**Ben Waddell**

**CPSC 5130 – Assignment 1**

**Question 1:**

1. SELECT a.firstname, a.lastname, m.movie\_title

FROM actors a

LEFT JOIN movie\_cast mc ON a.actor\_id = mc.actor\_id

JOIN movies m ON mc.movie\_id = m.movie\_id;

1. SELECT d.firstname, d.lastname, c.city, c.state

FROM directors d

LEFT JOIN cities c ON d.city\_id = c.city\_id;

1. SELECT a.firstname, a.lastname, m.movie\_title, m.international\_sales

FROM actors a

JOIN movie\_cast mc ON a.actor\_id = mc.actor\_id

JOIN movies m ON mc.movie\_id = m.movie\_id

UNION

SELECT d.firstname, d.lastname, m.movie\_title, m.international\_sales

FROM directors d

JOIN movies m ON d.director\_id = m.director\_id

ORDER BY international\_sales;

**Question 2:**

Cross join returns all combinations of each row of one table with each row of another table. This is useful if you want to get all possible combinations of the rows between tables. This differs from a natural join in that the cross join is finding all possible combinations of tables, whereas a natural join combines attributes of tables based on common matching columns.

For example, assume we have the following tables:

|  |  |
| --- | --- |
| **students** |  |
| **student\_id** | **student\_name** |
| 001 | Ben |
| 002 | Anna |
| 003 | James |

|  |  |
| --- | --- |
| **professors** |  |
| **course\_number** | **professor\_name** |
| CPSC5120 | David |
| MATH1000 | Ryan |
| CPSC5500 | Chris |

A **cross join** returns the following result:

|  |  |  |  |
| --- | --- | --- | --- |
| **student\_id** | **student\_name** | **course\_number** | **professor\_name** |
| 001 | Ben | CPSC5120 | David |
| 001 | Ben | MATH1000 | Ryan |
| 001 | Ben | CPSC5500 | Chris |
| 001 | Anna | CPSC5120 | David |
| 002 | Anna | MATH1000 | Ryan |
| 002 | Anna | CPSC5500 | Chris |
| 003 | James | CPSC5120 | David |
| 003 | James | MATH1000 | Ryan |
| 003 | James | CPSC5500 | Chris |

In this example, there are no matching columns, so the tables would not be joined in a **natural join**.

**Question 3:**

A natural join returns all attributes of tables based on common matching columns, excluding duplicate attributes. This is different from an inner join in that inner join will join tables on the basis on a matching column that is explicitly specified in the ON statement. Both joins will result in output that contains all attributes of both tables, but the natural join will remove the duplicate columns, whereas the inner join will include them.

For example, assume we have the following tables:

|  |  |  |
| --- | --- | --- |
| **students** |  |  |
| **student\_id** | **student\_name** | **course\_number** |
| 001 | Ben | CPSC5500 |
| 002 | Anna | MATH1000 |
| 003 | James | CPSC5120 |

|  |  |
| --- | --- |
| **professors** |  |
| **course\_number** | **professor\_name** |
| CPSC5120 | David |
| MATH1000 | Ryan |
| CPSC5500 | Chris |

A **natural join** will “automatically” match based on any common columns, in this case course\_number resulting in:

|  |  |  |  |
| --- | --- | --- | --- |
| **student\_id** | **student\_name** | **course\_number** | **professor\_name** |
| 001 | Ben | CPSC5500 | Chris |
| 002 | Anna | MATH1000 | Ryan |
| 003 | James | CPSC5120 | David |

An **inner join** would join the two tables with an *ON students.course\_number = professors.course\_number* statement and result in the following:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **student\_id** | **student\_name** | **students.course\_number** | **professors.course\_number** | **professor\_name** |
| 001 | Ben | CPSC5500 | CPSC5500 | Chris |
| 002 | Anna | MATH1000 | MATH1000 | Ryan |
| 003 | James | CPSC5120 | CPSC5120 | David |