

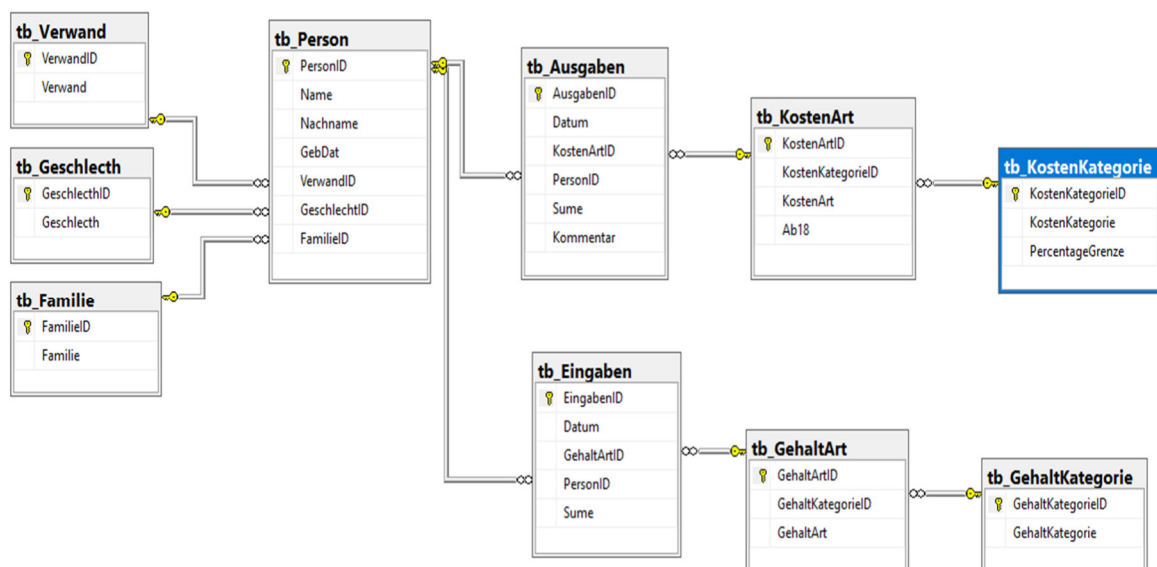
Projektbeschreibung Gruppe C: „Fameliie-Budget“ (HomeDBD2)

28 Okt. 2022

- Pooja Vinayak A.
- Natalie A.
- Mengyang T.

Projektlogik: (**Rot** = eigene Tabelle)

1. Jede Familie hat einen Budgetplan für einen Monat.
2. Folgende **Person** daten werden erfasst: **Name**, **Vorname**, **GebDat**, **VerwandID**, **GeschlechtID**, **FamilielD**.
3. Die Fameliie-Budgets Planung kann aus mehreren **Familien** bestehen und ist nach **Geschlecht** und **Verwand** aufgeteilt.
4. Innerhalb einer Familie gibt es zum bestimmten **Datum** und **Summe** von unterschiedlichen **Eingaben den Personen**. Neben ihrer **Funktion** werden auch **Summe** in verschiedenen **GehaltArt** des **GehaltKategorie**s nach **Person** und **Datum** erfasst.
5. Innerhalb einer Familie gibt es zum bestimmten **Datum** und **Summe** von unterschiedlichen **Ausgaben den Personen**. Neben ihrer **Funktion** werden auch **Summe** in verschiedenen **KostenArt** des **KostenKategorie**s nach **Person** und **Datum** erfasst.
6. Besonderes Interesse besteht an der **PercentageGrenz**. Daher sollte alle **KostenArt** angemessen zugeordnet werden. (*Prozedur*)
7. Außerdem dürfen die Erwachsene also **Ab18** Alkohol kaufen und müssen den notwendigen Lebensunterhalt bezahlen (Miete, NebenKosten, Internet, Versicherung, Rente, Urlaub und Alkohol. (*Prozedur*).
8. Überwachen Sie, ob eine Person unter 16 Jahren mehr als 200 ausgegeben hat.(*Trigger*)



Diagram_HomeDBD2

Details aller Dateien

Verzeichnis	Dateinamen des Skripts	Beschreibung	Verwendete Technik
001-DB-Create-Skripts	001_01_createDatabase_HomeDBD2.sql	Erstellung der Datenbank HomeDBD2	
	dbo.tb_Ausgaben.Table, tb_Eingaben.sql, dbo.tb_Familie.Table, dbo.tb_GehaltArt.Table, dbo.tb_GehaltKategorie.Table, dbo.tb_Geschlecht.Table, dbo.tb_KostenArt.Table, dbo.tb_KostenKategorie.Table, dbo.tb_Person.Table, dbo.tb_Verwand.Table	Erstellung der Tabelle tb_Eingaben, welche die Eingaben der Person einkommen	
	IX_gruppiert_Ausgaben.sql IX_nichtGruppiert_GehaltArt	Indizes	
002-Abfragen und Sichten	002_001_Query_Ausgabe_per_KategoriePerson.sql 002_001_View_Ausgabe_per_KategoriePerson.sql	Abfrage der Ausgaben per Kategorie per Person	Mit Abfrage Elementen von Left outer join, inner join und systemfunktion Month
	002_002_Query_Ausgabe_per_MonthPerson.sql 002_002_View_Ausgabe_per_MonthPerson.sql	Abfrage der Ausgabe per Month per Person	Mit Abfrage Elementen von Left outer join, inner join und systemfunktion Month
	002_003_Query_Eingabe_per_KategoriePerson.sql 002_003_View_Eingabe_per_KategoriePerson.sql	Abfrage der Eingabe per Kategorie per Person	Mit Abfrage Elementen von Left outer join, inner join
	002_004_Query_Eingabe_per_MonthPerson.sql 002_004_View_Eingabe_per_MonthPerson.sql	Abfrage der Eingabe per Month per person	Mit Abfrage Elementen von Left outer join, inner join und systemfunktion Month
	002_005_SELECT_fuer_tf_BilansPerson.sql	Abfrage der Eingabe und Ausgabe für ein person	Mit Abfrage Elementen von inner join und systemfunktion Convert
	002_006_Query_KostenArtCount.sql 002_006_View_KostenArtCount.sql	Abfrage wie viele der Ausgabe per Person	Mit Abfrage Elementen von inner join und systemfunktion Count
003-StoredFunctions	003_001_tf_Ausgabe_per_KategoriePerson.sql 003_001_tf_test_Ausgabe_per_KategoriePerson.sql	Tf umrechnet die Liste von alle Ausgaben für ein monat, ein person und ein Kategorie	Mit Abfrage-Elementen von left outer join, order by verwendet
	003_002_tf_Ausgabe_Datumvon_undbis.sql 003_002_tf_test_Ausgabe_Datumvon_undbis.sql	Tf Die Funktion geben alle ausgaben Zwischen Zwei Datum	Mit Abfrage-Elementen von left outer join, inner join und order by verwendet
	003_003_tf_Eingabe_per_KategoriePerson.sql 003_003_tf_test_Eingabe_per_KategoriePerson.sql	Tf umrechnet die Liste von aller Eingaben für ein monat, ein person und ein Kategorie	Mit Abfrage-Elementen von inner join und order by verwendet
	003_004_tf_Eingabe_per_MonthPerson.sql 003_004_tf_test_Eingabe_per_MonthPerson.sql	Tf Umrechnet alle eingaben ein person per month	Mit abfrage elementen von inner join Mit systemfunktion Month verwendet
	003_005_tf_Balance_perPersonDatumVonBis.sql 003_005_tf_test_Balance_perPersonDatumVonBis.sql	Tf umrechnet Alle eingabe und ausgabe für ein person per month	Mit Systemfunktion Convert, mit Abfrage-Elementen von inner join, union, verwendet
	003_006_tf_KostenArtCount.sql 003_006_tf_test_KostenArtCount.sql	Tf umrechnet wie viele mals per Kostenart per month	Mit systemfunktion count, mit abfrage elementen von inner

			join, group by verwendet
	003_001_sf_Getage.sql 003_001_test_sf_Getage.sql	Sf geben die alter von einem person	Mit systemfunktion getdate, datediff, year, month, day und auch if else schleife
	003_002_sf_SumofEingaben.sql 003_002_sf_test_SumofEingaben.sql	Sf Geben sum of alle Eingaben	Mit systemfunktion sum und month
	003_003_sf_SumofAusgaben.sql 003_003_sf_test_SumofAusgaben.sql	Sf Geben sum von aller Ausgaben	Mit systemfunktion month und sum Mit Abfrage elementen von inner join
	003_004_sf_SumofAusgabePerKategorie.sql 003_004_sf_test_SumofAusgabePerKategorie.sql	Sf Geben sum von alle Ausgaben per kategorie	Mit systemfunktion sum und month
	003_005_sf_istKostenarterlaubtab18.sql 003_005_sf_test_istKostenarterlaubtab18.sql	Sf kostenart erlaubt ab18 oder nicht	Mit Abfragen Elementen von
	003_006_sf_GetPercentage.sql 003_006_sf_test_GetPercentage.sql	Sf Geben percent für ein kategorie	Mit systemfunktion Round und mit Abfrage inner join
004- StoredProcedures	004_01_CREATE_sp_AddAusgabe.sql 004_01_sp_Test_AddAusgabe.sql	Procedure für Insert in Ausgabe tabellen	Unter Verwendung eines IF Anweisung und einer selbst definierten Funktion (dbo.sf_istKostenarterlaubtab18, dbo.sf_GetAge, dbo.sf_GetPercentageKategorie)
	004_02_sp_Backupmitzeitstempel.sql 004_02_sp_test_Backupmitzeitstempel.sql	Procedure für Backup mit Zeitstempel	Unter Verwendung eines ErrorMessage systemfunktion, convert, Error_Line und Error_Procedure
005-DML- Trigger	005_01_tr_Ausgaben_Kind_hoher100.sql 005_01_TEST_tr_Ausgaben_Kind_hoher100.sql	Ausgabe testen - pruefen, ob Kind Sume > 200 -- in tb_Ausgaben	Unter Verwendung eines IF Anweisung und einer selbst definierten Funktion(dbo.sf_GetAge)
006- Login_Userrolle	006_Create_User_AndreaSchreiber.sql BenutzerSohn.sql BenutzerSohn_Aktivieren.sql	DB-Benutzer mit nur AndreaSchreiber	
007-Backup	007_001_SuperSkript_HomeDBD2.sql HomeDBD2-20221028-110551773.bak	Gesamtskript für Datenbank HomeDBD2 incl aller DB-Objekte alle Tabellen, Sichten, Funktionen, Prozeduren, Trigger alle Logins, User, Rollen alle Daten	
008_Ausgaben_import	008_001_Ausgaben_importieren.sql	Importieren das csv file ins Ausgabe Taball	