

SQL Assignment #01: Suggested Solutions

1	SELECT * FROM Members ORDER BY Surname, Forename;
2	SELECT * FROM Tickets ORDER BY Purch_Date DESC;
3	SELECT * FROM Tickets WHERE Purch_Date LIKE '%/2009';
4	SELECT * FROM Members WHERE Surname LIKE 'K%';
5	SELECT * FROM Members WHERE Surname LIKE '%th';
6	SELECT * FROM Members WHERE Surname LIKE '%oo%';
7	SELECT Ticket_No, Surname, Forename FROM Tickets T, Members M WHERE T.Mem_ID = M.Mem_ID;
8	SELECT P.Ticket_No, Prize_Amount, Surname, Forename FROM Prizes P, Tickets T, Members M WHERE P.Ticket_No = T.Ticket_No AND T.Mem_ID = M.Mem_ID;
9	SELECT COUNT(*) FROM Members;
10	SELECT MAX(Prize_Amount), MIN(Prize_Amount) FROM Prizes;
11	SELECT SUM(Prize_Amount) FROM Prizes;
12	SELECT AVG(Prize_Amount) FROM Prizes;

13	SELECT * FROM Prizes WHERE Prize_Amount > (SELECT AVG(Prize_Amount) FROM Prizes) ORDER BY Draw_Date ASC;
14	SELECT Ticket_No, COUNT(*) FROM Prizes GROUP BY Ticket_No;
15	SELECT Mem_ID, COUNT(*) FROM Tickets GROUP BY Mem_ID;
16	SELECT Draw_Date, SUM(Prize_Amount) FROM Prizes GROUP BY Draw_Date;
17	SELECT Draw_Date, SUM(Prize_Amount) FROM Prizes GROUP BY Draw_Date HAVING SUM(Prize_Amount) > 15000;
18	SELECT Ticket_No FROM Tickets T, Prizes P WHERE T.Ticket_No = P.Ticket_No(+) AND Prize_Amount IS NULL; OR SELECT Ticket_No FROM Tickets T LEFT JOIN Prizes P ON T.Ticket_No = P.Ticket_No WHERE Prize_Amount IS NULL;
19	SELECT * FROM Members WHERE Mem_ID NOT IN (SELECT DISTINCT Mem_ID FROM Tickets WHERE Ticket_No IN (SELECT DISTINCT Ticket_No FROM Prizes));
20	CREATE TABLE Prizes (Prize_ID numeric(6), Draw_Date Date, Ticket_No numeric(6), Prize_Amount numeric(8,2), CONSTRAINT pk_Prizes PRIMARY KEY (Prize_ID), CONSTRAINT fk_Prizes_Tickets FOREIGN KEY (Ticket_No) REFERENCES Tickets);

