-- 1

SELECT

\*

FROM

naep;

-- 2

SELECT

\*

FROM

naep

LIMIT

50;

-- 3

SELECT

state,

AVG(avg\_math\_4\_score),

MAX(avg\_math\_4\_score),

MIN(avg\_math\_4\_score),

COUNT(avg\_math\_4\_score)

FROM

naep

GROUP BY

state

ORDER BY

state;

-- 4

SELECT

state,

AVG(avg\_math\_4\_score),

MAX(avg\_math\_4\_score),

MIN(avg\_math\_4\_score),

COUNT(avg\_math\_4\_score)

FROM

naep

GROUP BY

1

HAVING

MAX(avg\_math\_4\_score) - MIN(avg\_math\_4\_score) > 30

ORDER BY

1;

-- 5

SELECT

state AS bottom\_10\_states

FROM

naep

WHERE

year = 2000

ORDER BY

avg\_math\_4\_score DESC

LIMIT

10;

-- 6

SELECT

ROUND(AVG(avg\_math\_4\_score), 2)

FROM

naep

WHERE

year = 2000

GROUP BY

year;

-- 7

SELECT

state AS below\_average\_states\_y2000

FROM

naep

WHERE

(avg\_math\_4\_score <= (SELECT

AVG(avg\_math\_4\_score)

FROM

naep

WHERE

year = 2000

GROUP BY

year))

AND

year = 2000;

-- 8

SELECT

state AS scores\_missing\_y2000

FROM

naep

WHERE

year = 2000

AND

avg\_math\_4\_score IS NULL;

-- 9

WITH state\_math AS (

SELECT

id,

state,

avg\_math\_4\_score

FROM

naep

WHERE

year = 2000

AND

avg\_math\_4\_score IS NOT NULL)

SELECT

finance.id,

state\_math.state,

finance.total\_expenditure,

ROUND(avg\_math\_4\_score, 2)

FROM

state\_math

JOIN

finance

ON

state\_math.id = finance.id

ORDER BY 3 DESC;