

Query:	Code:	Opmerking:
<b>Uitslagen_verrijkt</b>	SELECT Uitslagen.*, CarTeMaken1+Db1*StrafCaramboles() AS CarTM1, CarTeMaken2+Db1*StrafCaramboles() AS CarTM2, IIF(Car1=CarTM1,Beurten) AS BeurtenAlsUit1, IIF(Car2=CarTM2,Beurten) AS BeurtenAlsUit2, PartijPunten(Car1,CarTM1,Car2,CarTM2,Beurten)+0 AS PartijP1, PartijPunten(Car2,CarTM2,Car1,CarTM1,Beurten)+0 AS PartijP2 FROM Uitslagen;	Tabel Uitslagen verrijkt met CarTeMaken1/2 (dat is met strafcaramboles), BeurtenAlsUit1/2 (dat is Beurten indien hij uit is [tbv kortste partij]), PartijPunten1/2.
<b>Uitslagen_verrijkt_DbIS</b>	SELECT SpelerID1 AS SpelerID, Uitslagen_verrijkt.* FROM Uitslagen_verrijkt UNION ALL SELECT SpelerID2, Uitslagen_verrijkt.* FROM Uitslagen_verrijkt;	Uitslagen_verrijkt verrijkt met SpelerID1 + Uitslagen_verrijkt verrijkt met SpelerID2
<b>Spelers_verrijkt</b>	SELECT Spelers.*, BasisDatum() AS Moyenne_LAMU, caramboles(Spelers.Moyenne)+0 AS CarBeg, caramboles(Spelers.MoyNaXpar)+0 AS CarNaXpar, caramboles(Spelers.MoyNaYpar)+0 AS CarNaYpar, caramboles(Spelers.MoyNaZpar)+0 AS CarNaZpar, caramboles(Spelers.MoyRonde2)+0 AS CarRonde2, caramboles(Spelers.MoyRonde3)+0 AS CarRonde3, caramboles(Spelers.MoyRonde4)+0 AS CarRonde4 FROM Spelers;	Tabel Spelers verrijkt met ingangsdatum Moyenne (= #01/01/1900#), alle bijbehorende caramboles.
<b>SpelersAggregate</b>	SELECT SpelerID, Count(*) AS AantalPar, Sum(Car) AS SomVanCar, Sum(Beurten) AS SomVanBeurten, INT(1000*SomVanCar/SomVanBeurten+0.5)/1000 AS Gem, Max(HS) AS MaxVanHS, Sum(PartijP) AS SomVanPartijP, Min(BeurtenAlsUit) AS MinBeurtenAlsUit FROM (SELECT SpelerID1 AS SpelerID, Car1 AS Car, Beurten, HS1 AS HS, PartijP1 AS PartijP, BeurtenAlsUit1 AS BeurtenAlsUit FROM Uitslagen_verrijkt UNION ALL SELECT SpelerID2, Car2, Beurten, HS2, PartijP2, BeurtenAlsUit2 FROM Uitslagen_verrijkt) AS Q GROUP BY SpelerID;	Rode tekst zijn de specifieke gegevens van SpelerID1 (maar met het algemene gegeven Beurten). Daaraan gekoppeld (via UNION ALL) soortgelijks van SpelerID2.  Daarna groeperen per SpelerID. Velden tellen, sommeren, maximum bepalen enz.
<b>SpelersOverzicht</b>	SELECT Teams.TeamNaam, Spelers_verrijkt.*, SpelersAggregate.* FROM Teams INNER JOIN (Spelers_verrijkt LEFT JOIN SpelersAggregate ON Spelers_verrijkt.ID = SpelersAggregate.SpelerID) ON Teams.ID = Spelers_verrijkt.TeamID.Value;	Spelers_verrijkt gekoppeld aan SpelersAggregate. Zo weten we de totaalgegevens en de basisgegevens zoals Moyenne en CarBeg. Ook spelers die niet in SpelersAggregate zitten worden getoond (LEFT JOIN). Alleen Actieve spelers.  Het geheel wordt aan tabel Teams gekoppeld om de teamnaam te bepalen (ivm sorteren).
<b>Create_TempGespGem</b>	SELECT * INTO TempGespGem FROM (SELECT SpelerID AS ID, MaxOfList(Gem,0.001)+0 AS Gemid, AantalPar FROM SpelersAggregate) AS Q;	SpelerID, Gemid, AantalPar overzetten naar een tijdelijke tabel. Gemid mag niet lager worden dan 0.001. Query tbv moyenne herberekening.
<b>MoyUpdates</b>	SELECT * FROM SpelersOverzicht WHERE MoyNaXpar_LAMU is not null or MoyNaYpar_LAMU is not null or MoyNaZpar_LAMU is not null;	Subset van SpelersOverzicht.
<b>MoyUpdates_LW</b>	SELECT * FROM MoyUpdates WHERE MoyNaXpar_LAMU>=StartLastFullPlayWeek() OR MoyNaYpar_LAMU>=StartLastFullPlayWeek() OR MoyNaZpar_LAMU>=StartLastFullPlayWeek();	MoyUpdates sinds begin van laatste hele speelweek.
<b>SpelersOverzicht_AlleenGesp</b>	SELECT * FROM SpelersOverzicht WHERE AantalPar Is Not Null;	SpelersOverzicht alleen van spelers die daadwerkelijk hebben gespeeld.
<b>WedstrMetPnt</b>	SELECT Wedstrijden.ID, Wedstrijden.Ronde, Wedstrijden.Datum, Wedstrijden.TeamID1, Wedstrijden.TeamID2, SumOfList(Y.SomVanPartijP1,Wedstrijden.BP1)+0 AS WP1, SumOfList(Y.SomVanPartijP2,Wedstrijden.BP2)+0 AS WP2, Wedstrijden.BP1 Is Not Null Or Wedstrijden.BP2 Is Not Null AS BP FROM Wedstrijden LEFT JOIN (SELECT Uitslagen_verrijkt.WedID, Sum(Uitslagen_verrijkt.PartijP1) AS SomVanPartijP1, Sum(Uitslagen_verrijkt.PartijP2) AS SomVanPartijP2 FROM Uitslagen_verrijkt GROUP BY Uitslagen_verrijkt.WedID) AS Y ON Wedstrijden.ID = Y.WedID WHERE Wedstrijden.TeamID1 In (SELECT ID FROM Teams WHERE Actief) AND Wedstrijden.TeamID2 In (SELECT ID FROM Teams WHERE Actief) AND NOT Wedstrijden.Vervallen;	Sommeer de PartijPunten van de partijen per WedID (voor thuis en uit). Tel de wedstrijdbonuspunten erbij op. SumOfList lost het probleem met Null waarden op.  Zelfs wedstrijden die niet voor komen in Uitslagen_verrijkt worden getoond (LEFT JOIN).  Alleen wedstrijden waarbij thuis- en uitteam actief zijn en de wedstrijd niet vervallen is.
<b>WedstrMetPnt_DbIT</b>	SELECT TeamID1 AS TeamID, WedstrMetPnt.* FROM WedstrMetPnt UNION ALL SELECT TeamID2, WedstrMetPnt.* FROM WedstrMetPnt;	WedstrMetPnt verrijkt met TeamID1 + WedstrMetPnt verrijkt met TeamID2

Query:	Code:	Opmerking:
<b>Wedstrijden_details</b>	SELECT WedstrMetPnt.*, Uitslagen_verrijkt.* FROM WedstrMetPnt LEFT JOIN Uitslagen_verrijkt ON WedstrMetPnt.ID = Uitslagen_verrijkt.WedID;	WedstrMetPnt gekoppeld aan Uitslagen_verrijkt. Zelfs wedstrijden die niet voor komen in Uitslagen_verrijkt worden getoond (LEFT JOIN).
<b>Wedstrijden_details_LW</b>	SELECT * FROM Wedstrijden_details WHERE Datum Between StartLastFullPlayWeek() And Date();	Wedstrijden_details sinds begin van laatste hele speelweek t/m vandaag.
<b>WedstrPerTeam</b>	SELECT * FROM Teams INNER JOIN WedstrMetPnt_DblT ON Teams.ID = WedstrMetPnt_DblT.TeamID;	WedstrMetPnt_DblT gekoppeld aan Teams om de TeamNaam op te zoeken (ivm sorteren).
<b>WedstrPnt_Team</b>	SELECT X.TeamID, Count(X.WP) AS Aantal, Sum(X.WP) AS WPtot FROM (SELECT TeamID1 AS TeamID, WP1 AS WP FROM WedstrMetPnt UNION ALL SELECT TeamID2, WP2 FROM WedstrMetPnt) AS X WHERE WP Is Not Null GROUP BY X.TeamID;	TeamID1 en WP1 koppelen aan TeamID2 en WP2. Echter alleen indien het WP Not Null is (nog geen uitslag). Groeperen per TeamID. WP sommeren zodat ik de totale wedstrijdpunten weet.
<b>Spelers_details</b>	SELECT * FROM Teams INNER JOIN (WedstrMetPnt INNER JOIN (Spelers INNER JOIN Uitslagen_verrijkt_DblS ON Spelers.ID = Uitslagen_verrijkt_DblS.SpelerID) ON WedstrMetPnt.ID = Uitslagen_verrijkt_DblS.WedID) ON Teams.ID = Spelers.TeamID.Value;	WedstrMetPnt gekoppeld aan Uitslagen_verrijkt_DblS. Ook gekoppeld aan Teams, Spelers om de Teamnaam en SpelerNaam te bepalen (ivm sorteren).  Alleen actieve spelers.
<b>Hlp_Invoer_Check</b>	SELECT Nr, SpelerID_1, GespCar_1, CarTeMaken_1+DbI_1*StrafCaramboles() AS CarTeMaken_1_Dbl, HS_1, SpelerID_2, GespCar_2, CarTeMaken_2+DbI_2*StrafCaramboles() AS CarTeMaken_2_Dbl, HS_2, Beurten FROM Hlp_Invoer;	Tabel Hlp_Invoer maar dan de te maken caramboles verhoogd met strafcaramboles.
<b>Hlp_XLSinvoer</b>	SELECT XLS_Beurten.Nr, XLS_Team1.Dbl, Spelers.ID AS SpelerID1, XLS_Team1.[Gesp Car], XLS_Team1.[H Serie], XLS_Team2.Dbl, Spelers_1.ID AS SpelerID2, XLS_Team2.[Gesp Car], XLS_Team2.[H Serie], XLS_Beurten.Brt FROM (((XLS_Team2 INNER JOIN (XLS_Beurten INNER JOIN XLS_Team1 ON XLS_Beurten.Nr = XLS_Team1.Nr) ON XLS_Team2.Nr = XLS_Beurten.Nr) INNER JOIN Spelers ON XLS_Team1.Naam = Spelers.SpelerNaam) INNER JOIN Spelers AS Spelers_1 ON XLS_Team2.Naam = Spelers_1.SpelerNaam) INNER JOIN Teams AS Teams_1 ON Spelers_1.TeamID.Value = Teams_1.ID) INNER JOIN (Teams INNER JOIN XLS_Wedstrijd ON Teams.TeamNaam = XLS_Wedstrijd.ThuisClub) ON (Teams_1.TeamNaam = XLS_Wedstrijd.UitClub) AND (Teams.ID = Spelers.TeamID.Value) WHERE XLS_Team1.Verv=0 AND XLS_Team1.Dbl>=0 And XLS_Team1.Dbl<=2 AND XLS_Team1.[Gesp Car]>=0 And XLS_Team1.[Gesp Car]<=XLS_Team1.[H Serie]*XLS_Beurten.Brt AND XLS_Team1.[H Serie]>=0 And XLS_Team1.[H Serie]<=XLS_Team1.[Gesp Car] AND XLS_Team2.Dbl>=0 And XLS_Team2.Dbl<=2 AND XLS_Team2.[Gesp Car]>=0 And XLS_Team2.[Gesp Car]<=XLS_Team2.[H Serie]*XLS_Beurten.Brt AND XLS_Team2.[H Serie]>=0 And XLS_Team2.[H Serie]<=XLS_Team2.[Gesp Car] AND XLS_Beurten.Brt>0;	Controle voor goede records in aangeleverde XLS-bestand. Aanpassen voor elke klant.
<b>Hlp_XLSinvoer2</b>	SELECT XLS_Team1.Verv, XLS_Beurten.Nr, XLS_Team1.Dbl, XLS_Team1.[Gesp Car], XLS_Team1.[H Serie], XLS_Team2.Dbl, XLS_Team2.[Gesp Car], XLS_Team2.[H Serie], XLS_Beurten.Brt, XLS_Team1.Naam, XLS_Team2.Naam FROM XLS_Team2 INNER JOIN (XLS_Beurten INNER JOIN XLS_Team1 ON XLS_Beurten.Nr = XLS_Team1.Nr) ON XLS_Team2.Nr = XLS_Beurten.Nr WHERE XLS_Team1.Verv=1 AND (XLS_Team1.Dbl Is Null Or XLS_Team1.Dbl=0) AND (XLS_Team2.Dbl Is Null Or XLS_Team2.Dbl=0) AND XLS_Team1.[Gesp Car] Is Null AND XLS_Team1.[H Serie] Is Null AND XLS_Team1.Naam Is Null AND XLS_Team2.[Gesp Car] Is Null AND XLS_Team2.[H Serie] Is Null AND XLS_Team2.Naam Is Null AND XLS_Beurten.Brt Is Null;	Controle voor goede records in aangeleverde XLS-bestand (vervallen partijen). Aanpassen voor elke klant.
<b>NietSpeelweken</b>	PARAMETERS NietSpelen DateTime; UPDATE Wedstrijden SET Datum = Datum+7 WHERE Datum>=[NietSpelen];	Schuif wedstrijddatum 7 dagen op. Doe dit vanaf [NietSpelen]