

Write the following queries in relational algebra, using the university schema. (Fig. 2.8)

- a. Find the ID and name of each instructor in the Physics department.

$\Pi \text{ id, name } (\sigma \text{ dept_name} = \text{'Physics'} (\text{instructor}))$

query selects all rows from instructor relation where the dept_name is "Physics". Displays only id and name from result

- b. Find the ID and name of each instructor in a department located in the building "Watson".

$\Pi \text{ id, name } (\sigma \text{ building} = \text{'Watson'} (\text{instructor} \times \text{department}))$

Query perform,s natural join/ cartesian product (x) between the "instructor " and "department" based on "dept_name"

Then query selects all rows from result where building is equal to Watson and displays id and name

- c. Find the ID and name of each student who has taken at least one course in the "Comp. Sci." department.

$\Pi \text{ id, name } (\sigma \text{ dept_name} = \text{'Computer science'} (\text{student} \times \text{takes}))$

Cartesian product between students and takes based on their common attribute "ID"

Selects all rows from result where dept_name attribute is equal to 'Computer Science'

- d. Find the ID and name of each student who has taken at least one course section in the year 2018.

$\Pi \text{ id, name } (\sigma \text{ year} = 2018 (\text{student} \times \text{takes}))$

Cartesian product between student and takes with common ID, selects all rows where year is equal to 2018

- e. Find the ID and name of each student who has not taken any course section in the year 2018.

$\Pi \text{ id, name } (\text{takes} - \sigma \text{ year} = 2018 (\text{student} \times \text{takes}))$

Cartesian product between student and takes to find all students who took courses

Then query selects all rows from student and subtracts the result of previous join where year is equal to 2018

Subtract gives students who have not taken and displays id and name