a)

SELECT

AVG(Amount) AS Average

FROM

Budget

WHERE

BuID = 2 OR BuID = 4;

Graphical user interface, text, application

Description automatically generated

b)

SELECT

Staff.StaffID,

Staff.FirstName,

Staff.LastName,

Campus.CaID

FROM

Staff

INNER JOIN Campus ON Staff.CaID = Campus.CaID

WHERE

Staff.LeftD BETWEEN '2019-08-21 00:00:00' AND '2019-08-21 23:59:59';

Graphical user interface, text, application

Description automatically generated

c)

INSERT INTO Campus(

Address,

GmName,

Country,

STATUS

)

VALUES(

'112, Pineapple Avenue, New York',

'Michael Carlos',

'USA',

'Open Soon'

);

Graphical user interface, text, application

Description automatically generated

d)

SELECT

Title,

FirstName,

LastName

FROM

Staff

INNER JOIN Department ON Staff.DeptID = Department.DeptID

WHERE

DeptName = "EEECS";

Graphical user interface, text, application

Description automatically generated

e)

SELECT

\*

FROM

Staff

ORDER BY

LastName ASC

LIMIT 10;

Graphical user interface, application

Description automatically generated

f)

SELECT

COUNT(Allocation.AcID) AS NoOfAllocations,

FirstName,

LastName

FROM

Staff

INNER JOIN Allocation,

Activity

WHERE

Allocation.StaffID = Staff.StaffID AND Allocation.AcID = Activity.AcID

GROUP BY

FirstName,

LastName;

Graphical user interface, text, application

Description automatically generated

g)

SELECT

StudentName,

Address,

Enrolment.Score

FROM

Student

INNER JOIN Enrolment ON Student.StudentID = Enrolment.StudentID

INNER JOIN Module ON Enrolment.ModuleID = Module.ModuleID

WHERE

ModuleName = "Java Programming"

ORDER BY

Score

DESC

Graphical user interface, text, application, email

Description automatically generated

h)

SELECT

SUM(Amount) AS "Lisa's Spend"

FROM

Budget

WHERE

Approver = "Lisa Amy"

Graphical user interface, application

Description automatically generated

i)

SELECT

\*

FROM

Staff

WHERE

LastName LIKE "C%" AND CaID = 1;

Graphical user interface, text, application

Description automatically generated

j)

SELECT

StudentName,

ModuleName,

Semester

FROM

Student

INNER JOIN Enrolment ON Student.StudentID = Enrolment.StudentID

INNER JOIN Module ON Enrolment.ModuleID = Module.ModuleID

WHERE

Enrolment.Year = 2020

ORDER BY

Enrolment.Score

DESC

LIMIT 1, 1;

Graphical user interface, text, application

Description automatically generated