

OFFERS IMPLEMENTATION

Chittaranjan Saha

18th Nov 2025

Table of Contents

Database (SQL Server).....	2
General Quarries	2
Tables	5
Placeholders Table	5
Offers Table	6
Customers Table.....	10
CustomerOffers Table.....	11
PlaceholderOffers Table	13
Bits Table	14
CustomerOfferStatsDaily Table	15
Functions.....	16
fn_IsCustomerValid Function	16
fn_IsPlaceholderValid.....	17
Store Procedures.....	18
GetCustomerDetails.....	18
GetOfferDetails	20
GetPlaceholderDetails	22
InsertBit.....	24
GetBitDetails	26
GetCustomerOffer.....	28
Trigger	32
Trg_UpdateClickViewStatusDaily	32

Database (SQL Server)

```
CREATE DATABASE SampleOffersAPI;
GO
USE SampleOffersAPI;
```

Database used – Microsoft SQL Server

Local Project path - C:\Users\sahac\Desktop\Education\Codes\Others\Personal Projects\Offers Implementation-API\Database

General Quaries

```
-- Get all offers for a given customer
SELECT
    c.Cust_ID,
    c.Cust_Fed_ID,
    c.Cust_Name,
    o.Offer_ID,
    o.Offer_Name
FROM CustomerOffers co
INNER JOIN Offers o ON co.Offer_ID = o.Offer_ID
INNER JOIN Customers c ON co.Cust_Fed_ID = c.Cust_Fed_ID
WHERE c.Cust_Fed_ID = '6049382751';

-- Get customer and placeholder details
SELECT
    c.Cust_Fed_ID AS cust_fed_id,
    p.Placeholder_Guid AS placeholder_guid,
    p.Placeholder_Name AS placeholder_name,
    p.Channel_Name AS channel_name,
    p.Channel_Type AS channel_type,
    p.Placeholder_Offer_Limit AS placeholder_offer_limit,
    p.Placeholder_Meta AS placeholder_meta
FROM Customers c
INNER JOIN Placeholders p ON p.Placeholder_Guid = '55C5AA3B-AB8B-F011-9F57-64D69AE7FAE4'
WHERE c.Cust_Fed_ID = '6049382751';

-- Get default offers for a placeholder
SELECT
    o.Offer_Guid,
    o.Offer_Name,
    o.Offer_Description,
    o.Offer_Type,
    o.Offer_Priority,
    o.Offer_View_Threshold,
    o.Offer_Click_Threshold,
```

```

    o.Offer_Banner_Type,
    o.Offer_Content,
    o.Offer_Extra_Content,
    o.Offer_Meta
FROM PlaceholderOffers po
INNER JOIN Placeholders p2 ON p2.Placeholder_ID = po.Placeholder_ID
INNER JOIN Offers o ON o.Offer_ID = po.Offer_ID
WHERE p2.Placeholder_Guid = '55C5AA3B-AB8B-F011-9F57-64D69AE7FAE4'
    AND o.Offer_Type = 'Default'
    AND o.Offer_Visibility_Flag = 1;

-- Get sum of views and click counts for last three days
SELECT
    Cust_Fed_ID,
    Offer_Guid,
    SUM(View_Count) AS Views3Days,
    SUM(Click_Count) AS Clicks3Days
FROM CustomerOfferStatsDaily
WHERE Stat_Date >= CAST(DATEADD(DAY, -2, CAST(GETDATE() AS DATE)) AS DATE)
GROUP BY Cust_Fed_ID, Offer_Guid

-- Get sum of views and click counts for last three days
WITH Stats3Days AS (
    SELECT
        Cust_Fed_ID,
        Offer_Guid,
        SUM(View_Count) AS Views3Days,
        SUM(Click_Count) AS Clicks3Days
    FROM CustomerOfferStatsDaily
    WHERE Stat_Date >= CAST(DATEADD(DAY, -2, CAST(GETDATE() AS DATE)) AS DATE)
    GROUP BY Cust_Fed_ID, Offer_Guid
)

SELECT DISTINCT
    o.Offer_Guid AS offer_guid,
    o.Offer_Name AS offer_name,
    o.Offer_Description AS offer_description,
    o.Offer_Type AS offer_type,
    o.Offer_Priority AS offer_priority,
    o.Offer_View_Threshold AS offer_view_threshold,
    o.Offer_Click_Threshold AS offer_click_threshold,
    o.Offer_Banner_Type AS offer_banner_type,
    o.Offer_Content AS offer_content,
    o.Offer_Extra_Content AS offer_extra_content,
    o.Offer_Meta AS offer_meta
FROM CustomerOffers co
INNER JOIN Offers o ON co.Offer_ID = o.Offer_ID
INNER JOIN PlaceholderOffers po ON o.Offer_ID = po.Offer_ID
INNER JOIN Placeholders p2 ON po.Placeholder_ID = p2.Placeholder_ID
LEFT JOIN Stats3Days s ON co.Cust_Fed_ID = s.Cust_Fed_ID AND o.Offer_Guid = s.Offer_Guid
WHERE
    co.Cust_Fed_ID = '6049382751'
    AND p2.Placeholder_Guid = '55C5AA3B-AB8B-F011-9F57-64D69AE7FAE4'
    AND o.Offer_Visibility_Flag = 1
    AND ISNULL(s.Views3Days, 0) < o.Offer_View_Threshold
    AND ISNULL(s.Clicks3Days, 0) < o.Offer_Click_Threshold

```

```
--Outputs is a single JSON object containing alll the offers for that placeholder
SELECT
    c.Cust_Fed_ID AS cust_fed_id,
    p.Placeholder_Guid AS placeholder_guid,
    p.Placeholder_Name AS placeholder_name,
    p.Channel_Name AS channel_name,
    p.Channel_Type AS channel_type,
    p.Placeholder_Offer_Limit AS placeholder_offer_limit,
    p.Placeholder_Meta AS placeholder_meta
FROM Customers c
    INNER JOIN CustomerOffers co ON c.cust_fed_id = co.cust_fed_id
    INNER JOIN Placeholders p ON p.Placeholder_Guid = '55C5AA3B-AB8B-F011-9F57-
64D69AE7FAE4'
WHERE c.cust_fed_id = '9283746150'
FOR JSON PATH, WITHOUT_ARRAY_WRAPPER

--JSON Output of above quary after beautifying
{
    "cust_fed_id": "9283746150",
    "placeholder_guid": "55C5AA3B-AB8B-F011-9F57-64D69AE7FAE4",
    "placeholder_name": "Offer For You",
    "channel_name": "NLI",
    "channel_type": "Web",
    "placeholder_offer_limit": 2,
    "placeholder_meta": "{\"service_type\":\"Retail\", \"placeholder_description\":\"\"}"
}
```

Tables

Placeholders Table

```
CREATE TABLE Placeholders
(
    Placeholder_ID INT PRIMARY KEY IDENTITY,
    Placeholder_Guid UNIQUEIDENTIFIER NOT NULL UNIQUE DEFAULT NEWSEQUENTIALID(),
    Placeholder_Name VARCHAR(100) NOT NULL,
    Channel_Name VARCHAR(100),
    Channel_Type VARCHAR(100) CHECK (Channel_Type IN ('Web', 'Mobile', 'Other')),
    Placeholder_Offer_Limit INT CHECK (Placeholder_Offer_Limit >= 0),
    Placeholder_Creation_Date DATETIME DEFAULT GETDATE(),
    Placeholder_Meta NVARCHAR(MAX) NULL,
    Placeholder_Is_Active BIT NOT NULL DEFAULT 1
);

INSERT INTO Placeholders
(Placeholder_Name, Channel_Name, Channel_Type, Placeholder_Offer_Limit, Placeholder_Meta,
Placeholder_Is_Active)
VALUES
('Home Page Banner', 'NLI', 'Web', 15,
N'{"service_type":"Retail","placeholder_description":""}', 1),
('Hero Banner', 'iMobile', 'Mobile', 3,
N'{"service_type":"Retail","placeholder_description":""}', 1),
('What's New', 'RIB', 'Web', 8,
N'{"service_type":"Retail%","placeholder_description":""}', 1),
('Offer For You', 'NLI', 'Web', 2,
N'{"service_type":"Retail","placeholder_description":""}', 0),
('Marchent Dashboard', 'InstaBIZ', 'Mobile', 4,
N'{"service_type":"Corporate","placeholder_description":""}', 1),
('Notification', 'NLI', 'Web', 6,
N'{"service_type":"Retail","placeholder_description":""}', 1),
('FT Success', 'InstaBIZ', 'Mobile', 10,
N'{"service_type":"Corporate","placeholder_description":""}', 1),
('GST Success', 'CIB', 'Web', 7,
N'{"service_type":"Corporate","placeholder_description":""}', 1),
('Loans', 'RIB', 'Web', 12,
N'{"service_type":"Retail","placeholder_description":""}', 0),
('Rewards', 'iMobile', 'Mobile', 15,
N'{"service_type":"Retail","placeholder_description":""}', 1);

SELECT * FROM Placeholders;

-- DROP TABLE Placeholders;
```

Output –

Placeholder	Placeholder	Placeholder	Channel_Nam	Channel_Typ	Placeholder	Placeholder	Placeholder	Placeholder	Placeholder
152C5AA3B-AE	Home Page B	NLI	Web		15	07-09-2025	{"service_type": "Retail", "placeholder_description": ""}	1	
253C5AA3B-AE	Hero Banner	iMobile	Mobile		3	07-09-2025	{"service_type": "Retail", "placeholder_description": ""}	1	
354C5AA3B-AE	What's New	RIB	Web		8	07-09-2025	{"service_type": "Retail%", "placeholder_description": ""}	1	
455C5AA3B-AE	Offer For You	NLI	Web		2	07-09-2025	{"service_type": "Retail", "placeholder_description": ""}	0	
556C5AA3B-AE	Marchent Da	InstaBIZ	Mobile		4	07-09-2025	{"service_type": "Corporate", "placeholder_description": ""}	1	

Offers Table

```
CREATE TABLE Offers
(
    Offer_ID INT PRIMARY KEY IDENTITY,
    Offer_GUID UNIQUEIDENTIFIER NOT NULL UNIQUE DEFAULT NEWSEQUENTIALID(),
    Offer_Name VARCHAR(100),
    Offer_Description VARCHAR(500) NULL,
    Offer_Type VARCHAR(100),
    Offer_Start_Date DATETIME NOT NULL,
    Offer_End_Date DATETIME NOT NULL,
    Offer_Priority INT,
    Offer_View_Threshold INT,
    Offer_Click_Threshold INT,
    Offer_Banner_Type VARCHAR(100),
    Offer_Content NVARCHAR(MAX) NULL,
    Offer_Extra_Content NVARCHAR(MAX) NULL,
    Offer_Meta NVARCHAR(MAX) NULL,
    Last_Modification_Date DATETIME NOT NULL DEFAULT GETDATE(),
    Offer_Visibility_Flag BIT NOT NULL DEFAULT 1,
    CONSTRAINT CK_Offers_Date CHECK (Offer_End_Date >= Offer_Start_Date)
);

INSERT INTO Offers
(Offer_Name, Offer_Description, Offer_Type, Offer_Start_Date, Offer_End_Date,
Offer_Priority, Offer_View_Threshold, Offer_Click_Threshold, Offer_Banner_Type,
Offer_Content, Offer_Extra_Content, Offer_Meta, Offer_Visibility_Flag)
VALUES

('Pre-Approved Home Loan Offer', 'Offer Description', 'NBA', '2025-09-01', '2026-12-31', 1, 10, 3, 'Static',
N'{"redirection_URL": "https://www.example.com/offers/123", "image_URL": {"mobile": "https://cdn.example.com/images/offer_mobile.jpg", "desktop": "https://cdn.example.com/images/offer_desktop.jpg", "tablet": "https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '', N'{"product": "HL"}', 1),

('Personal Loan Asset Offer', 'Offer Description', 'NBA', '2025-09-01', '2026-12-31', 2, 10, 3, 'Static',
N'{"redirection_URL": "https://www.example.com/offers/123", "image_URL": {"mobile": "https://cdn.example.com/images/offer_mobile.jpg", "desktop": "https://cdn.example.com/images/offer_desktop.jpg", "tablet": "https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '', N'{"product": "PL"}', 1),

('Forex Offer', 'Offer Description', 'NBA', '2025-09-01', '2026-12-31', 3, 10, 3, 'Static',
N'{"redirection_URL": "https://www.example.com/offers/123", "image_URL": {"mobile": "https://cdn.example.com/images/offer_mobile.jpg", "desktop": "https://cdn.example.com/images/offer_desktop.jpg", "tablet": "https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '', N'{"product": "Forex"}', 1),

('Credit Card Upgrade Offer', 'Offer Description', 'NBA', '2025-09-01', '2026-12-31', 4, 10, 3, 'Static',
N'{"redirection_URL": "https://www.example.com/offers/123", "image_URL": {"mobile": "https://cdn.example.com/images/offer_mobile.jpg", "desktop": "https://cdn.example.com/images/offer_desktop.jpg", "tablet": "https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '', N'{"product": "CC"}', 1),

('Gold Loan Offer', 'Offer Description', 'NBA', '2025-09-01', '2026-12-31', 5, 10, 3, 'Static',
N'{"redirection_URL": "https://www.example.com/offers/123", "image_URL": {"mobile": "https://cdn.example.com/images/offer_mobile.jpg", "desktop": "https://cdn.example.com/images/offer_desktop.jpg", "tablet": "https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '', N'{"product": "GL"}', 1),
```

```
('Scan & Pay Offer','Offer Description','NBA','2025-09-01','2026-12-31',6,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"SP"}',1),  
  
('Real Time MF Investment','Offer Description','NBA','2025-09-01','2026-12-31',7,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"MF"}',1),  
  
('Pay to Contact Offer','Offer Description','NBA','2025-09-01','2026-12-31',8,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"PC"}',1),  
  
('Mutual Fund Offer','Offer Description','NBA','2025-09-01','2026-12-31',9,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"MF"}',1),  
  
('iWish Offer','Offer Description','NBA','2025-09-01','2026-12-31',10,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"iWish"}',1),  
  
('Two Wheeler Loan Offer','Offer Description','NBA','2025-09-01','2026-12-31',11,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"TWL"}',1),  
  
('Fastag Offer','Offer Description','NBA','2025-09-01','2026-12-31',12,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"Fastag"}',1),  
  
('Education Loan Offer','Offer Description','NBA','2025-09-01','2026-12-31',13,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"EL"}',1),  
  
('Electricity Billpay Offer','Offer Description','NBA','2025-09-01','2026-12-31',14,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"BBPS"}',1),  
  
('Credit Card Limit Increase Offer','Offer Description','NBA','2025-09-01','2026-12-31',15,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL":{"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"},"offer_text":"Get 20% off on your first purchase!"}', '',N'{"product":"CLI"}',1),
```

```

('Consumer EMI Offer','Offer Description','NBA','2025-09-01','2026-12-31',16,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL": {"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '',N'{"product":"EMI"}',1),
('BBPS Electricity Offer','Offer Description','NBA','2025-09-01','2026-12-31',17,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL": {"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '',N'{"product":"BBPS"}',1),
('Debit Card Upgrade Offer','Offer Description','NBA','2025-09-01','2026-12-31',18,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL": {"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '',N'{"product":"DCU"}',1),
('Auto Loan Default Offer','Offer Description','Default','2025-09-01','2026-12-31',19,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL": {"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '',N'{"product":"AL"}',1),
('Demat Default Offer','Offer Description','Default','2025-09-01','2026-12-31',20,10,3,'Static',
N'{"redirection_URL":"https://www.example.com/offers/123","image_URL": {"mobile":"https://cdn.example.com/images/offer_mobile.jpg","desktop":"https://cdn.example.com/images/offer_desktop.jpg","tablet":"https://cdn.example.com/images/offer_tablet.jpg"}, "offer_text": "Get 20% off on your first purchase!"}', '',N'{"product":"Demat"}',1);

SELECT * FROM Offers;

-- Update default offers threshold
UPDATE Offers
SET Offer_Visibility_Flag = 0
WHERE Offer_ID = 19

-- Validate JSON field
SELECT ISJSON(Offer_Content)
FROM Offers
WHERE Offer_ID = 19;

-- Update specific Offer Content
UPDATE Offers
SET Offer_Content = N'{'
    "redirection_URL": "https://www.chittaranjansaha.com",
    "image_URL": {
        "mobile": "https://raw.githubusercontent.com/git-chittaranjan/offers-implementation-api/main/Frontend/Contents/Desktop/Demat_Default_Offer.gif",
        "desktop": "https://raw.githubusercontent.com/git-chittaranjan/offers-implementation-api/main/Frontend/Contents/Desktop/Demat_Default_Offer.gif",
        "tablet": "https://raw.githubusercontent.com/git-chittaranjan/offers-implementation-api/main/Frontend/Contents/Desktop/Demat_Default_Offer.gif"
    },
    "offer_text": "Demat_Default_Offer.gif"
}'
WHERE Offer_ID = 20;

-- DROP TABLE Offers;

```

Output –

Offer	Offer_GUI	Offer_Nam	Offer_Desc	Offer_Type	Offer_Start	Offer_End	Offer_Prio	Offer_View	Offer_Click	Offer_Ban	Offer_Content	Offer_Extra	Offer_Met	Last_Modifi	Offer_Visib
1	CBD47CC0	Pre-Approv	Offer Descr	NBA	01-09-2025	31-12-2026	1	10	3	Static	{"redirection_URL": "https://www.product.com/offer1"}	01-09-2025	1		
2	5D108C19	Personal Lc	Offer Descr	NBA	01-09-2025	31-12-2026	2	10	3	Static	{"redirection_URL": "https://www.product.com/offer2"}	01-09-2025	1		
3	F6E26247-	Forex Offer	Offer Descr	NBA	01-09-2025	31-12-2026	3	10	3	Static	{"redirection_URL": "https://www.product.com/offer3"}	01-09-2025	1		
4	F7E26247-	Credit Card	Offer Descr	NBA	01-09-2025	31-12-2026	4	10	3	Static	{"redirection_URL": "https://www.product.com/offer4"}	01-09-2025	1		
5	F8E26247-	Gold Loan	Offer Descr	NBA	01-09-2025	31-12-2026	5	10	3	Static	{"redirection_URL": "https://www.product.com/offer5"}	01-09-2025	1		
6	F9E26247-	Scan & Pay	Offer Descr	NBA	01-09-2025	31-12-2026	6	10	3	Static	{"redirection_URL": "https://www.product.com/offer6"}	01-09-2025	1		
7	FAE26247-	Real Time	Offer Descr	NBA	01-09-2025	31-12-2026	7	10	3	Static	{"redirection_URL": "https://www.product.com/offer7"}	01-09-2025	1		
8	FBE26247-	Pay to Cont	Offer Descr	NBA	01-09-2025	31-12-2026	8	10	3	Static	{"redirection_URL": "https://www.product.com/offer8"}	01-09-2025	1		
9	FCE26247-	Mutual Fur	Offer Descr	NBA	01-09-2025	31-12-2026	9	10	3	Static	{"redirection_URL": "https://www.product.com/offer9"}	01-09-2025	1		
10	FDE26247-	iWish Offe	Offer Descr	NBA	01-09-2025	31-12-2026	10	10	3	Static	{"redirection_URL": "https://www.product.com/offer10"}	01-09-2025	1		
11	FEE26247-	Two Whee	Offer Descr	NBA	01-09-2025	31-12-2026	11	10	3	Static	{"redirection_URL": "https://www.product.com/offer11"}	01-09-2025	1		
12	FFE26247-	Fastag Offe	Offer Descr	NBA	01-09-2025	31-12-2026	12	10	3	Static	{"redirection_URL": "https://www.product.com/offer12"}	01-09-2025	1		
13	00E36247-	Education	Offer Descr	NBA	01-09-2025	31-12-2026	13	10	3	Static	{"redirection_URL": "https://www.product.com/offer13"}	01-09-2025	1		
14	1ECB7B32-	Electricity	Offer Descr	NBA	01-09-2025	31-12-2026	14	10	3	Static	{"redirection_URL": "https://www.product.com/offer14"}	01-09-2025	1		
15	1FCB7B32-	Credit Card	Offer Descr	NBA	01-09-2025	31-12-2026	15	10	3	Static	{"redirection_URL": "https://www.product.com/offer15"}	01-09-2025	1		
16	20CB7B32-	Consumer	Offer Descr	NBA	01-09-2025	31-12-2026	16	10	3	Static	{"redirection_URL": "https://www.product.com/offer16"}	01-09-2025	1		
17	21CB7B32-	BBPS Elect	Offer Descr	NBA	01-09-2025	31-12-2026	17	10	3	Static	{"redirection_URL": "https://www.product.com/offer17"}	01-09-2025	1		
18	22CB7B32-	Debit Card	Offer Descr	NBA	01-09-2025	31-12-2026	18	10	3	Static	{"redirection_URL": "https://www.product.com/offer18"}	01-09-2025	1		
19	23CB7B32-	Auto Loan	Offer Descr	Default	01-09-2025	31-12-2026	19	[NULL]	[NULL]	Static	{"redirection_URL": "https://www.product.com/offer19"}	01-09-2025	0		
20	24CB7B32-	Demat Def	Offer Descr	Default	01-09-2025	31-12-2026	20	[NULL]	[NULL]	Static	{"redirection_URL": "https://www.product.com/offer20"}	01-09-2025	1		

Notes –

- Check for **Offer_Type**, Default offers will not have suppression
- Check for **Offer_Start_Date**, **Offer_End_Date** for offers to return in APIs
- Offer_Priority** for ranking offers
- Offer_View_Threshold** and **Offer_Click_Threshold** are for suppression
- Offer_content** contains the offers content o be displayed
- Offer_Visibility_Flag** is to check active offers

Customers Table

```
CREATE TABLE Customers
(
    Cust_ID INT IDENTITY,
    Cust_Fed_ID VARCHAR(100) PRIMARY KEY,
    Cust_Name VARCHAR(100),
    Cust_Data NVARCHAR(MAX),
    Cust_Entry_Date DATETIME NOT NULL DEFAULT GETDATE(),
    Cust_Is_Active BIT NOT NULL DEFAULT 1
);

INSERT INTO Customers (Cust_Fed_ID, Cust_Name, Cust_Data)
VALUES
('4839201746', 'Rahul Sharma', N'{"age": 32, "city": "Mumbai", "gender": "Male"}'),
('9283746150', 'Ananya Gupta', N'{"age": 28, "city": "Delhi", "gender": "Female"}'),
('7152903846', 'Vikram Singh', N'{"age": 40, "city": "Bengaluru", "gender": "Male"}'),
('6049382751', 'Priya Nair', N'{"age": 35, "city": "Chennai", "gender": "Female"}'),
('8391046275', 'Arjun Mehta', N'{"age": 30, "city": "Kolkata", "gender": "Male"}');

SELECT * FROM Customers ORDER BY Cust_ID;

-- DROP TABLE Customers;

UPDATE Customers
    SET Cust_Is_Active = 0
WHERE Cust_Fed_ID = '8391046275'
```

Output –

Cust_ID	Cust_Fed_ID	Cust_Name	Cust_Data	Cust_Entry_D	Cust_Is_Activ
1	4839201746	Rahul Sharma	{"age": 32, "city": "Mu	50:43.0	1
2	9283746150	Ananya Gupta	{"age": 28, "city": "Del	50:43.0	1
3	7152903846	Vikram Singh	{"age": 40, "city": "Ber	50:43.0	1
4	6049382751	Priya Nair	{"age": 35, "city": "Che	50:43.0	1
5	8391046275	Arjun Mehta	{"age": 30, "city": "Kol	50:43.0	0

Notes –

- Check for Customer_Is_Active status

CustomerOffers Table

```
CREATE TABLE CustomerOffers
(
    Cust_Fed_ID VARCHAR(100) NOT NULL,
    Offer_ID INT NOT NULL,
    Offer_Guid UNIQUEIDENTIFIER NOT NULL,
    Assigned_Date DATETIME DEFAULT GETDATE(),
    PRIMARY KEY (Cust_Fed_ID, Offer_ID),
    FOREIGN KEY (Cust_Fed_ID) REFERENCES Customers(Cust_Fed_ID),
    FOREIGN KEY (Offer_ID) REFERENCES Offers(Offer_ID)
);

INSERT INTO CustomerOffers (Cust_Fed_ID, Offer_ID, Assigned_Date)
VALUES
('4839201746', 1, '2025-08-01'), ('4839201746', 3, '2025-08-05'),
('4839201746', 5, '2025-08-06'), ('4839201746', 7, '2025-08-12'),
('4839201746', 9, '2025-08-19'), ('4839201746', 11, '2025-08-20'),
('4839201746', 13, '2025-08-20'), ('4839201746', 15, '2025-08-20'),
('4839201746', 17, '2025-08-20'), ('9283746150', 2, '2025-08-03'),
('9283746150', 4, '2025-08-14'), ('9283746150', 6, '2025-08-14'),
('9283746150', 8, '2025-08-14'), ('9283746150', 10, '2025-08-14'),
('9283746150', 12, '2025-08-14'), ('9283746150', 14, '2025-08-14'),
('9283746150', 16, '2025-08-14'), ('9283746150', 18, '2025-08-14'),
('7152903846', 3, '2025-08-07'), ('7152903846', 6, '2025-08-08'),
('7152903846', 9, '2025-08-11'), ('7152903846', 12, '2025-08-11'),
('7152903846', 15, '2025-08-11'), ('7152903846', 18, '2025-08-11'),
('6049382751', 1, '2025-08-04'), ('6049382751', 3, '2025-08-10'),
('6049382751', 5, '2025-08-16'), ('6049382751', 7, '2025-08-18'),
('6049382751', 9, '2025-08-18'), ('6049382751', 10, '2025-08-18'),
('6049382751', 12, '2025-08-18'), ('6049382751', 14, '2025-08-18'),
('6049382751', 16, '2025-08-18'), ('6049382751', 18, '2025-08-18'),
('8391046275', 1, '2025-08-13'), ('8391046275', 2, '2025-08-15'),
('8391046275', 3, '2025-08-17'), ('8391046275', 4, '2025-08-17'),
('8391046275', 5, '2025-08-17'), ('8391046275', 6, '2025-08-17'),
('8391046275', 7, '2025-08-17'), ('8391046275', 8, '2025-08-17'),
('8391046275', 9, '2025-08-17'), ('8391046275', 10, '2025-08-17'),
('8391046275', 11, '2025-08-17'), ('8391046275', 12, '2025-08-17'),
('8391046275', 13, '2025-08-17'), ('8391046275', 14, '2025-08-17'),
('8391046275', 15, '2025-08-17'), ('8391046275', 16, '2025-08-17'),
('8391046275', 17, '2025-08-17'), ('8391046275', 18, '2025-08-17');

SELECT * FROM CustomerOffers;

SELECT
    c.Cust_ID,
    c.Cust_Fed_ID,
    c.Cust_Name,
    o.Offer_ID,
    o.Offer_GUID,
    o.Offer_Name
FROM CustomerOffers co
INNER JOIN Offers o ON co.Offer_ID = o.Offer_ID
INNER JOIN Customers c ON co.Cust_Fed_ID = c.Cust_Fed_ID
WHERE c.Cust_Fed_ID = '9283746150';

-- DROP TABLE CustomerOffers;
```

Output –

Cust_ID	Cust_Fed_ID	Cust_Name	Offer_ID	Offer_GUID	Offer_Name
2	9283746150	Ananya Gupta	2	5D108C19-54	Personal Loan Asset Offer
2	9283746150	Ananya Gupta	4	F7E26247-558	Credit Card Upgrade Offer
2	9283746150	Ananya Gupta	6	F9E26247-558	Scan & Pay Offer
2	9283746150	Ananya Gupta	8	FBE26247-558	Pay to Contact Offer
2	9283746150	Ananya Gupta	10	FDE26247-558	iWish Offer
2	9283746150	Ananya Gupta	12	FFE26247-558	Fastag Offer
2	9283746150	Ananya Gupta	14	1ECB7B32-56	Electricity Billpay Offer
2	9283746150	Ananya Gupta	16	20CB7B32-56	Consumer EMI Offer
2	9283746150	Ananya Gupta	18	22CB7B32-56	Debit Card Upgrade Offer

Cust_Fed_ID	Offer_ID	Assigned_Date
4839201746	1	01-08-2025
4839201746	3	05-08-2025
4839201746	5	06-08-2025
4839201746	7	12-08-2025
4839201746	9	19-08-2025
4839201746	11	20-08-2025
4839201746	13	20-08-2025
4839201746	15	20-08-2025
4839201746	17	20-08-2025
6049382751	1	04-08-2025
6049382751	3	10-08-2025
6049382751	5	16-08-2025
6049382751	7	18-08-2025
6049382751	9	18-08-2025
6049382751	10	18-08-2025
6049382751	12	18-08-2025
6049382751	14	18-08-2025
6049382751	16	18-08-2025
6049382751	18	18-08-2025
7152903846	3	07-08-2025
7152903846	6	08-08-2025
7152903846	9	11-08-2025
7152903846	12	11-08-2025
7152903846	15	11-08-2025
7152903846	18	11-08-2025
8391046275	1	13-08-2025
8391046275	2	15-08-2025
8391046275	3	17-08-2025
8391046275	4	17-08-2025
8391046275	5	17-08-2025
8391046275	6	17-08-2025
8391046275	7	17-08-2025
8391046275	8	17-08-2025
8391046275	9	17-08-2025
8391046275	10	17-08-2025
8391046275	11	17-08-2025
8391046275	12	17-08-2025
8391046275	13	17-08-2025
8391046275	14	17-08-2025
8391046275	15	17-08-2025
8391046275	16	17-08-2025
8391046275	17	17-08-2025
8391046275	18	17-08-2025
9283746150	2	03-08-2025
9283746150	4	14-08-2025
9283746150	6	14-08-2025
9283746150	8	14-08-2025
9283746150	10	14-08-2025
9283746150	12	14-08-2025
9283746150	14	14-08-2025
9283746150	16	14-08-2025
9283746150	18	14-08-2025

PlaceholderOffers Table

```

CREATE TABLE PlaceholderOffers
(
    Offer_ID INT NOT NULL,
    Placeholder_ID INT NOT NULL,
    PRIMARY KEY (Offer_ID, Placeholder_ID),
    FOREIGN KEY (Offer_ID) REFERENCES Offers(Offer_ID),
    FOREIGN KEY (Placeholder_ID) REFERENCES Placeholders(Placeholder_ID)
);

INSERT INTO PlaceholderOffers (Placeholder_ID, Offer_ID) VALUES
(1, 1), (1, 2), (1, 3), (1, 4), (1, 5), (1, 6), (1, 7), (1, 8), (1, 9), (1, 10),
(1, 11), (1, 12), (1, 13), (1, 14), (1, 15), (1, 16), (1, 17), (1, 18), (1, 19), (1, 20),
(2, 3), (2, 6), (2, 9), (2, 12), (2, 15), (2, 18), (2, 19), (2, 20), (3, 1), (3, 2),
(3, 3), (3, 4), (3, 5), (3, 6), (3, 7), (3, 8), (3, 9), (3, 10), (3, 15), (3, 16),
(3, 17), (3, 18), (4, 2), (4, 4), (4, 6), (4, 8), (4, 10), (4, 12), (4, 14), (4, 16),
(4, 18), (4, 20), (5, 1), (5, 3), (5, 5), (5, 7), (5, 9), (5, 11), (5, 13), (5, 15),
(5, 17), (5, 19);

SELECT * FROM PlaceholderOffers;

SELECT
    p.Placeholder_ID,
    p.Placeholder_Guid,
    p.Placeholder_Name,
    p.Channel_Name,
    o.Offer_ID,
    o.Offer_Name
FROM PlaceholderOffers po
INNER JOIN Placeholders p ON po.Placeholder_ID = p.Placeholder_ID
INNER JOIN Offers o ON po.Offer_ID = o.Offer_ID
WHERE p.Placeholder_ID = 5;

-- DROP TABLE PlaceholderOffers;

```

Output –

Placeholder_ID	Placeholder_Guid	Placeholder_Name	Channel_Name	Offer_ID	Offer_Name
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	1	Pre-Approved Home Loan Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	3	Forex Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	5	Gold Loan Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	7	Real Time MF Investment
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	9	Mutual Fund Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	11	Two Wheeler Loan Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	13	Education Loan Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	15	Credit Card Limit Increase Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	17	BBPS Electricity Offer
5	56C5AA3B-AB8B-F0	Marchent Dashboard	InstaBIZ	19	Auto Loan Default Offer

Bits Table

```

CREATE TABLE Bits
(
    Bit_ID INT PRIMARY KEY IDENTITY,
    Bit_Guid UNIQUEIDENTIFIER NOT NULL UNIQUE DEFAULT NEWSEQUENTIALID(),
    Cust_Fed_ID VARCHAR(100),
    Offer_Guid UNIQUEIDENTIFIER,
    Offer_Name VARCHAR(100),
    Bit_Type VARCHAR(100) NOT NULL CHECK (Bit_Type IN ('VIEW','CLICK')),
    Channel_Name VARCHAR(100),
    Placeholder_Guid UNIQUEIDENTIFIER,
    Placeholder_Name VARCHAR(100),
    Action_Timestamp DATETIME2 DEFAULT SYSDATETIME()
);

INSERT INTO Bits (Cust_Fed_ID, Offer_Guid, Offer_Name, Bit_Type, Channel_Name,
Placeholder_Guid, Placeholder_Name)
VALUES
('4839201746', 'CBDA7CC0-5387-F011-9F57-64D69AE7FAE4', 'Pre-Approved Home Loan Offer',
'CLICK', 'RIB', '54C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'What''s New'),
('4839201746', 'CBDA7CC0-5387-F011-9F57-64D69AE7FAE4', 'Pre-Approved Home Loan Offer',
'VIEW', 'iMobile', '53C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Hero Banner'),
('4839201746', 'CBDA7CC0-5387-F011-9F57-64D69AE7FAE4', 'Pre-Approved Home Loan Offer',
'CLICK', 'RIB', '54C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'What''s New'),
('4839201746', 'CBDA7CC0-5387-F011-9F57-64D69AE7FAE4', 'Pre-Approved Home Loan Offer',
'VIEW', 'NLI', '52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner'),
('6049382751', 'F8E26247-5587-F011-9F57-64D69AE7FAE4', 'Gold Loan Offer', 'CLICK', 'NLI',
'52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner'),
('6049382751', 'F8E26247-5587-F011-9F57-64D69AE7FAE4', 'Gold Loan Offer', 'VIEW', 'NLI',
'52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner'),
('6049382751', 'F8E26247-5587-F011-9F57-64D69AE7FAE4', 'Gold Loan Offer', 'VIEW', 'NLI',
'52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner'),
('6049382751', 'F8E26247-5587-F011-9F57-64D69AE7FAE4', 'Gold Loan Offer', 'NLI',
'52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner'),
('6049382751', 'F8E26247-5587-F011-9F57-64D69AE7FAE4', 'Gold Loan Offer', 'CLICK', 'NLI',
'52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner'),
('6049382751', 'F8E26247-5587-F011-9F57-64D69AE7FAE4', 'Gold Loan Offer', 'VIEW', 'NLI',
'52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Home Page Banner');

INSERT INTO Bits (Cust_Fed_ID, Offer_Guid, Offer_Name, Bit_Type, Channel_Name,
Placeholder_Guid, Placeholder_Name) VALUES
('6049382751', 'F6E26247-5587-F011-9F57-64D69AE7FAE4', 'Forex Offer', 'VIEW', 'InstaBIZ',
'56C5AA3B-AB8B-F011-9F57-64D69AE7FAE4', 'Marchent Dashboard')

SELECT * FROM Bits
-- DROP TABLE Bits

```

Output –

Bit_ID	Bit_Guid	Cust_Fed_ID	Offer_Guid	Offer_Name	Bit_Type	Channel_Name	Placeholder_Guid	Placeholder_Name	Action_Timestamp
1	92060F14-820	4839201746	CBDA7CC0-53	Pre-Approved Home Loa	CLICK	RIB	54C5AA3B-AE	What's New	13-11-2025
2	93060F14-820	4839201746	CBDA7CC0-53	Pre-Approved Home Loa	VIEW	iMobile	53C5AA3B-AE	Hero Banner	13-11-2025
3	94060F14-820	4839201746	CBDA7CC0-53	Pre-Approved Home Loa	CLICK	RIB	54C5AA3B-AE	What's New	13-11-2025
4	95060F14-820	4839201746	CBDA7CC0-53	Pre-Approved Home Loa	VIEW	NLI	52C5AA3B-AE	Home Page Banner	13-11-2025
5	96060F14-820	6049382751	F8E26247-558	Gold Loan Offer	CLICK	NLI	52C5AA3B-AE	Home Page Banner	13-11-2025
6	97060F14-820	6049382751	F8E26247-558	Gold Loan Offer	VIEW	NLI	52C5AA3B-AE	Home Page Banner	13-11-2025
7	98060F14-820	6049382751	F8E26247-558	Gold Loan Offer	VIEW	NLI	52C5AA3B-AE	Home Page Banner	13-11-2025
8	99060F14-820	6049382751	F8E26247-558	Gold Loan Offer	CLICK	NLI	52C5AA3B-AE	Home Page Banner	13-11-2025
9	9A060F14-82	6049382751	F8E26247-558	Gold Loan Offer	VIEW	NLI	52C5AA3B-AE	Home Page Banner	13-11-2025

CustomerOfferStatsDaily Table

```
CREATE TABLE CustomerOfferStatsDaily (
    Stat_Date DATE NOT NULL,
    Cust_Fed_ID VARCHAR(100) NOT NULL,
    Offer_Guid UNIQUEIDENTIFIER NOT NULL,
    View_Count INT DEFAULT 0,
    Click_Count INT DEFAULT 0,
    CONSTRAINT PK_CustomerOfferStatsDaily
        PRIMARY KEY (Stat_Date, Cust_Fed_ID, Offer_Guid)
);

--DROP TABLE CustomerOfferStatsDaily

SELECT * FROM CustomerOfferStatsDaily
```

Output –

Stat_Date	Cust_Fed_ID	Offer_Guid	View_Count	Click_Count
22-09-2025	4839201746	CBDA7CC0-53	2	3
22-09-2025	6049382751	CBDA7CC0-53	0	2
22-09-2025	6049382751	F6E26247-558	6	0
22-09-2025	6049382751	F8E26247-558	3	2

The table **CustomerOfferStatsDaily** stores **daily aggregated statistics** of how many times each customer **viewed** or **clicked** a particular offer. The trigger **trg_UpdateClickViewStatusDaily**, which runs **after new records are inserted into the Bits table**, automatically:

1. **Groups the newly inserted Bits records** by date, customer, and offer.
2. Calculates **how many VIEW events** and **how many CLICK events** happened.
3. Updates the **CustomerOfferStatsDaily** table:
 - o If a record for that date + customer + offer already exists → **add** (increment) the view and click counts.
 - o If not → **insert a new daily record** with the counts.
 - o **MERGE** operation to either update existing daily totals or insert a new record if none exists yet.

Functions

fn_IsCustomerValid Function

```
CREATE OR ALTER FUNCTION fn_IsCustomerValid
(
    @Cust_Fed_ID VARCHAR(100)
)
RETURNS INT
AS
BEGIN
    DECLARE @Result INT = 0;

    -- EXISTS: checks if subquery returns at least one row (TRUE if yes, FALSE if no)
    -- IF EXISTS: run BEGIN...END only if matching rows exist, else go to ELSE
    IF EXISTS (SELECT 1 FROM Customers WHERE Cust_Fed_ID = @Cust_Fed_ID)
        BEGIN
            -- Customer exists
            IF EXISTS (
                SELECT 1
                FROM Customers
                WHERE Cust_Fed_ID = @Cust_Fed_ID
                    AND Cust_Is_Active = 1
            )
                BEGIN
                    SET @Result = 1; -- Exists & Active
                END
            ELSE
                BEGIN
                    SET @Result = 0; -- Exists but Inactive
                END
        END
    ELSE
        BEGIN
            SET @Result = -1; -- Not Found
        END

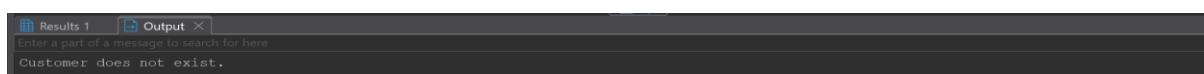
    RETURN @Result;
END;

-- ===== Function Execution =====
DECLARE @Status INT;
SET @Status = dbo.fn_IsCustomerValid('715290384');

SELECT @Status AS CustomerStatus;

IF @Status = 1
    PRINT 'Customer exists and is active.';
ELSE IF @Status = 0
    PRINT 'Customer exists but is inactive.';
ELSE
    PRINT 'Customer does not exist.';
```

Output for Cust_Fed_ID ('715290384') –



CustomerStatus	Message
-1	Customer does not exist.

fn_IsPlaceholderValid

```
CREATE OR ALTER FUNCTION fn_IsPlaceholderValid
(
    @Placeholder_Guid UNIQUEIDENTIFIER
)
RETURNS INT
AS
BEGIN
    DECLARE @Result INT = 0;

    -- EXISTS: checks if subquery returns at least one row (TRUE if yes, FALSE if no)
    -- IF EXISTS: run BEGIN...END only if matching rows exist, else go to ELSE
    IF EXISTS (SELECT 1 FROM Placeholders WHERE Placeholder_Guid = @Placeholder_Guid)
        BEGIN
            -- Placeholder exists
            IF EXISTS (
                SELECT 1
                FROM Placeholders
                WHERE Placeholder_Guid = @Placeholder_Guid
                    AND Placeholder_Is_Active = 1
            )
                BEGIN
                    SET @Result = 1; -- Exists & Active
                END
            ELSE
                BEGIN
                    SET @Result = 0; -- Exists but Inactive
                END
        END
    ELSE
        BEGIN
            SET @Result = -1; -- Not Found
        END

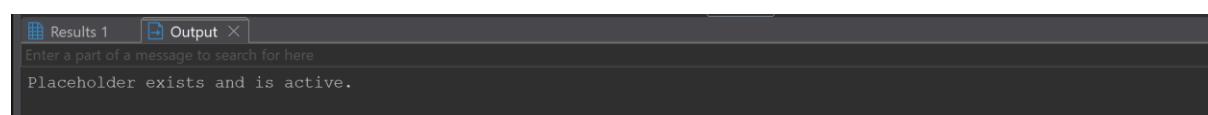
    RETURN @Result;
END;

-- ===== Function Execution =====
DECLARE @Status INT;
SET @Status = dbo.fn_IsPlaceholderValid('54C5AA3B-AB8B-F011-9F57-64D69AE7FAE4');

SELECT @Status AS PlaceholderStatus;

IF @Status = 1
    PRINT 'Placeholder exists and is active.';
ELSE IF @Status = 2
    PRINT 'Placeholder exists but is inactive.';
ELSE
    PRINT 'Placeholder does not exist.';
```

Output for Placeholder_Guid ('54C5AA3B-AB8B-F011-9F57-64D69AE7FAE4') –



The screenshot shows the SSMS output window with two tabs: 'Results 1' and 'Output'. The 'Output' tab is selected. It contains a search bar and the following text:
Placeholder exists and is active.

Store Procedures

GetCustomerDetails

```
CREATE PROCEDURE GetCustomerDetails
    @Cust_Fed_ID VARCHAR(100),
    @StatusCode INT OUTPUT -- 0 = Success, 1 = Customer Not Found, -1 = Error
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        -- Check if customer exists
        IF EXISTS (
            SELECT 1 FROM Customers
            WHERE Cust_Fed_ID = @Cust_Fed_ID
        )
        BEGIN
            SET @StatusCode = 0; -- Success

            SELECT
                Cust_Fed_ID,
                Cust_Name,
                Cust_Data,
                Cust_Entry_Date,
                Cust_Is_Active,
                'Success' AS Message
            FROM Customers
            WHERE Cust_Fed_ID = @Cust_Fed_ID;
        END
        ELSE
        BEGIN
            SET @StatusCode = 1; -- Not Found

            SELECT
                @Cust_Fed_ID AS Cust_Fed_ID,
                CAST(NULL AS NVARCHAR(100)) AS Cust_Name,
                CAST(NULL AS NVARCHAR(MAX)) AS Cust_Data,
                CAST(NULL AS DATETIME) AS Cust_Entry_Date,
                CAST(NULL AS BIT) AS Cust_Is_Active,
                'No customer found with the given Customer Fed ID' AS Message;
        END
    END TRY

    BEGIN CATCH
        SET @StatusCode = -1; -- Error

        SELECT
            @Cust_Fed_ID AS Cust_Fed_ID,
            CAST(NULL AS NVARCHAR(100)) AS Cust_Name,
            CAST(NULL AS NVARCHAR(MAX)) AS Cust_Data,
            CAST(NULL AS DATETIME) AS Cust_Entry_Date,
            CAST(NULL AS BIT) AS Cust_Is_Active,
            N'Error: ' + ERROR_MESSAGE() AS Message; -- fixed
    END CATCH

    RETURN @StatusCode; -- optional but cleaner
END;
```

```
-- ===== SP Execution =====

DECLARE @Status INT;

EXEC GetCustomerDetails
    @Cust_Fed_ID = '4839201746',
    @StatusCode = @Status OUTPUT;

SELECT @Status AS StatusCode;
```

Output –

1. If Cust_Fed_ID exists (@Status = 0)

Grid	AZ Cust_Fed_ID	AZ Cust_Name	AZ Cust_Data	⌚ Cust_Entry_Date	I23 Cust_Is_Active	AZ Message
ext	1 4839201746	Rahul Sharma	{"age": 32, "city": "Mumbai", "gender": "Male"}	2025-09-02 12:50:43.043	1	Success

1. If Cust_Fed_ID does not exists (@Status = 1)

Grid	AZ Cust_Fed_ID	AZ Cust_Name	AZ Cust_Data	⌚ Cust_Entry_Date	I23 Cust_Is_Active	AZ Message
ext	1 4839201745	[NULL]	[NULL]	[NULL]	[NULL]	No customer found with the given Customer Fed ID

GetOfferDetails

```
CREATE OR ALTER PROCEDURE GetOfferDetails
    @Offer_Guid UNIQUEIDENTIFIER,
    @StatusCode INT OUTPUT -- 0 = Success, 1 = Offer Not Found, -1 = Error
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        -- Check if offer exists
        IF EXISTS (
            SELECT 1 FROM Offers
            WHERE Offer_Guid = @Offer_Guid
        )
        BEGIN
            SET @StatusCode = 0; -- Success

            SELECT
                Offer_GUID,
                Offer_Name,
                Offer_Description,
                Offer_Type,
                Offer_Start_Date,
                Offer_End_Date,
                Offer_Priority,
                Offer_View_Threshold,
                Offer_Click_Threshold,
                Offer_Banner_Type,
                Offer_Content,
                Offer_Extra_Content,
                Offer_Meta,
                Last_Modification_Date,
                Offer_Visibility_Flag,
                'Success' AS Message
            FROM Offers
            WHERE Offer_Guid = @Offer_Guid;
        END
        ELSE
        BEGIN
            SET @StatusCode = 1; -- Not Found

            SELECT
                @Offer_Guid AS Offer_GUID,
                CAST(NULL AS VARCHAR(100)) AS Offer_Name,
                CAST(NULL AS VARCHAR(500)) AS Offer_Description,
                CAST(NULL AS VARCHAR(100)) AS Offer_Type,
                CAST(NULL AS DATETIME) AS Offer_Start_Date,
                CAST(NULL AS DATETIME) AS Offer_End_Date,
                CAST(NULL AS INT) AS Offer_Priority,
                CAST(NULL AS INT) AS Offer_View_Threshold,
                CAST(NULL AS INT) AS Offer_Click_Threshold,
                CAST(NULL AS VARCHAR(100)) AS Offer_Banner_Type,
                CAST(NULL AS NVARCHAR(MAX)) AS Offer_Content,
                CAST(NULL AS NVARCHAR(MAX)) AS Offer_Extra_Content,
                CAST(NULL AS NVARCHAR(MAX)) AS Offer_Meta,
                CAST(NULL AS DATETIME) AS Last_Modification_Date,
                CAST(NULL AS BIT) AS Offer_Visibility_Flag,
                'No offer found with the given Offer ID.' AS Message;
        END
    END TRY
    BEGIN CATCH
```

```
SET @StatusCode = -1; -- Error

SELECT
    @Offer_Guid AS Offer_GUID,
    CAST(NULL AS VARCHAR(100)) AS Offer_Name,
    CAST(NULL AS VARCHAR(500)) AS Offer_Description,
    CAST(NULL AS VARCHAR(100)) AS Offer_Type,
    CAST(NULL AS DATETIME) AS Offer_Start_Date,
    CAST(NULL AS DATETIME) AS Offer_End_Date,
    CAST(NULL AS INT) AS Offer_Priority,
    CAST(NULL AS INT) AS Offer_View_Threshold,
    CAST(NULL AS INT) AS Offer_Click_Threshold,
    CAST(NULL AS VARCHAR(100)) AS Offer_Banner_Type,
    CAST(NULL AS NVARCHAR(MAX)) AS Offer_Content,
    CAST(NULL AS NVARCHAR(MAX)) AS Offer_Extra_Content,
    CAST(NULL AS NVARCHAR(MAX)) AS Offer_Meta,
    CAST(NULL AS DATETIME) AS Last_Modification_Date,
    CAST(NULL AS BIT) AS Offer_Visibility_Flag,
    N'Error: ' + ERROR_MESSAGE() AS Message;

END CATCH
END;
```

-- ===== SP Execution =====

```
DECLARE @Status INT;

EXEC GetOfferDetails
    @Offer_Guid = 'CBDA7CC0-5387-F011-9F57-64D69AE7FAE4',
    @StatusCode = @Status OUTPUT;

SELECT @Status AS StatusCode;
```

Output for valid offer guid –

Offer_GUID	Offer_Name	Offer_Descripti	Offer_Type	Offer_Start_Dat	Offer_End_Dat	Offer_Priority	View_Thres	Click_Thres	Offer_Banner_Ty	Offer_Content	Offer_Extra_Con	Offer_Meta	Modification_T	Offer_Visibility	Message
CBDAA7CC0-53	Pre-Approved	Offer Descrip	NBA	00:00:0	00:00:0	1	10	3	Static	{"redirection_URL": "https://product.com/H		20:24:0	1	Success	

Output for invalid offer guid –

GetPlaceholderDetails

```
CREATE OR ALTER PROCEDURE GetPlaceholderDetails
    @Placeholder_Guid UNIQUEIDENTIFIER,
    @StatusCode INT OUTPUT -- 0 = Success, 1 = Not Found, -1 = Error
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        -- Check if placeholder exists
        IF EXISTS (
            SELECT 1 FROM Placeholders
            WHERE Placeholder_Guid = @Placeholder_Guid
        )
        BEGIN
            SET @StatusCode = 0; -- Success

            SELECT
                Placeholder_Guid,
                Placeholder_Name,
                Channel_Name,
                Channel_Type,
                Placeholder_Offer_Limit,
                Placeholder_Creation_Date,
                Placeholder_Meta,
                Placeholder_Is_Active,
                'Success' AS Message
            FROM Placeholders
            WHERE Placeholder_Guid = @Placeholder_Guid;
        END
        ELSE
        BEGIN
            SET @StatusCode = 1; -- Not Found

            SELECT
                @Placeholder_Guid AS Placeholder_Guid,
                CAST(NULL AS NVARCHAR(100)) AS Placeholder_Name,
                CAST(NULL AS NVARCHAR(100)) AS Channel_Name,
                CAST(NULL AS NVARCHAR(100)) AS Channel_Type,
                CAST(NULL AS INT) AS Placeholder_Offer_Limit,
                CAST(NULL AS DATETIME) AS Placeholder_Creation_Date,
                CAST(NULL AS NVARCHAR(MAX)) AS Placeholder_Meta,
                CAST(NULL AS BIT) AS Placeholder_Is_Active,
                'No Placeholder found with the given Placeholder ID.' AS Message;
        END
    END TRY

    BEGIN CATCH
        SET @StatusCode = -1; -- Error

        SELECT
            @Placeholder_Guid AS Placeholder_Guid,
            CAST(NULL AS NVARCHAR(100)) AS Placeholder_Name,
            CAST(NULL AS NVARCHAR(100)) AS Channel_Name,
            CAST(NULL AS NVARCHAR(100)) AS Channel_Type,
            CAST(NULL AS INT) AS Placeholder_Offer_Limit,
            CAST(NULL AS DATETIME) AS Placeholder_Creation_Date,
            CAST(NULL AS NVARCHAR(MAX)) AS Placeholder_Meta,
            CAST(NULL AS BIT) AS Placeholder_Is_Active,
            N'Error: ' + ERROR_MESSAGE() AS Message;
    END CATCH
END;
```

```
-- ===== SP Execution =====

DECLARE @Status INT;

EXEC GetPlaceholderDetails
    @Placeholder_Guid = '54C5AA3B-AB8B-F011-9F57-64D69AE7FAE4',
    @StatusCode = @Status OUTPUT;

SELECT @Status AS StatusCode;
```

Output with valid placeholder guid –

Placeholder_Guid	Placeholder_Name	Channel_Name	Channel_Type	Placeholder_Offer	Placeholder_Created	Placeholder_Me	Placeholder_Is_A	Message
54C5AA3B-AE	What's New	RIB	Web	8	56:41.5 {"service_typ		1	Success

Output with invalid placeholder guid –

Placeholder_Guid	Placeholder_Name	Channel_Name	Channel_Type	Placeholder_Offer	Placeholder_Created	Placeholder_Me	Placeholder_Is_A	Message
54C5AA3B-AE	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	[NULL]	No Placeholder found with the g

InsertBit

```
CREATE OR ALTER PROCEDURE InsertBit
    @Cust_Fed_ID           VARCHAR(100) = NULL,
    @Offer_Guid             UNIQUEIDENTIFIER = NULL,
    @Offer_Name              VARCHAR(100) = NULL,
    @Bit_Type                VARCHAR(100),      -- mandatory
    @Channel_Name            VARCHAR(100) = NULL,
    @Placeholder_Guid        UNIQUEIDENTIFIER = NULL, -- fixed typo
    @Placeholder_Name        VARCHAR(100) = NULL,
    @InsertedBitGuid         UNIQUEIDENTIFIER OUTPUT, -- return newly inserted GUID
    @StatusCode               INT OUTPUT,          -- 0 = Success, -1 = Error
    @ErrorMessage             NVARCHAR(4000) OUTPUT

AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        DECLARE @NewGuid TABLE (Bit_Guid UNIQUEIDENTIFIER);

        -- Insert without specifying Bit_Guid (default NEWSEQUENTIALID() will generate it)
        INSERT INTO Bits
        (
            Cust_Fed_ID,
            Offer_Guid,
            Offer_Name,
            Bit_Type,
            Channel_Name,
            Placeholder_Guid,
            Placeholder_Name
        )
        OUTPUT INSERTED.Bit_Guid INTO @NewGuid -- capture and insert auto-generated GUID
        into temp table
        VALUES
        (
            @Cust_Fed_ID,
            @Offer_Guid,
            @Offer_Name,
            @Bit_Type,
            @Channel_Name,
            @Placeholder_Guid,
            @Placeholder_Name
        );

        -- Success
        -- Set the output variable from the table variable
        SELECT TOP 1 @InsertedBitGuid = Bit_Guid FROM @NewGuid;
        SET @StatusCode = 0;
        SET @ErrorMessage = NULL;
    END TRY
    BEGIN CATCH
        SET @InsertedBitGuid = NULL;
        SET @StatusCode = -1;
        SET @ErrorMessage = ERROR_MESSAGE();
    END CATCH
END;
```

```
-- ===== SP Execution =====

DECLARE @InsertedBitGuid UNIQUEIDENTIFIER,
        @StatusCode INT,
        @ErrorMessage NVARCHAR(4000);

EXEC InsertBit
    @Cust_Fed_ID = '8391046275',
    @Offer_Guid = 'F8E26247-5587-F011-9F57-64D69AE7FAE4',
    @Offer_Name = 'Gold Loan Offer',
    @Bit_Type = 'VIEW',
    @Channel1_Name = 'NLI',
    @Placeholder_Guid = '52C5AA3B-AB8B-F011-9F57-64D69AE7FAE4',
    @Placeholder_Name = 'Home Page Banner',
    @InsertedBitGuid = @InsertedBitGuid OUTPUT,
    @StatusCode = @StatusCode OUTPUT,
    @ErrorMessage = @ErrorMessage OUTPUT;

SELECT @InsertedBitGuid AS BitGuid, @StatusCode AS StatusCode, @ErrorMessage AS
ErrorMessage;
```

Output after entering the bit –

BitGuid	StatusCode	ErrorMessage
33DCFBBB-19C5-F011-9F79-CD6F5F8138AD	0	[NULL]

GetBitDetails

```
CREATE OR ALTER PROCEDURE GetBitDetails
    @Bit_Guid UNIQUEIDENTIFIER,
    @StatusCode INT OUTPUT -- 0 = Success, 1 = Not Found, -1 = Error
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        -- Check if Bit exists
        IF EXISTS (
            SELECT 1 FROM Bits
            WHERE Bit_Guid = @Bit_Guid
        )
        BEGIN
            SET @StatusCode = 0; -- Success

            SELECT
                Bit_Guid,
                Cust_Fed_ID,
                Offer_Guid,
                Offer_Name,
                Bit_Type,
                Channel_Name,
                Placeholder_Guid,
                Placeholder_Name,
                Action_Timestamp,
                'Success' AS Message
            FROM Bits
            WHERE Bit_Guid = @Bit_Guid;
        END
        ELSE
        BEGIN
            SET @StatusCode = 1; -- Not Found

            SELECT
                @Bit_Guid AS Bit_Guid,
                CAST(NULL AS VARCHAR(100)) AS Cust_Fed_ID,
                CAST(NULL AS UNIQUEIDENTIFIER) AS Offer_Guid,
                CAST(NULL AS VARCHAR(100)) AS Offer_Name,
                CAST(NULL AS VARCHAR(100)) AS Bit_Type,
                CAST(NULL AS VARCHAR(100)) AS Channel_Name,
                CAST(NULL AS UNIQUEIDENTIFIER) AS Placeholder_Guid,
                CAST(NULL AS VARCHAR(100)) AS Placeholder_Name,
                CAST(NULL AS DATETIME) AS Action_Timestamp,
                'No Bit found with the given Bit ID' AS Message;
        END
    END TRY
    BEGIN CATCH
        SET @StatusCode = -1; -- Error

        SELECT
            @Bit_Guid AS Bit_Guid,
            CAST(NULL AS VARCHAR(100)) AS Cust_Fed_ID,
            CAST(NULL AS UNIQUEIDENTIFIER) AS Offer_Guid,
            CAST(NULL AS VARCHAR(100)) AS Offer_Name,
            CAST(NULL AS VARCHAR(100)) AS Bit_Type,
            CAST(NULL AS VARCHAR(100)) AS Channel_Name,
            CAST(NULL AS UNIQUEIDENTIFIER) AS Placeholder_Guid,
            CAST(NULL AS VARCHAR(100)) AS Placeholder_Name,
            CAST(NULL AS DATETIME) AS Action_Timestamp,
```

```
N'Error: ' + ERROR_MESSAGE() AS Message;
END CATCH
END;

-- ===== SP Execution =====

DECLARE @Status INT;

EXEC GetBitDetails
    @Bit_Guid = '442994FF-EA8B-F011-9F57-64D69AE7FAE4',
    @StatusCode = @Status OUTPUT;

SELECT @Status AS StatusCode;
```

Output value for valid Bit Guid –

Bit_Guid	Cust_Fed_ID	Offer_Guid	Offer_Name	Bit_Type	channel_Nam	placeholder_Guc	placeholder_Nation	Timestamp	Message
33DCFBBB-19	8391046275	F8E26247-55	Gold Loan Of	VIEW	NLI	52C5AA3B-AE	Home Page B	01:17.7	Success

Output for invalid Bit Guid –

GetCustomerOffer

What is best option

- **Option 1** - Return JSON data directly from sql server database
- **Option 2** - Return a table and then convert the data into JSON using ASP.NET Core code

Best Option: Option 2 — Return a Table and Convert to JSON in ASP.NET Core

Performance & Scalability

SQL Server is optimized for set-based operations, not heavy JSON formatting.
ASP.NET Core is faster and more scalable for serialization using System.Text.Json.

Maintainability

Application-layer JSON generation is easier to debug and change.
SQL JSON queries become complex and harder to maintain for large data models.

Flexibility

In ASP.NET Core you can:

- Shape JSON dynamically
 - Add fields, remove fields
 - Apply mapping profiles (AutoMapper)
 - Version APIs easily
- SQL cannot do these cleanly.

Error Handling

ASP.NET Core provides proper model validation, exception handling, and logging.
SQL Server gives limited error visibility when generating JSON.

Security

Better control of:

- Filtering
 - Data masking
 - Authorization
- before outputting JSON.

When Option 1 is OK (Return JSON directly)

- Response is small and predictable
- No transformation is needed
- Your SQL already uses FOR JSON PATH cleanly
- You want a very lightweight API layer

Example scenario: *simple fetch + direct respond REST API without business logic.*

Summary Table

Factor	Option 1: JSON in SQL	Option 2: JSON in API (Best)
Performance	✗ Slower	<input checked="" type="checkbox"/> Faster
Code Maintainability	✗ Hard	<input checked="" type="checkbox"/> Easy
Debugging	✗ Limited	<input checked="" type="checkbox"/> Strong
API Flexibility	✗ Low	<input checked="" type="checkbox"/> High
Error Handling	✗ Poor	<input checked="" type="checkbox"/> Great
Complex Data Models	✗ Painful	<input checked="" type="checkbox"/> Easy

Table 1: Customer + Placeholder metadata

Table 2: Offer list (each row = one offer)

```

CREATE OR ALTER PROCEDURE GetCustomerOffers
    @Cust_Fed_ID VARCHAR(100),
    @Placeholder_Guid UNIQUEIDENTIFIER,
    @StatusCode INT OUTPUT,
    @Message NVARCHAR(500) OUTPUT
AS
BEGIN
    SET NOCOUNT ON;

    BEGIN TRY
        -----
        -- CUSTOMER VALIDATION
        -----
        IF dbo.fn_IsCustomerValid(@Cust_Fed_ID) = 0
        BEGIN
            SET @StatusCode = 0;
            SET @Message = 'Customer is inactive';
            RETURN;
        END

        IF dbo.fn_IsCustomerValid(@Cust_Fed_ID) = -1
        BEGIN
            SET @StatusCode = 0;
            SET @Message = 'Customer not found';
            RETURN;
        END

        -----
        -- PLACEHOLDER VALIDATION
        -----
        IF dbo.fn_IsPlaceholderValid(@Placeholder_Guid) = 0
        BEGIN
            SET @StatusCode = 0;
            SET @Message = 'Placeholder is inactive';
            RETURN;
        END

        IF dbo.fn_IsPlaceholderValid(@Placeholder_Guid) = -1
        BEGIN
            SET @StatusCode = 0;
            SET @Message = 'Placeholder not found';
            RETURN;
        END
    END TRY
    BEGIN CATCH
        IF @@TRANCOUNT > 0
            ROLLBACK TRANSACTION;
        ELSE
            IF XACT_STATE() < 0
                ROLLBACK TRANSACTION;
            ELSE
                IF XACT_STATE() > 0
                    COMMIT TRANSACTION;
        END CATCH
    END

```

```

-----  

-- TABLE 1: CUSTOMER + PLACEHOLDER DETAILS  

-----  

SELECT
    c.Cust_Fed_ID AS cust_fed_id,
    p.Placeholder_Guid AS placeholder_guid,
    p.Placeholder_Name AS placeholder_name,
    p.Channel_Name AS channel_name,
    p.Channel_Type AS channel_type,
    p.Placeholder_Offer_Limit AS placeholder_offer_limit,
    p.Placeholder_Meta AS placeholder_meta
FROM Customers c
INNER JOIN Placeholders p ON p.Placeholder_Guid = @Placeholder_Guid
WHERE c.Cust_Fed_ID = @Cust_Fed_ID;

-----  

-- TABLE 2: OFFER LIST  

-----  

;WITH Stats3Days AS (
    SELECT
        Cust_Fed_ID,
        Offer_Guid,
        SUM(View_Count) AS Views3Days,
        SUM(Click_Count) AS Clicks3Days
    FROM CustomerOfferStatsDaily
    WHERE Stat_Date >= CAST(DATEADD(DAY, -2, CAST(GETDATE() AS DATE)) AS DATE)
    GROUP BY Cust_Fed_ID, Offer_Guid
)

SELECT DISTINCT
    o.Offer_Guid AS offer_guid,
    o.Offer_Name AS offer_name,
    o.Offer_Description AS offer_description,
    o.Offer_Type AS offer_type,
    o.Offer_Priority AS offer_priority,
    o.Offer_View_Threshold AS offer_view_threshold,
    o.Offer_Click_Threshold AS offer_click_threshold,
    o.Offer_Banner_Type AS offer_banner_type,
    o.Offer_Content AS offer_content,
    o.Offer_Extra_Content AS offer_extra_content,
    o.Offer_Meta AS offer_meta
FROM CustomerOffers co
INNER JOIN Offers o ON co.Offer_ID = o.Offer_ID
INNER JOIN PlaceholderOffers po ON o.Offer_ID = po.Offer_ID
INNER JOIN Placeholders p2 ON po.Placeholder_ID = p2.Placeholder_ID
LEFT JOIN Stats3Days s ON co.Cust_Fed_ID = s.Cust_Fed_ID
    AND o.Offer_Guid = s.Offer_Guid
WHERE co.Cust_Fed_ID = @Cust_Fed_ID
    AND p2.Placeholder_Guid = @Placeholder_Guid
    AND o.Offer_Visibility_Flag = 1
    AND ISNULL(s.Views3Days,0) < o.Offer_View_Threshold
    AND ISNULL(s.Clicks3Days,0) < o.Offer_Click_Threshold

UNION

-----  

-- TABLE 2: OFFER LIST (Default Offers)  

-----  

SELECT
    o.Offer_Guid,
    o.Offer_Name,
    o.Offer_Description,
    o.Offer_Type,
    o.Offer_Priority,
    o.Offer_View_Threshold,

```

```

        o.Offer_Click_Threshold,
        o.Offer_Banner_Type,
        o.Offer_Content,
        o.Offer_Extra_Content,
        o.Offer_Meta
    FROM PlaceholderOffers po
    INNER JOIN Placeholders p2 ON p2.Placeholder_ID = po.Placeholder_ID
    INNER JOIN Offers o ON o.Offer_ID = po.Offer_ID
    WHERE p2.Placeholder_Guid = @Placeholder_Guid
        AND o.Offer_Type = 'Default'
        AND o.Offer_Visibility_Flag = 1;

    -- SUCCESS STATUS
    SET @StatusCode = 1;
    SET @Message = 'Success';

END TRY
BEGIN CATCH
    SET @StatusCode = 0;
    SET @Message = 'Error: ' + ERROR_MESSAGE();
END CATCH
END

-- ===== SP Execution =====
DECLARE @Status INT;
DECLARE @Msg NVARCHAR(500);

EXEC GetCustomerOffers
    @Cust_Fed_ID = '6049382751',
    @Placeholder_Guid = '56C5AA3B-AB8B-F011-9F57-64D69AE7FAE4',
    @StatusCode = @Status OUTPUT,
    @Message = @Msg OUTPUT;

SELECT @Status AS StatusCode,@Msg AS Message;

```

The above query will return a dataset which will have two tables

Output of Table 1 (Customer + Placeholder)

cust_fed_id	placeholder_placeholder_n	channel_name	channel_type	placeholder_placeholder_meta
6049382751	56C5AA3B-AB8B-F011-9F57-64D69AE7FAE4	Marchent Dasl	InstaBIZ	Mobile

Output of table 2 (Offer Table)

offer_guid	offer_name	offer_descrip	offer_type	offer_priority	offer_view_t	offer_click_t	offer_banner	offer_content	offer_extra_c	offer_meta
CBDA7CC0-53	Pre-Approved Home Loa	Offer Descrip	NBA		1	10	3	Static	{ "redirection_URL": "https://www.product.com", "product": "Home Loan Offer" }	
F6E26247-558	Forex Offer	Offer Descrip	NBA		3	10	3	Static	{ "redirection_URL": "https://www.product.com", "product": "Forex Offer" }	
F8E26247-558	Gold Loan Offer	Offer Descrip	NBA		5	10	3	Static	{ "redirection_URL": "https://www.product.com", "product": "Gold Loan Offer" }	
FAE26247-558	Real Time MF Investmen	Offer Descrip	NBA		7	10	3	Static	{ "redirection_URL": "https://www.product.com", "product": "Real Time MF Investment Offer" }	
FCE26247-558	Mutual Fund Offer	Offer Descrip	NBA		9	10	3	Static	{ "redirection_URL": "https://www.product.com", "product": "Mutual Fund Offer" }	

Trigger

Trg_UpdateClickViewStatusDaily

```
CREATE TRIGGER trg_UpdateClickViewStatusDaily
ON Bits
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;

    MERGE CustomerOfferStatsDaily AS target
    USING (
        SELECT
            CAST(Action_Timestamp AS DATE) AS Stat_Date,
            Cust_Fed_ID,
            Offer_Guid,
            SUM(CASE WHEN Bit_Type='VIEW' THEN 1 ELSE 0 END) AS ViewCnt,
            SUM(CASE WHEN Bit_Type='CLICK' THEN 1 ELSE 0 END) AS ClickCnt
        FROM inserted
        GROUP BY CAST(Action_Timestamp AS DATE), Cust_Fed_ID, Offer_Guid
    ) AS src
    ON (target.Stat_Date = src.Stat_Date
        AND target.Cust_Fed_ID = src.Cust_Fed_ID
        AND target.Offer_Guid = src.Offer_Guid)
    WHEN MATCHED THEN
        UPDATE SET
            View_Count = target.View_Count + src.ViewCnt,
            Click_Count = target.Click_Count + src.ClickCnt
    WHEN NOT MATCHED THEN
        INSERT (Stat_Date, Cust_Fed_ID, Offer_Guid, View_Count, Click_Count)
        VALUES (src.Stat_Date, src.Cust_Fed_ID, src.Offer_Guid, src.ViewCnt, src.ClickCnt);
END;

-- ===== SP Execution =====
EXEC sp_helptext 'trg_UpdateClickViewStatusDaily'
```

Use – This trigger automatically updates daily offer interaction statistics whenever new records are inserted into the **Bits** table. It aggregates VIEW and CLICK events from the inserted rows, groups them by date, customer, and offer, and then performs a **MERGE** operation on **CustomerOfferStatsDaily**. If a matching record already exists, it increments the existing view and click counters. If no matching record is found, it inserts a new daily stats row. This ensures real-time, consistent, and consolidated tracking of user interactions without requiring additional application-level logic.

Note – A trigger is tied to a table. So, if the table is dropped & recreated → **trigger disappears**. So if you delete the Bits table then this trigger will also get deleted.