table[Date])
25. ytd sales = TOTALYTD(SUM(ecommerce_data[sales_per_order]),date_table[Date])

BANK LOAN PROJECT DAX

```
1.  Bad Loan Applications = CALCULATE([Total Loan Applications],bank_loan_data[good vs
    bad loans]="bad loan")
2.  Bad Loan Funded Amount = CALCULATE([total funded amount],bank_loan_data[good vs bad
    loans]="bad loan")
3.  bad loan percentage = CALCULATE([Total Loan Applications],bank_loan_data[good vs
    bad loans]="bad loan")/[Total Loan Applications]
4.  Bad Loan Received Amount = CALCULATE([total amount received],bank_loan_data[good vs
    bad loans]="bad loan")
5.  Good Loan Applications = CALCULATE([Total Loan Applications],bank_loan_data[good vs
    bad loans]="good loan")
6.  Good Loan Funded Amount = CALCULATE([total funded amount],bank_loan_data[good vs
    bad loans]="good loan")
7.  good loan percentage = CALCULATE([Total Loan Applications],bank_loan_data[good vs
    bad loans]="good loan")/[Total Loan Applications]
8.  Good Loan Received Amount = CALCULATE([total amount received],bank_loan_data[good
    vs bad loans]="good loan")
9.  mom avg dti = ([Mtd avg dti]-[pmtd avg dti])/[pmtd avg dti]
10. mom intrest rate = ([Mtd avg intrest]-[pmtd intrest rate])/[pmtd intrest rate]
11. mom loan application = ([Mtd loan applications]-[pmtd loan applications])/[pmtd
    loan applications]
12. mom loan funded = ([Mtd funded amount]-[pmtd loan funded])/[pmtd loan funded]
13. mom loan received = ([Mtd amount received]-[pmtd loan received])/[pmtd loan
    received]
14. Mtd amount received = CALCULATE(TOTALMTD([total amount received],date_table[Date]))
15. Mtd avg dti = CALCULATE(TOTALMTD([avg dti],date_table[Date]))
16. Mtd avg intrest = CALCULATE(TOTALMTD([avg intrest rate],date_table[Date]))
17. Mtd funded amount = CALCULATE(TOTALMTD([total funded amount],date_table[Date]))
18. Mtd loan applications = CALCULATE(TOTALMTD([Total Loan
    Applications],date_table[Date]))
19. pmtd avg dti = CALCULATE([avg dti],DATESMTD(DATEADD(date_table[Date],-1,MONTH)))
20. pmtd intrest rate = CALCULATE([avg intrest
    rate],DATESMTD(DATEADD(date_table[Date],-1,MONTH)))
21. pmtd loan applications = CALCULATE([Total Loan
    Applications],DATESMTD(DATEADD(date_table[Date],-1,MONTH)))
22. pmtd loan funded = CALCULATE([total funded
    amount],DATESMTD(DATEADD(date_table[Date],-1,MONTH)))
23. pmtd loan received = CALCULATE([total amount
    received],DATESMTD(DATEADD(date_table[Date],-1,MONTH)))
24. total amount received = SUM(bank_loan_data[total_payment])
25. total funded amount = SUM(bank_loan_data[loan_amount])
26. Total Loan Applications = COUNT(bank_loan_data[id])
27. month = FORMAT(date_table[Date],"MMM")
28. month_number = MONTH(date_table[Date].[Date])
29. select measure = {
        ("total amount received", NAMEOF('bank_loan_data'[total amount received]), 0),
        ("total funded amount", NAMEOF('bank_loan_data'[total funded amount]), 1),
        ("Total Loan Applications", NAMEOF('bank_loan_data'[Total LoanApplications]),2)
    }
```

HEALTH CARE DAX FORMULS

1. day name = FORMAT(date_table[Date],"ddd")
2. month &year = date_table[month_name]& " "&date_table[year]
3. month number = MONTH(date_table[Date])
4. month_name = FORMAT(date_table[Date],"mmm")
5. weekday = WEEKDAY(date_table[Date],2)
6. year = YEAR(date_table[Date])
7. Admission hour = HOUR('Hospital ER_Data'[Patient Admission Date])
8. Avg wait time = FORMAT(AVERAGE('Hospital ER_Data'[Patient Waittime]),"0.0")& " "
   &"min"
9. no of patients = DISTINCTCOUNT('Hospital ER_Data'[Patient Id])
10. Patient Admission Status = IF('Hospital ER_Data'[Patient Admission Flag] = TRUE(),
    "Admitted", "Not Admitted")
11. patient admit date = DATE(YEAR('Hospital ER_Data'[Patient Admission
    Date]),MONTH('Hospital ER_Data'[Patient Admission Date]),DAY('Hospital
    ER_Data'[Patient Admission Date]))
12. Patient Age Group =

```
VAR PatientAge = 'Hospital ER_Data'[Patient age]
RETURN
    SWITCH(
        TRUE(),
        PatientAge >= 100, "100+",
        PatientAge >= 90 && PatientAge <= 99, "90-99",
        PatientAge >= 80 && PatientAge <= 89, "80-89",
        PatientAge >= 70 && PatientAge <= 79, "70-79",
        PatientAge >= 60 && PatientAge <= 69, "60-69",
        PatientAge >= 50 && PatientAge <= 59, "50-59",
        PatientAge >= 40 && PatientAge <= 49, "40-49",
        PatientAge >= 30 && PatientAge <= 39, "30-39",
        PatientAge >= 20 && PatientAge <= 29, "20-29",
        PatientAge >= 10 && PatientAge <= 19, "10-19",
        PatientAge >= 0 && PatientAge <= 9, "0-9",
        "Unknown"
    )
```

13. patient satisfaction = AVERAGE('Hospital ER_Data'[Patient Satisfaction Score])
14. Referred Patients =

```
 CALCULATE(
    COUNT('Hospital ER_Data'[Department Referral]),
    'Hospital ER_Data'[Department Referral] <> "NONE"
)
```

15. waiting intervel =
```
SWITCH(TRUE(),
 'Hospital ER_Data'[Admission hour]<2, "00-02",
 'Hospital ER_Data'[Admission hour]<4, "03-04",
 'Hospital ER_Data'[Admission hour]<6, "05-06",
 'Hospital ER_Data'[Admission hour]<8, "07-08",
 'Hospital ER_Data'[Admission hour]<10, "09-10",
 'Hospital ER_Data'[Admission hour]<12, "11-12",
 'Hospital ER_Data'[Admission hour]<14, "13-14",
 'Hospital ER_Data'[Admission hour]<16, "15-16",
 'Hospital ER_Data'[Admission hour]<18, "17-18",
 'Hospital ER_Data'[Admission hour]<20, "19-20",
 'Hospital ER_Data'[Admission hour]<22, "21-22",
 'Hospital ER_Data'[Admission hour]<24, "23-24",
 "Above 24"
)
```
16. waittime check = IF('Hospital ER_Data'[Patient Waittime]<=30,"Within target","Target missed")

HOSPATALITY(HOTEL) DAX FORMULAS

1. ADR = DIVIDE([Revenue], [Total Bookings],0)
2. ADR WoW change % =
```
Var selv =
IF(HASONEFILTER(dim_date[wn]),SELECTEDVALUE(dim_date[wn]),MAX(dim_date[wn]))
var revcw = CALCULATE([ADR],dim_date[wn]= selv)
var revpw =  CALCULATE([ADR],FILTER(ALL(dim_date),dim_date[wn]= selv-1))

return

DIVIDE(revcw,revpw,0)-1
```
3. Average Rating = AVERAGE(fact_bookings[ratings_given])
4. Booking % by Platform = DIVIDE([Total Bookings],
   CALCULATE([Total Bookings],
   ALL(fact_bookings[booking_platform])
    ))*100
5. Booking % by Room class = DIVIDE([Total Bookings],
   CALCULATE([Total Bookings],
   ALL(dim_rooms[room_class])
    ))*100
6. Cancellation % = DIVIDE([Total cancelled bookings],[Total Bookings])
7. DBRN = DIVIDE([Total Bookings], [No of days])
8. DSRN = DIVIDE([Total Capacity], [No of days])
9. DSRN WoW change % =
```
Var selv =
IF(HASONEFILTER(dim_date[wn]),SELECTEDVALUE(dim_date[wn]),MAX(dim_date[wn]))
var revcw = CALCULATE([DSRN],dim_date[wn]= selv)
var revpw =  CALCULATE([DSRN],FILTER(ALL(dim_date),dim_date[wn]= selv-1))

return

DIVIDE(revcw,revpw,0)-1
```
10. DURN = DIVIDE([Total Checked Out],[No of days])
11. No of days = DATEDIFF(MIN(dim_date[date]),MAX(dim_date[date]),DAY) +1
12. No Show rate % = DIVIDE([Total no show bookings],[Total Bookings])
13. Occupancy % = DIVIDE([Total Succesful Bookings],[Total Capacity],0)
14. Occupancy WoW change % =
```
Var selv =
IF(HASONEFILTER(dim_date[wn]),SELECTEDVALUE(dim_date[wn]),MAX(dim_date[wn]))
var revcw = CALCULATE([Occupancy %],dim_date[wn]= selv)
var revpw =  CALCULATE([Occupancy %],FILTER(ALL(dim_date),dim_date[wn]= selv-1))

return

DIVIDE(revcw,revpw,0)-1
```
15. Realisation % = 1- ([Cancellation %]+[No Show rate %])

```
16. Realisation WoW change % =
    Var selv =
    IF(HASONEFILTER(dim_date[wn]),SELECTEDVALUE(dim_date[wn]),MAX(dim_date[wn]))
    var revcw = CALCULATE([Realisation %],dim_date[wn]= selv)
    var revpw =  CALCULATE([Realisation %],FILTER(ALL(dim_date),dim_date[wn]= selv-1))

    return

    DIVIDE(revcw,revpw,0)-1
17. Revenue = SUM(fact_bookings[revenue_realized])
18. Revenue WoW change % =
    Var selv =
    IF(HASONEFILTER(dim_date[wn]),SELECTEDVALUE(dim_date[wn]),MAX(dim_date[wn]))
    var revcw = CALCULATE([Revenue],dim_date[wn]= selv)
    var revpw =  CALCULATE([Revenue],FILTER(ALL(dim_date),dim_date[wn]= selv-1))

    return

    DIVIDE(revcw,revpw,0)-1
19. RevPAR = DIVIDE([Revenue],[Total Capacity])
    Revpar WoW change % =
    Var selv =
    IF(HASONEFILTER(dim_date[wn]),SELECTEDVALUE(dim_date[wn]),MAX(dim_date[wn]))
    var revcw = CALCULATE([RevPAR],dim_date[wn]= selv)
    var revpw =  CALCULATE([RevPAR],FILTER(ALL(dim_date),dim_date[wn]= selv-1))

    return

    DIVIDE(revcw,revpw,0)-1
20. Total Bookings = COUNT(fact_bookings[booking_id])
21. Total cancelled bookings = CALCULATE([Total
    Bookings],fact_bookings[booking_status]="Cancelled")
22. Total Capacity = SUM(fact_aggregated_bookings[capacity])
23. Total Checked Out = CALCULATE([Total
    Bookings],fact_bookings[booking_status]="Checked Out")
24. Total no show bookings = CALCULATE([Total
    Bookings],fact_bookings[booking_status]="No Show")
25. Total Succesful Bookings = SUM(fact_aggregated_bookings[successful_bookings])
26. day type =

     Var wkd = WEEKDAY(dim_date[date],1)

     return
     IF(
     wkd>5,"Weekend","Weekday")
27. wn = WEEKNUM(dim_date[date])
28.
```