

CS-GY 6083 - B, FALL 2025

Principles of Database Systems

PROJECT PART1

(Submission Date - 7 Nov, 2025)

TEAM MEMBERS :

- 1) Shivangi Raj (sr7731)
- 2) Soham Tyagi (sx2564)
- 3) Divij Kapur (dk4999)

TABLE OF CONTENTS

A. Introduction

- 1) Project Overview and Business Case (*Netflix NEWS System*)
- 2) Scope and Objectives

B. Assumptions

- 1) Business Rule Assumptions
- 2) Technical Assumptions

C. Logical Diagram

D. Relation Diagram

E. CHECK Constraints and DDL Implementation

- 1) CHECK Constraints (Business Rules and Descriptions)
- 2) DDL Code for CHECK Constraints

F. Data Population

G. Data Dictionary Queries

H. Full DDL Code :

I. Conclusion :

A) INTRODUCTION :

1) Project Overview and Business Case :

This report presents the design and implementation of a relational database system for the **Netflix Episodes Web System (NEWS)**. The objective of the project is to build a robust **Online Transaction Processing (OLTP)** database that effectively manages detailed information about web series, including their seasons, episodes, production houses, and associated creative teams.

The database serves as a scalable and reliable data backbone for NEWS, supporting efficient content management, maintaining high levels of data integrity, and ensuring rapid information access. The design strictly follows recognized database design standards and normalization principles—up to the **Third Normal Form (3NF)**—to eliminate redundancy, preserve consistency, and optimize transactional performance.

2) Scope and Objectives :

This phase focuses on designing and implementing the database foundation for the **Netflix Episodes Web System (NEWS)**. It defines the complete logical and physical structure required for reliable data storage and retrieval.

The main objectives include:

- Developing a full **Logical ERD** that identifies all entities, attributes, and relationships.
- Converting the logical model into a **Relational Schema** with primary, foreign, and check constraints.
- Writing executable **DDL and DML scripts** to create and populate all tables.
- Building a detailed **Data Dictionary** describing metadata and schema properties.
- Ensuring full compliance with normalization and database design standards for efficiency and integrity.

B) ASSUMPTIONS :

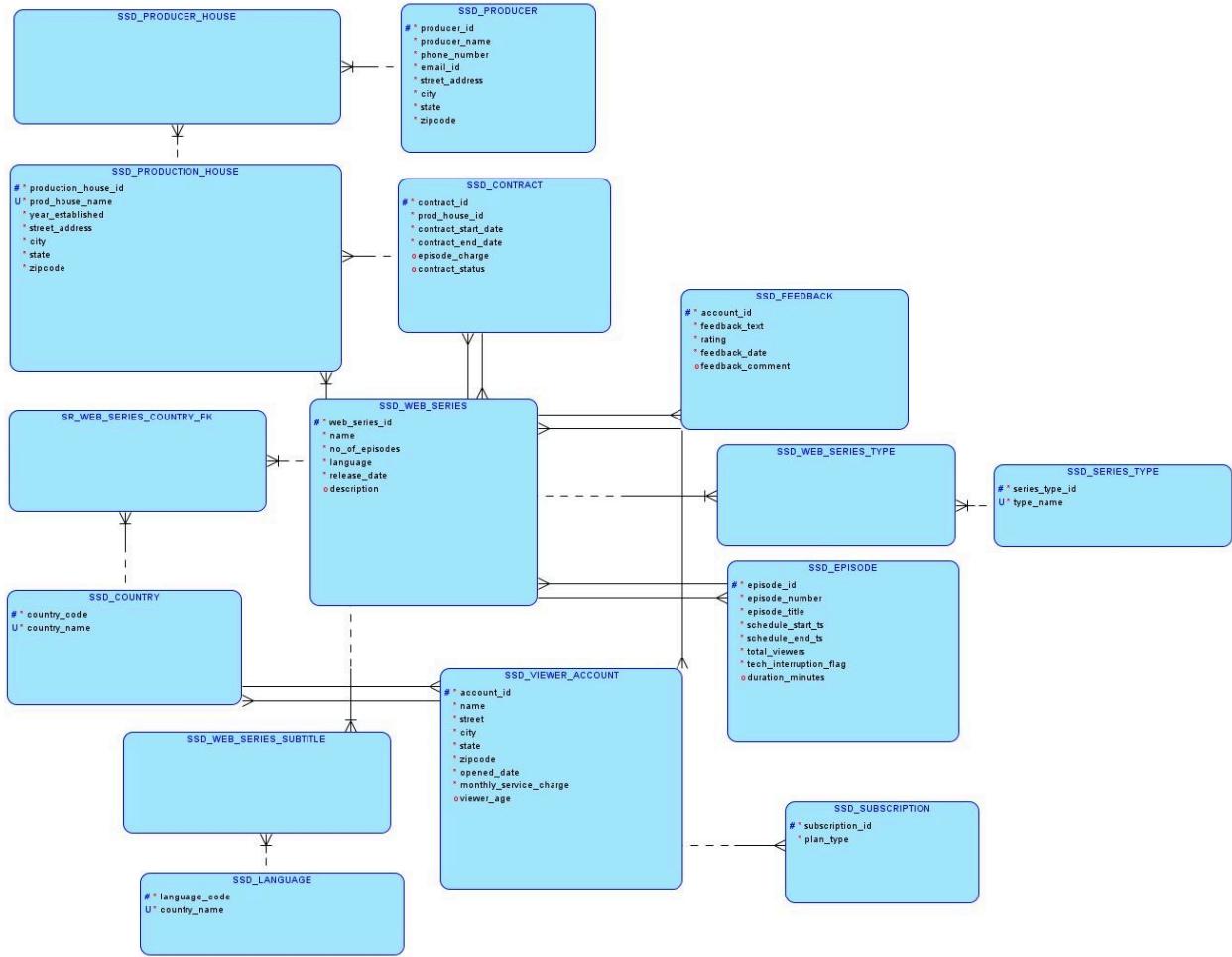
1 Business Rule Assumptions

- Each **Producer** can collaborate with multiple **Production Houses**, and vice versa, through an associative table (**SSD_PRODUCER_HOUSE**).
- Every **Web Series** belongs to one **Production House** but may be released in multiple **Countries**, available in various **Languages** (original, dubbed, subtitled).
- Each **Web Series** consists of one or more **Episodes**, and each episode records schedule, total viewers, and technical interruption status.
- Every **Viewer Account** can provide only one feedback entry per web series, with ratings restricted to a 1–5 scale.
- **Contracts** define the relationship between Producers and Production Houses, ensuring valid start and end dates and positive per-episode charges.
- The **Subscription Plan** (Basic, Standard, or Premium) governs account access and is linked to each **Viewer Account**.

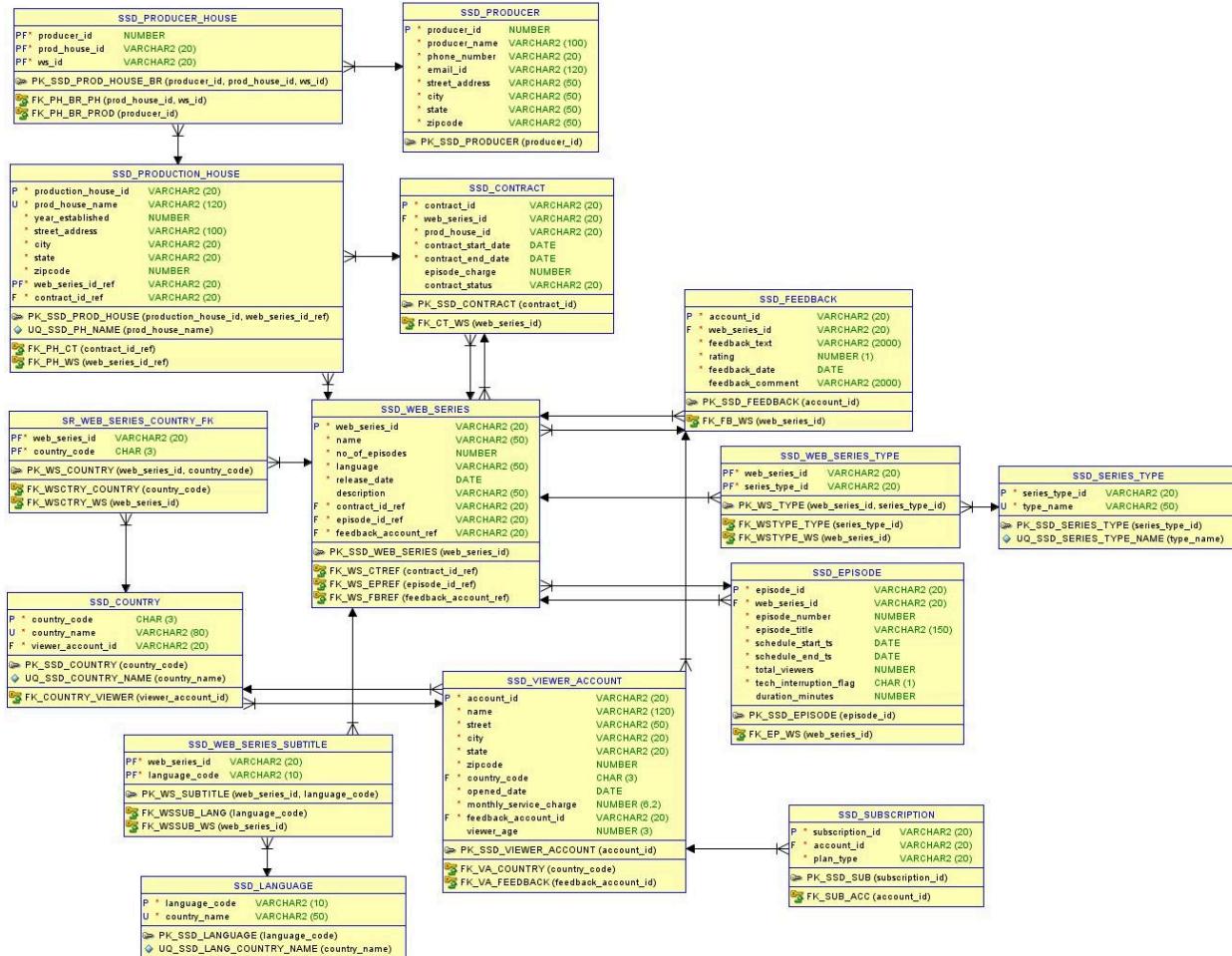
2 Technical Assumptions

- All tables follow the naming convention prefix **SSD_** and maintain **Third Normal Form (3NF)** to reduce redundancy and ensure consistency.
- Each table includes a **primary key**, with all relationships maintained via properly defined **foreign keys** enforcing referential integrity.
- **CHECK constraints** enforce business logic—such as valid age (≥ 13), positive service charges, rating range, and logical contract dates.
- Textual attributes (e.g., names, addresses, city) use **VARCHAR2/VARCHAR** types with standardized lengths; date/time attributes use **DATE** or **TIMESTAMP**.
- All identifiers are unique numeric values, generated either through Oracle identity columns or MySQL auto-increment mechanisms.
- Each table is populated with at least **10–30 meaningful records** to satisfy data completeness and validation requirements.

C) LOGICAL DIAGRAM:



D) RELATIONAL DIAGRAM:



E) CHECK CONSTRAINTS AND DDL IMPLEMENTATION:

1) CHECK Constraints (Business Rules and Descriptions)

#	Table	Constraint Name	Condition (Check Logic)	Rule Description
1	SSD_CONTRACT	CHK_CT_END_GT_STA RT	contract_end_date > contract_start_date	Ensures the contract cannot end before it starts.
2	SSD_CONTRACT	CHK_CT_STATUS	contract_status IN ('Active', 'Expired', 'Pending', 'Terminated')	Restricts contract status to predefined valid values.
3	SSD_EPISODE	CHK_EP_DUR_RANGE	duration_minutes BETWEEN 1 AND 600	Limits episode duration to a realistic range (1–600 min).
4	SSD_EPISODE	CHK_EP_NUM_POS	episode_number > 0	Guarantees every episode has a positive number.
5	SSD_EPISODE	CHK_EP_TECH_FLAG	tech_interruption_flag IN ('Y', 'N')	Ensures technical interruption flag holds only Y/N values.
6	SSD_FEEDBACK	CHK_FB_CMT_REQ	NOT (rating <= 2 AND feedback_comment IS NULL)	Requires feedback comment when rating ≤ 2 (poor reviews).
7	SSD_FEEDBACK	CHK_FB_RATING	rating BETWEEN 1 AND 5	Restricts feedback ratings to a 1–5 scale.
8	SSD_SUBSCRIPTION	CHK_SUB_PLAN_TYPE	plan_type IN ('Basic', 'Standard', 'Premium')	Ensures valid subscription plans only.
9	SSD_VIEWER_ACCOUNT	CHK_VA_MIN_AGE	viewer_age >= 13	Enforces minimum viewer age (13+).

10	SSD_VIEWER_ACCOUNT	CHK_VA_MSC_POS	monthly_service_charge > 0	Ensures monthly service charge is a positive value.
----	--------------------	----------------	----------------------------	---

2) DDL Code for CHECK Constraints

```
-- 1. Contract End Date > Start Date
ALTER TABLE SSD_CONTRACT
    ADD CONSTRAINT CHK_CT_END_GT_START
        CHECK (contract_end_date > contract_start_date);

-- 2. Contract Status Validity
ALTER TABLE SSD_CONTRACT
    ADD CONSTRAINT CHK_CT_STATUS
        CHECK (contract_status IN
('Active','Expired','Pending','Terminated'));

-- 3. Episode Duration Range
ALTER TABLE SSD_EPISODE
    ADD CONSTRAINT CHK_EP_DUR_RANGE
        CHECK (duration_minutes BETWEEN 1 AND 600);

-- 4. Episode Number Must Be Positive
ALTER TABLE SSD_EPISODE
    ADD CONSTRAINT CHK_EP_NUM_POS
        CHECK (episode_number > 0);

-- 5. Technical Interruption Flag Validity
ALTER TABLE SSD_EPISODE
    ADD CONSTRAINT CHK_EP_TECH_FLAG
        CHECK (tech_interruption_flag IN ('Y','N'));

-- 6. Feedback Comment Requirement for Low Rating
ALTER TABLE SSD_FEEDBACK
    ADD CONSTRAINT CHK_FB_CMT_REQ
        CHECK (NOT (rating <= 2 AND feedback_comment IS NULL));

-- 7. Feedback Rating Range
ALTER TABLE SSD_FEEDBACK
    ADD CONSTRAINT CHK_FB_RATING
        CHECK (rating BETWEEN 1 AND 5);
```

```
-- 8. Subscription Plan Type Restriction
ALTER TABLE SSD_SUBSCRIPTION
    ADD CONSTRAINT CHK_SUB_PLAN_TYPE
        CHECK (plan_type IN ('Basic','Standard','Premium'));

-- 9. Viewer Minimum Age Requirement
ALTER TABLE SSD_VIEWER_ACCOUNT
    ADD CONSTRAINT CHK_VA_MIN_AGE
        CHECK (viewer_age >= 13);

-- 10. Monthly Service Charge Must Be Positive
ALTER TABLE SSD_VIEWER_ACCOUNT
    ADD CONSTRAINT CHK_VA_MSC_POS
        CHECK (monthly_service_charge > 0);
```

F) POPULATE DATA :

QUERY:

```
•   SELECT 'SSD_LANGUAGE' AS Table_Name, COUNT(*) AS Total_Records FROM SSD_LANGUAGE
UNION ALL
SELECT 'SSD_SERIES_TYPE', COUNT(*) FROM SSD_SERIES_TYPE
UNION ALL
SELECT 'SSD_PRODUCER', COUNT(*) FROM SSD_PRODUCER
UNION ALL
SELECT 'SSD_COUNTRY', COUNT(*) FROM SSD_COUNTRY
UNION ALL
SELECT 'SSD_VIEWER_ACCOUNT', COUNT(*) FROM SSD_VIEWER_ACCOUNT
UNION ALL
SELECT 'SSD_SUBSCRIPTION', COUNT(*) FROM SSD_SUBSCRIPTION
UNION ALL
SELECT 'SSD_PRODUCTION_HOUSE', COUNT(*) FROM SSD_PRODUCTION_HOUSE
UNION ALL
SELECT 'SSD_WEB_SERIES', COUNT(*) FROM SSD_WEB_SERIES
UNION ALL
SELECT 'SSD_EPISODE', COUNT(*) FROM SSD_EPISODE
UNION ALL
SELECT 'SSD_FEEDBACK', COUNT(*) FROM SSD_FEEDBACK
UNION ALL
SELECT 'SSD_CONTRACT', COUNT(*) FROM SSD_CONTRACT
UNION ALL
SELECT 'SR_WEB_SERIES_COUNTRY_FK', COUNT(*) FROM SR_WEB_SERIES_COUNTRY_FK
UNION ALL
SELECT 'SSD_WEB_SERIES_SUBTITLE', COUNT(*) FROM SSD_WEB_SERIES_SUBTITLE
UNION ALL
SELECT 'SSD_WEB_SERIES_TYPE', COUNT(*) FROM SSD_WEB_SERIES_TYPE
UNION ALL
SELECT 'SSD_PRODUCER_HOUSE', COUNT(*) FROM SSD_PRODUCER_HOUSE;
```

RESULT:

Table_Name	Total_Records
SSD_LANGUAGE	26
SSD_SERIES_TYPE	26
SSD_PRODUCER	26
SSD_COUNTRY	26
SSD_VIEWER_ACCOUNT	26
SSD_SUBSCRIPTION	26
SSD_PRODUCTION_HOUSE	26
SSD_WEB_SERIES	26
SSD_EPISODE	26
SSD_FEEDBACK	26
SSD_CONTRACT	26
SR_WEB_SERIES_COUNT...	26
SSD_WEB_SERIES_SUBTI...	26
SSD_WEB_SERIES_TYPE	27
SSD_PRODUCER_HOUSE	26

G) DATA DICTIONARY QUERIES:

QUERY 1 :

Lists all tables in your database with creation details

```
-- Query 1: Table Summary
SELECT
    table_name AS "Table Name",
    table_type AS "Type",
    engine AS "Storage Engine",
    table_rows AS "Approx Rows",
    create_time AS "Created On"
FROM
    information_schema.tables
WHERE
    table_schema = 'netflix_news'
ORDER BY
    Table_name;
```

RESULT :

Table Name	Type	Storage Engine	Approx Rows	Created On
SR_WEB_SERIES_COUNTRY_FK	BASE TABLE	InnoDB	26	2025-11-07 20:44:52
SSD_CONTRACT	BASE TABLE	InnoDB	26	2025-11-07 20:40:00
SSD_COUNTRY	BASE TABLE	InnoDB	26	2025-11-07 20:44:40
SSD_EPISODE	BASE TABLE	InnoDB	26	2025-11-07 20:44:52
SSD_FEEDBACK	BASE TABLE	InnoDB	26	2025-11-07 20:44:52
SSD_LANGUAGE	BASE TABLE	InnoDB	26	2025-11-07 20:38:28
SSD_PRODUCER	BASE TABLE	InnoDB	26	2025-11-07 20:39:29
SSD_PRODUCER_HOUSE	BASE TABLE	InnoDB	26	2025-11-07 20:44:52
SSD_PRODUCTION_HOUSE	BASE TABLE	InnoDB	26	2025-11-07 20:41:48
SSD_SERIES_TYPE	BASE TABLE	InnoDB	26	2025-11-07 20:38:28
SSD_SUBSCRIPTION	BASE TABLE	InnoDB	26	2025-11-07 20:40:00
SSD_VIEWER_ACCOUNT	BASE TABLE	InnoDB	26	2025-11-07 20:40:00
SSD_WEB_SERIES	BASE TABLE	InnoDB	26	2025-11-07 20:39:29
SSD_WEB_SERIES_SUBTITLE	BASE TABLE	InnoDB	26	2025-11-07 20:44:52
SSD_WEB_SERIES_TYPE	BASE TABLE	InnoDB	27	2025-11-07 20:44:53

QUERY 2 :

Lists each column with data type, size, and whether it's nullable.

```
-- Query 2: Column Details (Data Type, Size, Nullability)
SELECT
    table_name AS "Table Name",
    column_name AS "Column Name",
    column_type AS "Data Type & Size",
    is_nullable AS "Nullable",
    column_default AS "Default Value",
    column_comment AS "Column Comment"
FROM
    information_schema.columns
WHERE
    table_schema = 'netflix_news'
ORDER BY
    table_name, ordinal_position;
```

RESULT :

Table Name	Column Name	Data Type & Size	Nullable	Default Value	Column Comment
SR_WEB_SERIES_COUNTRY_FK	web_series_id	varchar(20)	NO	NULL	
SR_WEB_SERIES_COUNTRY_FK	country_code	char(3)	NO	NULL	
SSD_CONTRACT	contract_id	varchar(20)	NO	NULL	
SSD_CONTRACT	web_series_id	varchar(20)	YES	NULL	
SSD_CONTRACT	prod_house_id	varchar(20)	YES	NULL	
SSD_CONTRACT	contract_start_date	date	NO	NULL	
SSD_CONTRACT	contract_end_date	date	NO	NULL	
SSD_CONTRACT	episode_charge	decimal(10,2)	YES	NULL	
SSD_CONTRACT	contract_status	varchar(20)	YES	NULL	
SSD_COUNTRY	country_code	char(3)	NO	NULL	
SSD_COUNTRY	country_name	varchar(80)	NO	NULL	
SSD_EPISODE	viewer_account_id	varchar(20)	YES	NULL	
SSD_EPISODE	episode_id	varchar(20)	NO	NULL	
SSD_EPISODE	web_series_id	varchar(20)	NO	NULL	
SSD_EPISODE	episode_number	int	NO	NULL	
SSD_EPISODE	episode_title	varchar(150)	NO	NULL	
SSD_EPISODE	schedule_start_ts	date	NO	NULL	
SSD_EPISODE	schedule_end_ts	date	NO	NULL	
SSD_EPISODE	total_viewers	int	NO	NULL	
SSD_EPISODE	tech_interruption...	char(1)	NO	NULL	
SSD_EPISODE	duration_minutes	int	YES	NULL	
SSD_FEEDBACK	account_id	varchar(20)	NO	NULL	
SSD_FEEDBACK	web_series_id	varchar(20)	NO	NULL	
SSD_FEEDBACK	feedback_text	varchar(2000)	NO	NULL	
SSD_FEEDBACK	rating	int	NO	NULL	
SSD_FEEDBACK	feedback_date	date	NO	NULL	
SSD_FEEDBACK	feedback_comment	varchar(2000)	YES	NULL	
SSD_LANGUAGE	language_code	varchar(10)	NO	NULL	
SSD_LANGUAGE	country_name	varchar(50)	NO	NULL	
SSD_PRODUCER	producer_id	int	NO	NULL	
SSD_PRODUCER	producer_name	varchar(100)	NO	NULL	
SSD_PRODUCER	phone_number	varchar(20)	NO	NULL	
SSD_PRODUCER	email_id	varchar(120)	NO	NULL	
SSD_PRODUCER	street_address	varchar(50)	NO	NULL	
SSD_PRODUCER	city	varchar(50)	NO	NULL	
SSD_PRODUCER	state	varchar(50)	NO	NULL	
SSD_PRODUCER	zipcode	varchar(20)	NO	NULL	
SSD_PRODUCER_HOUSE	producer_id	int	NO	NULL	
SSD_PRODUCER_HOUSE	prod_house_id	varchar(20)	NO	NULL	
SSD_PRODUCER_HOUSE	ws_id	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	production_house...	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	prod_house_name	varchar(120)	NO	NULL	
SSD_PRODUCTION_HOUSE	year_established	int	NO	NULL	
SSD_PRODUCTION_HOUSE	street_address	varchar(100)	NO	NULL	

Table Name	Column Name	Data Type & Size	Nullable	Default Value	Column Comment
SSD_PRODUCER	street_address	varchar(50)	NO	NULL	
SSD_PRODUCER	city	varchar(50)	NO	NULL	
SSD_PRODUCER	state	varchar(50)	NO	NULL	
SSD_PRODUCER	zipcode	varchar(20)	NO	NULL	
SSD_PRODUCER_HOUSE	producer_id	int	NO	NULL	
SSD_PRODUCER_HOUSE	prod_house_id	varchar(20)	NO	NULL	
SSD_PRODUCER_HOUSE	ws_id	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	production_house...	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	prod_house_name	varchar(120)	NO	NULL	
SSD_PRODUCTION_HOUSE	year_established	int	NO	NULL	
SSD_PRODUCTION_HOUSE	street_address	varchar(100)	NO	NULL	
SSD_PRODUCTION_HOUSE	city	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	state	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	zipcode	int	NO	NULL	
SSD_PRODUCTION_HOUSE	web_series_id_ref	varchar(20)	NO	NULL	
SSD_PRODUCTION_HOUSE	contract_id_ref	varchar(20)	YES	NULL	
SSD_SERIES_TYPE	series_type_id	varchar(20)	NO	NULL	
SSD_SERIES_TYPE	type_name	varchar(50)	NO	NULL	
SSD_SUBSCRIPTION	subscription_id	varchar(20)	NO	NULL	
SSD_SUBSCRIPTION	account_id	varchar(20)	NO	NULL	
SSD_SUBSCRIPTION	plan_type	varchar(20)	NO	NULL	
SSD_VIEWER_ACCOUNT	account_id	varchar(20)	NO	NULL	
SSD_VIEWER_ACCOUNT	name	varchar(120)	NO	NULL	
SSD_VIEWER_ACCOUNT	street	varchar(50)	NO	NULL	
SSD_VIEWER_ACCOUNT	city	varchar(20)	NO	NULL	
SSD_VIEWER_ACCOUNT	state	varchar(20)	NO	NULL	
SSD_VIEWER_ACCOUNT	zipcode	int	NO	NULL	
SSD_VIEWER_ACCOUNT	country_code	char(3)	NO	NULL	
SSD_VIEWER_ACCOUNT	opened_date	date	NO	NULL	
SSD_VIEWER_ACCOUNT	monthly_service_...	decimal(6,2)	NO	NULL	
SSD_VIEWER_ACCOUNT	feedback_acoun...	varchar(20)	YES	NULL	
SSD_VIEWER_ACCOUNT	viewer_age	int	YES	NULL	
SSD_WEB_SERIES	web_series_id	varchar(20)	NO	NULL	
SSD_WEB_SERIES	name	varchar(50)	NO	NULL	
SSD_WEB_SERIES	no_of_episodes	int	NO	NULL	
SSD_WEB_SERIES	language	varchar(50)	NO	NULL	
SSD_WEB_SERIES	release_date	date	NO	NULL	
SSD_WEB_SERIES	description	varchar(255)	YES	NULL	
SSD_WEB_SERIES	contract_id_ref	varchar(20)	YES	NULL	
SSD_WEB_SERIES	episode_id_ref	varchar(20)	YES	NULL	
SSD_WEB_SERIES	feedback_acoun...	varchar(20)	YES	NULL	
SSD_WEB_SERIES_SUBTITLE	web_series_id	varchar(20)	NO	NULL	
SSD_WEB_SERIES_SUBTITLE	language_code	varchar(10)	NO	NULL	
SSD_WEB_SERIES_TYPE	web_series_id	varchar(20)	NO	NULL	
SSD_WEB_SERIES_TYPE	series_type_id	varchar(20)	NO	NULL	

QUERY 3 :

Constraints (Primary Key, Foreign Key, Unique, Check)

```
-- | Query 3: Constraint Definitions
SELECT
    tc.table_name AS "Table Name",
    tc.constraint_name AS "Constraint Name",
    tc.constraint_type AS "Constraint Type",
    kcu.column_name AS "Column Name",
    kcu.referenced_table_name AS "Referenced Table",
    kcu.referenced_column_name AS "Referenced Column"
FROM
    information_schema.table_constraints tc
LEFT JOIN
    information_schema.key_column_usage kcu
ON tc.constraint_name = kcu.constraint_name
AND tc.table_name = kcu.table_name
WHERE
    tc.table_schema = 'netflix_news'
ORDER BY
    tc.table_name, tc.constraint_type;
```

RESULT :

Table Name	Constraint Name	Constraint Ty...	Column Name	Referenced Table	Referenced Colu...
SR_WEB_SERIES_COUNTRY_FK	FK_WSCTRY_COUNTRY	FOREIGN KEY	country_code	SSD_COUNTRY	country_code
SR_WEB_SERIES_COUNTRY_FK	FK_WSCTRY_WS	FOREIGN KEY	web_series_id	SSD_WEB_SERIES	web_series_id
SR_WEB_SERIES_COUNTRY_FK	PRIMARY	PRIMARY KEY	web_series_id	NULL	NULL
SR_WEB_SERIES_COUNTRY_FK	PRIMARY	PRIMARY KEY	country_code	NULL	NULL
SSD_CONTRACT	CHK_CT_END_GT_START	CHECK	NULL	NULL	NULL
SSD_CONTRACT	CHK_CT_STATUS	CHECK	NULL	NULL	NULL
SSD_CONTRACT	FK_CT_WS	FOREIGN KEY	web_series_id	SSD_WEB_SERIES	web_series_id
SSD_CONTRACT	PRIMARY	PRIMARY KEY	contract_id	NULL	NULL
SSD_COUNTRY	PRIMARY	PRIMARY KEY	country_code	NULL	NULL
SSD_COUNTRY	UQ_SS_D_COUNTRY_NAME	UNIQUE	country_name	NULL	NULL
SSD_EPISODE	CHK_EP_DUR_RANGE	CHECK	NULL	NULL	NULL
SSD_EPISODE	CHK_EP_NUM_POS	CHECK	NULL	NULL	NULL
SSD_EPISODE	CHK_EP_TECH_FLAG	CHECK	NULL	NULL	NULL
SSD_EPISODE	FK_EP_WS	FOREIGN KEY	web_series_id	SSD_WEB_SERIES	web_series_id
SSD_EPISODE	PRIMARY	PRIMARY KEY	episode_id	NULL	NULL
SSD_FEEDBACK	CHK_FB_CMT_REQ	CHECK	NULL	NULL	NULL
SSD_FEEDBACK	CHK_FB_RATING	CHECK	NULL	NULL	NULL
SSD_FEEDBACK	FK_FB_WS	FOREIGN KEY	web_series_id	SSD_WEB_SERIES	web_series_id
SSD_FEEDBACK	PRIMARY	PRIMARY KEY	account_id	NULL	NULL
SSD_LANGUAGE	PRIMARY	PRIMARY KEY	language_code	NULL	NULL
SSD_LANGUAGE	UQ_SS_LANG_COUNTR...	UNIQUE	country_name	NULL	NULL
SSD_PRODUCER	PRIMARY	PRIMARY KEY	producer_id	NULL	NULL

SSD_PRODUCER_HOUSE	FK_PH_BR_PH	FOREIGN KEY	ws_id	SSD_PRODUCTIO...	web_series_id_ref
SSD_PRODUCER_HOUSE	FK_PH_BR_PH	FOREIGN KEY	prod_house_id	SSD_PRODUCTIO...	production_house_id
SSD_PRODUCER_HOUSE	FK_PH_BR_PROD	FOREIGN KEY	producer_id	SSD_PRODUCER	producer_id
SSD_PRODUCER_HOUSE	PRIMARY	PRIMARY KEY	prod_house_id	NULL	NULL
SSD_PRODUCER_HOUSE	PRIMARY	PRIMARY KEY	ws_id	NULL	NULL
SSD_PRODUCER_HOUSE	PRIMARY	PRIMARY KEY	producer_id	NULL	NULL
SSD_PRODUCTION_HOUSE	FK_PH_CT	FOREIGN KEY	contract_id_ref	SSD_CONTRACT	contract_id
SSD_PRODUCTION_HOUSE	FK_PH_WS	FOREIGN KEY	web_series_i...	SSD_WEB_SERIES	web_series_id
SSD_PRODUCTION_HOUSE	PRIMARY	PRIMARY KEY	production_h...	NULL	NULL
SSD_PRODUCTION_HOUSE	PRIMARY	PRIMARY KEY	web_series_i...	NULL	NULL
SSD_PRODUCTION_HOUSE	UQ_SSD_PH_NAME	UNIQUE	prod_house_...	NULL	NULL
SSD_PRODUCTION_HOUSE	UQ_SSD_PROD_HOUSE_ID	UNIQUE	production_h...	NULL	NULL
SSD_SERIES_TYPE	PRIMARY	PRIMARY KEY	series_type_id	NULL	NULL
SSD_SERIES_TYPE	UQ_SSD_SERIES_TYPE...	UNIQUE	type_name	NULL	NULL
SSD_SUBSCRIPTION	CHK_SUB_PLAN_TYPE	CHECK	NULL	NULL	NULL
SSD_SUBSCRIPTION	FK_SUB_ACC	FOREIGN KEY	account_id	SSD_VIEWER_AC...	account_id
SSD_SUBSCRIPTION	PRIMARY	PRIMARY KEY	subscription_id	NULL	NULL
SSD_VIEWER_ACCOUNT	CHK_VA_MIN_AGE	CHECK	NULL	NULL	NULL
SSD_VIEWER_ACCOUNT	CHK_VA_MSC_POS	CHECK	NULL	NULL	NULL
SSD_VIEWER_ACCOUNT	FK_VA_COUNTRY	FOREIGN KEY	country_code	SSD_COUNTRY	country_code
SSD_VIEWER_ACCOUNT	FK_VA_FEEDBACK	FOREIGN KEY	feedback_ac...	SSD_FEEDBACK	account_id
SSD_VIEWER_ACCOUNT	PRIMARY	PRIMARY KEY	account_id	NULL	NULL
SSD_WEB_SERIES	PRIMARY	PRIMARY KEY	web_series_id	NULL	NULL
SSD_WEB_SERIES_SUBTITLE	FK_WSSUB_LANG	FOREIGN KEY	language_code	SSD_LANGUAGE	language_code
SSD_WEB_SERIES_SUBTITLE	FK_WSSUB_WS	FOREIGN KEY	web_series_id	SSD_WEB_SERIES	web_series_id
SSD_WEB_SERIES_SUBTITLE	PRIMARY	PRIMARY KEY	web_series_id	NULL	NULL
SSD_WEB_SERIES_SUBTITLE	PRIMARY	PRIMARY KEY	language_code	NULL	NULL
SSD_WEB_SERIES_TYPE	FK_WSTYPE_TYPE	FOREIGN KEY	series_type_id	SSD_SERIES_TYPE	series_type_id
SSD_WEB_SERIES_TYPE	FK_WSTYPE_WS	FOREIGN KEY	web_series_id	SSD_WEB_SERIES	web_series_id
SSD_WEB_SERIES_TYPE	PRIMARY	PRIMARY KEY	web_series_id	NULL	NULL
SSD_WEB_SERIES_TYPE	PRIMARY	PRIMARY KEY	series_type_id	NULL	NULL

QUERY 4 :

MySQL stores CHECK constraints separately

```
-- Query 4: Check Constraints (Correct MySQL Version)

SELECT
    tc.table_name AS "Table Name",
    cc.constraint_name AS "Constraint Name",
    cc.check_clause AS "Check Condition"
FROM
    information_schema.check_constraints cc
JOIN
    information_schema.table_constraints tc
    ON cc.constraint_name = tc.constraint_name
    AND cc.constraint_schema = tc.table_schema
WHERE
    cc.constraint_schema = 'netflix_news'
ORDER BY
    tc.table_name;
```

RESULT :

Table Name	Constraint Name	Check Condition
SSD_CONTRACT	CHK_CT_END_GT_START	(`contract_end_date` > `contract_start_date`)
SSD_CONTRACT	CHK_CT_STATUS	(`contract_status` in (_utf8mb4'Active',_utf8mb4'Draft'))
SSD_EPISODE	CHK_EP_DUR_RANGE	(`duration_minutes` between 1 and 600)
SSD_EPISODE	CHK_EP_NUM_POS	(`episode_number` > 0)
SSD_EPISODE	CHK_EP_TECH_FLAG	(`tech_interruption_flag` in (_utf8mb4'Y',_utf8mb4'N'))
SSD_FEEDBACK	CHK_FB_CMT_REQ	((`rating` > 2) or (`feedback_comment` is not null))
SSD_FEEDBACK	CHK_FB_RATING	(`rating` between 1 and 5)
SSD_SUBSCRIPTION	CHK_SUB_PLAN_TYPE	(`plan_type` in (_utf8mb4'Basic',_utf8mb4'Standard',_utf8mb4'Premium'))
SSD_VIEWER_ACCOUNT	CHK_VA_MIN_AGE	(`viewer_age` >= 13)
SSD_VIEWER_ACCOUNT	CHK_VA_MSC_POS	(`monthly_service_charge` > 0)

Query 5 :

This query clearly shows **which table depends on which**, along with the **specific columns** involved:

```
-- Query 5: Foreign Key Relationship Mapping
SELECT
    kcu.TABLE_NAME AS "Child Table",
    kcu.COLUMN_NAME AS "Child Column",
    kcu.REFERENCED_TABLE_NAME AS "Parent Table",
    kcu.REFERENCED_COLUMN_NAME AS "Parent Column",
    rc.UPDATE_RULE AS "On Update",
    rc.DELETE_RULE AS "On Delete"
FROM
    information_schema.KEY_COLUMN_USAGE kcu
JOIN
    information_schema.REFERENTIAL_CONSTRAINTS rc
    ON kcu.CONSTRAINT_NAME = rc.CONSTRAINT_NAME
    AND kcu.CONSTRAINT_SCHEMA = rc.CONSTRAINT_SCHEMA
WHERE
    kcu.REFERENCED_TABLE_NAME IS NOT NULL
    AND kcu.CONSTRAINT_SCHEMA = 'netflix_news'
ORDER BY
    kcu.TABLE_NAME, kcu.REFERENCED_TABLE_NAME;
```

RESULT :

Child Table	Child Column	Parent Table	Parent Column	On Update	On Delete	
SR_WEB_SERIES_COUNTRY_FK	country_code	SSD_COUNTRY	country_code	NO ACTION	NO ACTION	
SR_WEB_SERIES_COUNTRY_FK	web_series_id	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	
SSD_CONTRACT	web_series_id	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	
SSD_EPISODE	web_series_id	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	
SSD_FEEDBACK	web_series_id	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	
SSD_PRODUCER_HOUSE	producer_id	SSD_PRODUCER	producer_id	NO ACTION	NO ACTION	
SSD_PRODUCER_HOUSE	prod_house_id	SSD_PRODUCTION_HOUSE	production_house_id	NO ACTION	NO ACTION	
SSD_PRODUCER_HOUSE	ws_id	SSD_PRODUCTION_HOUSE	web_series_id_ref	NO ACTION	NO ACTION	
SSD_PRODUCTION_HOUSE	contract_id_ref	SSD_CONTRACT	contract_id	NO ACTION	NO ACTION	
SSD_PRODUCTION_HOUSE	web_series_id_ref	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	
SSD_SUBSCRIPTION	account_id	SSD_VIEWER_ACCOUNT	account_id	NO ACTION	NO ACTION	
SSD_VIEWER_ACCOUNT	country_code	SSD_COUNTRY	country_code	NO ACTION	NO ACTION	
SSD_VIEWER_ACCOUNT	feedback_accou...	SSD_FEEDBACK	account_id	NO ACTION	NO ACTION	
SSD_WEB_SERIES_SUBTITLE	language_code	SSD_LANGUAGE	language_code	NO ACTION	NO ACTION	
SSD_WEB_SERIES_SUBTITLE	web_series_id	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	
SSD_WEB_SERIES_TYPE	series_type_id	SSD_SERIES_TYPE	series_type_id	NO ACTION	NO ACTION	
SSD_WEB_SERIES_TYPE	web_series_id	SSD_WEB_SERIES	web_series_id	NO ACTION	NO ACTION	

H) FULL DDL CODE:

```
-- Generated by Oracle SQL Developer Data Modeler 24.3.1.351.0831
-- at:          2025-11-07 20:15:53 GMT-05:00
-- site:        Oracle Database 21c
-- type:        Oracle Database 21c

-- predefined type, no DDL - MDSYS.SDO_GEOGRAPHY

-- predefined type, no DDL - XMLTYPE

CREATE TABLE SR_WEB_SERIES_COUNTRY_FK
(
    web_series_id VARCHAR2 (20) NOT NULL ,
    country_code  CHAR (3) NOT NULL
)
LOGGING
;

ALTER TABLE SR_WEB_SERIES_COUNTRY_FK
    ADD CONSTRAINT PK_WS_COUNTRY PRIMARY KEY ( web_series_id,
country_code ) ;

CREATE TABLE SSD_CONTRACT
(
    contract_id      VARCHAR2 (20) NOT NULL ,
    web_series_id    VARCHAR2 (20) NOT NULL ,
    prod_house_id   VARCHAR2 (20) NOT NULL ,
    contract_start_date DATE NOT NULL ,
    contract_end_date DATE NOT NULL ,
    episode_charge   NUMBER ,
    contract_status   VARCHAR2 (20)
)
LOGGING
;

ALTER TABLE SSD_CONTRACT
    ADD CONSTRAINT CHK_CT_END_GT_START
        CHECK (contract_end_date > contract_start_date)
;

ALTER TABLE SSD_CONTRACT
    ADD CONSTRAINT CHK_CT_STATUS
```

```

        CHECK (contract_status IN
('Active','Expired','Pending','Terminated'))
;
ALTER TABLE SSD_CONTRACT
ADD CONSTRAINT PK_SSD_CONTRACT PRIMARY KEY ( contract_id ) ;

CREATE TABLE SSD_COUNTRY
(
    country_code      CHAR (3) NOT NULL ,
    country_name      VARCHAR2 (80) NOT NULL ,
    viewer_account_id VARCHAR2 (20) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_COUNTRY
ADD CONSTRAINT PK_SSD_COUNTRY PRIMARY KEY ( country_code ) ;

ALTER TABLE SSD_COUNTRY
ADD CONSTRAINT UQ_SSD_COUNTRY_NAME UNIQUE ( country_name ) ;

CREATE TABLE SSD_EPISODE
(
    episode_id          VARCHAR2 (20) NOT NULL ,
    web_series_id       VARCHAR2 (20) NOT NULL ,
    episode_number      NUMBER NOT NULL ,
    episode_title       VARCHAR2 (150) NOT NULL ,
    schedule_start_ts   DATE NOT NULL ,
    schedule_end_ts     DATE NOT NULL ,
    total_viewers       NUMBER NOT NULL ,
    tech_interruption_flag CHAR (1) NOT NULL ,
    duration_minutes    NUMBER
)
LOGGING
;
ALTER TABLE SSD_EPISODE
ADD CONSTRAINT CHK_EP_DUR_RANGE
CHECK (duration_minutes BETWEEN 1 AND 600)
;
ALTER TABLE SSD_EPISODE
ADD CONSTRAINT CHK_EP_NUM_POS
CHECK (episode_number > 0)
;
ALTER TABLE SSD_EPISODE
ADD CONSTRAINT CHK_EP_TECH_FLAG

```

```

        CHECK (tech_interruption_flag IN ('Y','N'))
;
ALTER TABLE SSD_EPISODE
    ADD CONSTRAINT PK_SSD_EPISODE PRIMARY KEY ( episode_id ) ;

CREATE TABLE SSD_FEEDBACK
(
    account_id      VARCHAR2 (20) NOT NULL ,
    web_series_id   VARCHAR2 (20) NOT NULL ,
    feedback_text   VARCHAR2 (2000) NOT NULL ,
    rating          NUMBER (1) NOT NULL ,
    feedback_date   DATE NOT NULL ,
    feedback_comment VARCHAR2 (2000)
)
LOGGING
;
ALTER TABLE SSD_FEEDBACK
    ADD CONSTRAINT CHK_FB_CMT_REQ
    CHECK (NOT (rating <= 2 AND feedback_comment IS NULL))
;

ALTER TABLE SSD_FEEDBACK
    ADD CONSTRAINT CHK_FB_RATING
    CHECK (rating BETWEEN 1 AND 5)
;
ALTER TABLE SSD_FEEDBACK
    ADD CONSTRAINT PK_SSD_FEEDBACK PRIMARY KEY ( account_id ) ;

CREATE TABLE SSD_LANGUAGE
(
    language_code  VARCHAR2 (10) NOT NULL ,
    country_name   VARCHAR2 (50) NOT NULL
)
LOGGING
;
ALTER TABLE SSD_LANGUAGE
    ADD CONSTRAINT PK_SSD_LANGUAGE PRIMARY KEY ( language_code ) ;

ALTER TABLE SSD_LANGUAGE
    ADD CONSTRAINT UQ_SSD_LANG_COUNTRY_NAME UNIQUE ( country_name ) ;

CREATE TABLE SSD_PRODUCER
(

```

```

        producer_id      NUMBER  NOT NULL ,
        producer_name    VARCHAR2 (100) NOT NULL ,
        phone_number     VARCHAR2 (20)  NOT NULL ,
        email_id         VARCHAR2 (120) NOT NULL ,
        street_address   VARCHAR2 (50)  NOT NULL ,
        city              VARCHAR2 (50)  NOT NULL ,
        state             VARCHAR2 (50)  NOT NULL ,
        zipcode          VARCHAR2 (50)  NOT NULL
    )
    LOGGING
;

ALTER TABLE SSD_PRODUCER
    ADD CONSTRAINT PK_SSD_PRODUCER PRIMARY KEY ( producer_id ) ;

CREATE TABLE SSD_PRODUCER_HOUSE
(
    producer_id      NUMBER  NOT NULL ,
    prod_house_id    VARCHAR2 (20) NOT NULL ,
    ws_id             VARCHAR2 (20) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_PRODUCER_HOUSE
    ADD CONSTRAINT PK_SSD_PROD_HOUSE_BR PRIMARY KEY ( producer_id,
prod_house_id, ws_id ) ;

CREATE TABLE SSD_PRODUCTION_HOUSE
(
    production_house_id VARCHAR2 (20) NOT NULL ,
    prod_house_name    VARCHAR2 (120) NOT NULL ,
    year_established  NUMBER  NOT NULL ,
    street_address     VARCHAR2 (100) NOT NULL ,
    city               VARCHAR2 (20)  NOT NULL ,
    state              VARCHAR2 (20)  NOT NULL ,
    zipcode            NUMBER  NOT NULL ,
    web_series_id_ref VARCHAR2 (20) NOT NULL ,
    contract_id_ref   VARCHAR2 (20) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_PRODUCTION_HOUSE

```

```

        ADD CONSTRAINT PK_SSD_PROD_HOUSE PRIMARY KEY (
production_house_id, web_series_id_ref ) ;

ALTER TABLE SSD_PRODUCTION_HOUSE
        ADD CONSTRAINT UQ_SSD_PH_NAME UNIQUE ( prod_house_name ) ;

CREATE TABLE SSD_SERIES_TYPE
(
    series_type_id VARCHAR2 (20) NOT NULL ,
    type_name      VARCHAR2 (50) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_SERIES_TYPE
        ADD CONSTRAINT PK_SSD_SERIES_TYPE PRIMARY KEY ( series_type_id )
;

ALTER TABLE SSD_SERIES_TYPE
        ADD CONSTRAINT UQ_SSD_SERIES_TYPE_NAME UNIQUE ( type_name ) ;

CREATE TABLE SSD_SUBSCRIPTION
(
    subscription_id VARCHAR2 (20) NOT NULL ,
    account_id     VARCHAR2 (20) NOT NULL ,
    plan_type      VARCHAR2 (20) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_SUBSCRIPTION
        ADD CONSTRAINT CHK_SUB_PLAN_TYPE
        CHECK (plan_type IN ('Basic','Standard','Premium'))
;
ALTER TABLE SSD_SUBSCRIPTION
        ADD CONSTRAINT PK_SSD_SUB PRIMARY KEY ( subscription_id ) ;

CREATE TABLE SSD_VIEWER_ACCOUNT
(
    account_id      VARCHAR2 (20) NOT NULL ,
    name            VARCHAR2 (120) NOT NULL ,
    street          VARCHAR2 (50) NOT NULL ,
    city            VARCHAR2 (20) NOT NULL ,
    state           VARCHAR2 (20) NOT NULL ,
    zipcode         NUMBER NOT NULL ,

```

```

        country_code          CHAR (3) NOT NULL ,
        opened_date           DATE    NOT NULL ,
        monthly_service_charge NUMBER (6,2) NOT NULL ,
        feedback_account_id  VARCHAR2 (20) NOT NULL ,
        viewer_age            NUMBER (3)
    )
    LOGGING
;

ALTER TABLE SSD_VIEWER_ACCOUNT
    ADD CONSTRAINT CHK_VA_MIN_AGE
    CHECK (viewer_age >= 13)
;

ALTER TABLE SSD_VIEWER_ACCOUNT
    ADD CONSTRAINT CHK_VA_MSC_POS
    CHECK (monthly_service_charge > 0)
;
ALTER TABLE SSD_VIEWER_ACCOUNT
    ADD CONSTRAINT PK_SSD_VIEWER_ACCOUNT PRIMARY KEY ( account_id ) ;

CREATE TABLE SSD_WEB_SERIES
(
    web_series_id      VARCHAR2 (20) NOT NULL ,
    name               VARCHAR2 (50) NOT NULL ,
    no_of_episodes     NUMBER NOT NULL ,
    language           VARCHAR2 (50) NOT NULL ,
    release_date       DATE    NOT NULL ,
    description         VARCHAR2 (50) ,
    contract_id_ref   VARCHAR2 (20) NOT NULL ,
    episode_id_ref     VARCHAR2 (20) NOT NULL ,
    feedback_account_ref VARCHAR2 (20) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_WEB_SERIES
    ADD CONSTRAINT PK_SSD_WEB_SERIES PRIMARY KEY ( web_series_id ) ;

CREATE TABLE SSD_WEB_SERIES_SUBTITLE
(
    web_series_id VARCHAR2 (20) NOT NULL ,
    language_code VARCHAR2 (10) NOT NULL
)

```

```
LOGGING
;

ALTER TABLE SSD_WEB_SERIES_SUBTITLE
    ADD CONSTRAINT PK_WS_SUBTITLE PRIMARY KEY ( web_series_id,
language_code ) ;

CREATE TABLE SSD_WEB_SERIES_TYPE
(
    web_series_id VARCHAR2 (20) NOT NULL ,
    series_type_id VARCHAR2 (20) NOT NULL
)
LOGGING
;

ALTER TABLE SSD_WEB_SERIES_TYPE
    ADD CONSTRAINT PK_WS_TYPE PRIMARY KEY ( web_series_id,
series_type_id ) ;

ALTER TABLE SSD_COUNTRY
    ADD CONSTRAINT FK_COUNTRY_VIEWER FOREIGN KEY
(
    viewer_account_id
)
REFERENCES SSD_VIEWER_ACCOUNT
(
    account_id
)
NOT DEFERRABLE
;

ALTER TABLE SSD_CONTRACT
    ADD CONSTRAINT FK_CT_WS FOREIGN KEY
(
    web_series_id
)
REFERENCES SSD_WEB_SERIES
(
    web_series_id
)
NOT DEFERRABLE
;

ALTER TABLE SSD_EPISODE
    ADD CONSTRAINT FK_EP_WS FOREIGN KEY
```

```
(  
    web_series_id  
)  
REFERENCES SSD_WEB_SERIES  
(  
    web_series_id  
)  
NOT DEFERRABLE  
;
```

```
ALTER TABLE SSD_FEEDBACK  
ADD CONSTRAINT FK_FB_WS FOREIGN KEY  
(  
    web_series_id  
)  
REFERENCES SSD_WEB_SERIES  
(  
    web_series_id  
)  
NOT DEFERRABLE  
;
```

```
ALTER TABLE SSD_PRODUCER_HOUSE  
ADD CONSTRAINT FK_PH_BR_PH FOREIGN KEY  
(  
    prod_house_id,  
    ws_id  
)  
REFERENCES SSD_PRODUCTION_HOUSE  
(  
    production_house_id,  
    web_series_id_ref  
)  
NOT DEFERRABLE  
;
```

```
ALTER TABLE SSD_PRODUCER_HOUSE  
ADD CONSTRAINT FK_PH_BR_PROD FOREIGN KEY  
(  
    producer_id  
)  
REFERENCES SSD_PRODUCER  
(  
    producer_id  
)
```

```
NOT DEFERRABLE
;

ALTER TABLE SSD_PRODUCTION_HOUSE
ADD CONSTRAINT FK_PH_CT FOREIGN KEY
(
    contract_id_ref
)
REFERENCES SSD_CONTRACT
(
    contract_id
)
NOT DEFERRABLE
;

ALTER TABLE SSD_PRODUCTION_HOUSE
ADD CONSTRAINT FK_PH_WS FOREIGN KEY
(
    web_series_id_ref
)
REFERENCES SSD_WEB_SERIES
(
    web_series_id
)
NOT DEFERRABLE
;

ALTER TABLE SSD_SUBSCRIPTION
ADD CONSTRAINT FK_SUB_ACC FOREIGN KEY
(
    account_id
)
REFERENCES SSD_VIEWER_ACCOUNT
(
    account_id
)
NOT DEFERRABLE
;

ALTER TABLE SSD_VIEWER_ACCOUNT
ADD CONSTRAINT FK_VA_COUNTRY FOREIGN KEY
(
    country_code
)
REFERENCES SSD_COUNTRY
```

```
(  
    country_code  
)  
NOT DEFERRABLE  
;  
  
ALTER TABLE SSD_VIEWER_ACCOUNT  
    ADD CONSTRAINT FK_VA_FEEDBACK FOREIGN KEY  
    (  
        feedback_account_id  
)  
    REFERENCES SSD_FEEDBACK  
    (  
        account_id  
)  
NOT DEFERRABLE  
;  
  
ALTER TABLE SSD_WEB_SERIES  
    ADD CONSTRAINT FK_WS_CTRREF FOREIGN KEY  
    (  
        contract_id_ref  
)  
    REFERENCES SSD_CONTRACT  
    (  
        contract_id  
)  
NOT DEFERRABLE  
;  
  
ALTER TABLE SSD_WEB_SERIES  
    ADD CONSTRAINT FK_WS_EPREF FOREIGN KEY  
    (  
        episode_id_ref  
)  
    REFERENCES SSD_EPISODE  
    (  
        episode_id  
)  
NOT DEFERRABLE  
;  
  
ALTER TABLE SSD_WEB_SERIES  
    ADD CONSTRAINT FK_WS_FBREF FOREIGN KEY  
    (  
        
```

```
    feedback_account_ref
)
REFERENCES SSD_FEEDBACK
(
    account_id
)
NOT DEFERRABLE
;

ALTER TABLE SR_WEB_SERIES_COUNTRY_FK
ADD CONSTRAINT FK_WSCTRY_COUNTRY FOREIGN KEY
(
    country_code
)
REFERENCES SSD_COUNTRY
(
    country_code
)
NOT DEFERRABLE
;

ALTER TABLE SR_WEB_SERIES_COUNTRY_FK
ADD CONSTRAINT FK_WSCTRY_WS FOREIGN KEY
(
    web_series_id
)
REFERENCES SSD_WEB_SERIES
(
    web_series_id
)
NOT DEFERRABLE
;

ALTER TABLE SSD_WEB_SERIES_SUBTITLE
ADD CONSTRAINT FK_WSSUB_LANG FOREIGN KEY
(
    language_code
)
REFERENCES SSD_LANGUAGE
(
    language_code
)
NOT DEFERRABLE
;
```

```

ALTER TABLE SSD_WEB_SERIES_SUBTITLE
    ADD CONSTRAINT FK_WSSUB_WS FOREIGN KEY
    (
        web_series_id
    )
    REFERENCES SSD_WEB_SERIES
    (
        web_series_id
    )
    NOT DEFERRABLE
;

ALTER TABLE SSD_WEB_SERIES_TYPE
    ADD CONSTRAINT FK_WSTYPE_TYPE FOREIGN KEY
    (
        series_type_id
    )
    REFERENCES SSD_SERIES_TYPE
    (
        series_type_id
    )
    NOT DEFERRABLE
;

ALTER TABLE SSD_WEB_SERIES_TYPE
    ADD CONSTRAINT FK_WSTYPE_WS FOREIGN KEY
    (
        web_series_id
    )
    REFERENCES SSD_WEB_SERIES
    (
        web_series_id
    )
    NOT DEFERRABLE
;

CREATE OR REPLACE TRIGGER TRG_NXFER_WS_SUB
    BEFORE UPDATE OF web_series_id ON SSD_WEB_SERIES_SUBTITLE
BEGIN
    RAISE_APPLICATION_ERROR(
        -20225,
        'Non Transferable FK constraint on table SSD_WEB_SERIES_SUBTITLE
is violated'
    );
END;

```

/

```
-- Oracle SQL Developer Data Modeler Summary Report:  
--  
-- CREATE TABLE 15  
-- CREATE INDEX 0  
-- ALTER TABLE 49  
-- CREATE VIEW 0  
-- ALTER VIEW 0  
-- CREATE PACKAGE 0  
-- CREATE PACKAGE BODY 0  
-- CREATE PROCEDURE 0  
-- CREATE FUNCTION 0  
-- CREATE TRIGGER 1  
-- ALTER TRIGGER 0  
-- CREATE COLLECTION TYPE 0  
-- CREATE STRUCTURED TYPE 0  
-- CREATE STRUCTURED TYPE BODY 0  
-- CREATE CLUSTER 0  
-- CREATE CONTEXT 0  
-- CREATE DATABASE 0  
-- CREATE DIMENSION 0  
-- CREATE DIRECTORY 0  
-- CREATE DISK GROUP 0  
-- CREATE ROLE 0  
-- CREATE ROLLBACK SEGMENT 0  
-- CREATE SEQUENCE 0  
-- CREATE MATERIALIZED VIEW 0  
-- CREATE MATERIALIZED VIEW LOG 0  
-- CREATE SYNONYM 0  
-- CREATE TABLESPACE 0  
-- CREATE USER 0  
--  
-- DROP TABLESPACE 0  
-- DROP DATABASE 0  
--  
-- REDACTION POLICY 0  
--  
-- ORDS DROP SCHEMA 0  
-- ORDS ENABLE SCHEMA 0  
-- ORDS ENABLE OBJECT 0  
--  
-- ERRORS 0
```

I) CONCLUSION:

The **Netflix Episodes Web System (NEWS)** project successfully designed and implemented a fully normalized relational database that supports efficient management of web series, seasons, episodes, viewers, and production data. Using a systematic design approach—from the logical ERD to the physical relational model—the database ensures data integrity, scalability, and minimal redundancy.

All core business rules were enforced through **primary/foreign keys** and **CHECK constraints**, while **DML scripts** populated the schema with valid sample records for testing. The **Data Dictionary** further documented every entity and attribute, ensuring maintainability and clarity.