**View Creation for Use Cases**

1. How many total courses are available for Information Systems Program in Northeastern University?

CREATE VIEW InformationSystems\_Total\_Courses AS

SELECT a.program\_name,

COUNT(a.course\_name), b.program\_url

FROM neu\_course\_catalog a

LEFT OUTER JOIN

NEU\_Program b

ON a.program\_name = b.program\_name

WHERE b.program\_name = 'Information Systems'

GROUP BY a.program\_name;

1. Which courses do we need to take to do specialization in User Experience Concentration?

CREATE VIEW User\_Experience\_Specialization\_Courses AS

SELECT a.specialization\_name,b.course\_id,c.course\_name

FROM

neu\_specialization a

LEFT JOIN neu\_course\_specialization b

ON a.specialization\_id = b.specialization\_id

RIGHT JOIN neu\_course\_catalog c

ON b.course\_id = c.course\_id

WHERE a.specialization\_name = 'User Experience Concentration';

1. List top rated professors at NEU whose rating is above 4. Also mention the courses and its description taught by the professor.

CREATE VIEW NEU\_Top\_Rated\_Professors AS

SELECT a.faculty\_name,a.faculty\_rating,b.course\_id,c.course\_description

FROM neu\_facultys a

JOIN neu\_course\_faculty b

ON a.faculty\_name = b.faculty\_name

JOIN neu\_course\_catalog c

ON b.course\_id = c.course\_id

WHERE a.faculty\_rating >= 4;

1. What are the core course requirements for Information Systems, Software Engineering Systems and Data Analytics?

CREATE VIEW Programs\_Core\_Course\_requirement AS

SELECT a.program\_name, GROUP\_CONCAT(a.course\_id SEPARATOR '|'),

COUNT(a.course\_id), b.program\_url

FROM Course\_Core\_requirement a

LEFT JOIN NEU\_Program b

ON a.program\_name = b.program\_name

GROUP BY a.program\_name;

1. List courses available for Data Analytics Engineering along with its course\_outcome.

CREATE VIEW Data\_Analytics\_course\_outcome AS

SELECT

program\_name, course\_id,course\_name, course\_description

FROM

NEU\_Course\_Catalog

WHERE program\_name = 'Data Analytics Engineering';

### What is the average overall rating given by students to a professor for a given course?

Create VIEW Professor\_Avg\_rating AS

SELECT faculty\_name, round (AVG(faculty\_rating),1) as Average\_rating

FROM neu\_Facultys

GROUP BY faculty\_name;

1. Which courses students need to take to do specialization in Big Data Systems concentration and which resources are available for them?

CREATE VIEW Big\_data\_Course\_Resource AS

SELECT a.specialization\_name, b.course\_id,c.course\_name,d.software\_name,e.online\_platform

FROM neu\_specialization a

LEFT JOIN neu\_course\_specialization b

ON a.specialization\_id = b.specialization\_id

LEFT JOIN neu\_course\_catalog c

ON b.course\_id = c.course\_id

LEFT JOIN neu\_course\_resource d

ON c.course\_id = d.course\_id

LEFT JOIN neu\_resource\_materials e

ON d.software\_name = e.software\_name

WHERE a.specialization\_name like '%Big Data%';

1. Which job positions are available for students in Data Analytics Engineering and list the details related to those positions?

CREATE VIEW Data\_Analytics\_Job\_Positions AS

SELECT b.program\_name, a.job\_id,a.title,a.company\_name,a.description,a.location

FROM program\_jobs b

LEFT JOIN job\_info a

ON a.job\_id = b.job\_id

WHERE b.program\_name LIKE '%Data%';

1. Which professors teach courses related to User Experience and list the ratings given by students for the course?

CREATE VIEW User\_Experience\_Professors AS

SELECT a.specialization\_name, b.course\_id,c.program\_name, d.faculty\_name,e.faculty\_rating

FROM neu\_specialization a

LEFT JOIN neu\_course\_specialization b

ON a.specialization\_id = b.specialization\_id

LEFT JOIN neu\_course\_catalog c

ON b.course\_id = c.course\_id

LEFT JOIN neu\_course\_faculty d

ON d.course\_id = c.course\_id

LEFT JOIN neu\_facultys e

ON e.faculty\_name = d.faculty\_name

WHERE a.specialization\_name like '%User%';

1. Find the average course rating for Information Systems Program

CREATE VIEW Information\_System\_Avg\_Course\_rating AS

SELECT a.program\_name, AVG(b.course\_rating)

FROM neu\_course\_catalog a

RIGHT JOIN neu\_course\_resource b

ON a.course\_id = b.course\_id

WHERE a.program\_name ='Information Systems';

1. List courses with lowest rating.

CREATE VIEW Lowest\_Rating\_Courses As

SELECT a.course\_id, a.course\_name,a.course\_description,b.course\_rating

FROM neu\_course\_catalog a

LEFT JOIN neu\_course\_resource b

ON a.course\_id = b.course\_id

WHERE b.course\_rating = (select min(course\_rating) from neu\_course\_resource);

1. . Find the professors who taught DAMG 7350, INFO 5100, DAMG 6105, DAMG 7275 courses.

CREATE VIEW Professor\_DAMG7350\_INFO5100\_DAMG6105\_DAMG7275 AS

select a.faculty\_name, b.course\_id

from neu\_facultys a

JOIN NEU\_Course\_faculty b

ON a.faculty\_name = b.faculty\_name

where b.course\_id IN ('DAMG 7350', 'INFO 5100','DAMG 6105' ,'DAMG 7275');

1. Which resources are available for students who have chosen Information Systems

program?

CREATE VIEW Information\_Systems\_Resource\_Details AS

SELECT a.program\_name,a.course\_id,b.software\_name,

c.software\_download\_url,c.online\_platform

FROM neu\_course\_catalog a

JOIN neu\_course\_resource b

ON a.course\_id = b.course\_id

JOIN neu\_resource\_materials c

ON b.software\_name = c.software\_name

WHERE a.program\_name = 'Information Systems';